ProtectCall: Call Protection based on User Reputation

Ibrahim Tariq Javed, Khalifa Toumi, Noel Crespi
Institut Mines-Telecom
Telecom SudParis, Evry, France
Email: {Ibrahim_Tariq.Javed, Khalifa.Toumi, Noel.Crespi}@telecom-sudparis.eu

***Abstract*—Web calling services are exposed to numerous social security threats in which context of communication is manipulated. A attacker establishes a communication session to send numerous simultaneous pre-recorded advertisement calls (Robocalls), distribute malicious files or viruses and uses false identity to conduct phishing. User identification alone is not sufficient to provide a high level of trust between communicating participants. Therefore, we propose '*ProtectCall*' a trust model that allows web calling services to estimate the trustworthiness and reputation of their users based on the evaluation of three parameters: authenticity, credibility and popularity. The main objective of *ProtectCall* is to protect web communication services from social security threats. *ProtectCall* allows users to make decisions based on the trustworthiness of their communicating participants.**

Keywords*—WebRTC; Trust; Reputation; Recommendation; Popularity.

1. Introduction

Web Real-Time Communication (WebRTC) standard [1], is an open source web technology that provides real-time communication capabilities to browsers and web applications via simple APIs. With the launch of WebRTC, any web page can now provide communication services in an ubiquitous manner. WebRTC facilitates context based communication where information and conversational channels dealing with the same context can be provided simultaneously. Most importantly, WebRTC is used as the underlying technology for building web-centric communication solutions [2]. These web communication platforms enable features of cross-domain interoperability, identity portability and enhanced QoS that the current Over-The-Top services (such as Watsapp and Skype) do not offer [3].

Due to its strong emphasis on secure communication, WebRTC has a major advantage advantage over most of the existing VoIP solutions [4] [5]. However, web calling services are exposed to several threats in which context of communication is manipulated. For example, an attacker can send numerous simultaneous pre-recorded advertisement calls (Robocalls), distribute malicious files or viruses and use false identity to conduct phishing. This creates several security concerns for the potential adopters of WebRTC technology. In order to enhance user security, privacy and satisfaction, mechanisms are required to minimize such kind of unwanted and insecure communication. Estimating the level of trust between communicating peers is critical in reducing the uncertainty and risk involved in establishing a connection with an unknown party [6]. Various trust computational models already exist for P2P networks: EigenTrust [7], PeerTrust [8] and PowerTrust [9]. However, they are all particularly designed for file sharing applications and do not consider the threats and vulnerability inherent to web communications.

The importance of trust relationships in WebRTC has been highlighted in various works such as [10] [11] and [12]. The trust relationship between communicating peers is based on user identification facilitated by Independent Identity Providers (IdPs) [13]. This is essential to define trust, as identifying the communicating participant is the first step. However, identification alone cannot guarantee the trustworthiness of a peer. No matter how strong, independent and efficient authentication mechanisms may be, they are unable to predict the behavior of a caller. To anticipate behavior, reputation-based techniques are the most practical and effective solution that can be used over the Internet [14] [15].

Therefore, we present '*ProtectCall*' a reputation based trust model that protect users from social threats over web calling services. This model is used to enhance the overall security of a WebRTC calling service by overcoming the risk involved in establishing a communication session, especially with an unknown party. Most importantly, it will reduce unwanted and undesired communication activities by differentiating between legitimate and malicious peers. The major contributions of this paper can be summarized as follows:

- A threat model is introduced to describe the potential risk involved in web communication services. This risk is elaborated in a summary of the approaches adopted by malicious peers to enhance their reputation.
- A reputation based trust model is proposed by introducing three trust parameters: authenticity, popularity and credibility. Authenticity describes the genuineness of a peer based on the recommendations received by its communicating participants. The credibility parameter defines the sincerity of a participant in giving accurate recommendations. Whereas, popularity is used to categorize peers based on their acceptance and recognition

within the network.
- The feasibility and effectiveness of the model is shown under various types of web communication security threats.

The rest of the paper is structured as follows: the related work is described next in Section 2, and the security threats and risks involved in web communication are summarized in Section 3. The *ProtectCall* model based on three trust parameters: credibility, popularity and authenticity is proposed in Section 4. Simulations are conducted in Section 5, to prove the feasibility and effectiveness of the *ProtectCall* model under various threat scenarios. Finally, we present our conclusions and expectations for future work in Section 6.

2. Related Work

This section provides a comprehensive literature review of the relevant studies on WebRTC peer authentication and trust management, as well as a survey of existing trust computational models.

In WebRTC security architecture, there are two types of identities associated with any peer [16]. The first is a peers screen name, which is managed by the website itself in order to allow its subscribers to discover and contact each other [17]. The second type is the service-independent identity that is managed by the IdP to allow communicating participants to identify each other [5]. Moreover, peers also verify each other over an established media path to protect themselves from any man-in-the-middle attacks [4]. However, identification alone does not guarantee the trustworthiness of a peer [6]. For instance, if Bob is able to verify that Alice@yahoo.com is really owned by Alice, this does not imply that Bob should trust Alice in establishing a communication session.

Several researchers have highlighted the importance of defining trust in WebRTC. For instance in [12], new trust requirements are provided for WebRTC security architecture. Partially and full trust models for identity provisioning are presented in [11]. Trust relationships between different entities of WebRTC architecture are studied in order to present various trust issues that exist due to the introduction of IdP into web call model [10]. In [6], the importance of evaluating trust between communicating participants is highlighted. However, no attempts were made to establish trust between communicating peers.

In web communities reputation is widely used to predict the behavior of others [14] [15]. The existing reputation models are broadly classified into recommendation and interaction based trust models [18]. Most of the trust models in P2P networks are based on recommendations, wherein peers use the network structure to gather information about others. Some example are EigenTrust [7], PeerTrust [8] and PowerTrust [9] which leverage on the structure and relations within the network to collect recommendations. However, these models fail to capture actual interactions between peers. In contrast, on-line social networks use the behavior to evaluate trust. For instance, STrust [19] is a social trust model based on user interactions within the social network whereas in [20] a user's social reputation is used to evaluate trust.

In web communication both the graphic structure and call behavior provide vital information that can be used to predict the trustworthiness of communicating peers [21]. The network structure of call graphs shows how peers relate to each other whereas the frequency, duration and nature of their calls are important indicators to show their acceptance within the community. There are very few hybrid trust models that exist in literature. We therefore choose to explore this area of research by presenting a first hybrid trust model for web communication services.

3. Threat Model

In order to build a reliable and efficient trust computation model it is necessary to anticipate the behavior of those peers that cause harm over the communication network. Moreover, it is also necessary to foresee the mechanisms adopted by such peers to avoid their detection in reputation systems. In this section, we categorize the various types of malicious activities that make web communication services potentially insecure. Next, we present a set of common strategies adopted by peers to avoid their detection in reputation systems.

We use VoIP social security threats [22] [23] to classify the various types of malicious and unwanted behavior found in web communication services:

- ***Voice Spam***: Spam over Internet Telephony (SPIT), is automatically dialed unsolicited pre-recorded bulk phone calls that are broadcasted over Internet telephony for marketing and phishing purposes.
- ***Viruses and malware***: Web communication services are vulnerable to viruses, spywares and malwares. Malicious peers may distribute corrupted or virus-infected files that can be detected by browser or anti-virus.
- ***False Identity***: Impostors are those peers that deliberately use a false identity to communicate with others in order to gain some benefit or commit a fraud.
- ***Voice Phishing***: In phishing callers impersonate legitimate companies such as bank to gain access to victims confidential information.
- ***Telemarketing***: Telemarketers use high pressure sales techniques to persuade customers to buy their products which is usually considered as an unethical business practice.

Using the adversarial powers of malicious peers in reputation systems [15] [7], we summarize common strategies adopted by such peers in order to enhance their reputation:

- ***Liars***:- Malicious peers usually give false feedback to legitimate peers over the network in order to distort their reputation.
- ***Traitors***:- Malicious peers may behave properly for a period of time to earn a good reputation before they start behaving maliciously. This is an act of deception adopted by peers to maintain high reputation.

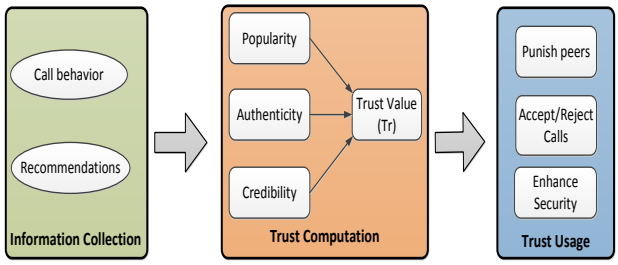


Figure 1. ProtectCall Trust Model

- ***Collusion***:- Malicious peers cooperate with each other in order to enhance their reputation. Such malicious peers provide misinformation about each other to enhance their reputation.
- ***Sybil Attacks***:- When a malicious peer claims multiple false identities to attack a system by falsifying a high reputation. Malicious peer can gain unfair advantage over a recommendation system by conducting a Sybil attack.
- ***White Washing***:- White washing occurs when a peer sheds its reputation by purposely leaving and re-entering the network.

4. ProtectCall Trust Model

We define trust in web communication as the firm belief that a communicating peer will act legitimately and securely over the communication session. Trust is dynamic in nature and thus may increase or decrease with time. Recent events are more important than old ones, since older events might become irrelevant over time. Therefore, we choose to define trust over a specified time period $\hat{T}$ that is divided into n subintervals $[t_0, t_1], [t_1, t_2], ..., [t_{n-1}, t_n]$.

In this section, we detail a hybrid reputation based trust model called '*ProtectCall*'. Figure 1 shows the framework for *ProtectCall* trust model. The information in *ProtectCall* is collected from two sources, (i) recommendations from communicating participants, and (ii) behavior based on communication sessions. Trust however is computed using three trust parameters: (i) Authenticity, (ii) Credibility and (iii) Popularity. The evaluated trust 'Tr' is used in various way to protect and secure web communication. For instance, a call request can be accepted or rejected based on the callers trust value. Service providers can use the evaluated trust to detect and punish malicious peers by blocking their calls or simply removing them from the network. Moreover, it can be used to enhance security over the session by controlling the amount of information. For example, a bank website providing remote financial assistance can use the trust value of a caller to limit the amount of information that it provides over the call.

4.1. Authenticity

Authenticity describes the legitimacy of a communicating peer to behave in an acceptable and desirable manner. It is based on the recommendations received from the communicating participants of each peer. In *ProtectCall*, feedback in terms of user satisfaction is bound to each call so that both participants recommend each other based on their experience. For instance, a peer will be rated malicious if it is a spammer, a telemarketer, being malignant or using a false identity. If the peer behaves in a desirable manner it is rated as legitimate. Any peer p_j can rate its communicating participant p_i as follows:

$$Rec_{p_j \rightarrow p_i} = \begin{cases} +1 & \text{for legitimate} \\ -1 & \text{for malicious} \end{cases} \quad (1)$$

If n_{p_i} are the total number of communicating participants of peer p_i then the authenticity is evaluated conventionally as the average aggregate of all of the recommendations received over peer's lifespan:

$$Auth(p_i) = \frac{\sum_{j=1}^{n_{p_i}} Rec_{p_j \rightarrow p_i}}{n_{p_i}} \quad (2)$$

where $Auth(p_i) \epsilon [-1, +1]$. However, trust is dynamic in nature and may increase or decrease with new interactions. In conventional evaluation recent ratings play insignificant role in altering peer's trust value. Malicious peers may easily adapt strategies to fool the recommendation system. For example, a peer may build a good reputation and then start acting maliciously occasionally. Peers recent behavior can be captured by weighting recommendations based on their positioning in time. Therefore, we model peer's authenticity in terms of the number of recommendations received over n subintervals of a specified time period $\hat{T}$ (for instance 3 weeks or 3 months). The authenticity at time t_i is represented as:

$$Auth_{t_i}(p_i) = \frac{\sum_{k=1}^{n} w_k \sum_{j=1}^{n_{p_i}^k} Rec_{p_j \rightarrow p_i}^k}{\sum_{k=1}^{n} n_{p_i}^k} \quad (3)$$

where n are the total number of subintervals of time period and $n_{p_i}^k$ are the total number of recommendations for p_i in k^{th} interval where $1 \leq k \leq n$. Each interval $[t_{k-1}, t_k]$ is weighted based on its position. Recommendations that occur in the older intervals of the time period are weighted less than the recommendations in recent intervals. We use the position weight w_k defined in [24] for each interval, using $w_k = \frac{k}{S}$ where $S = \frac{n(n+1)}{2}$. Recommendations older than the specified time period are discarded. The choice of time period $\hat{T}$ and number of intervals n is a matter of trust evaluation policy.

In Sybil attack, a malicious peer claims multiple false identities to attack the system in order to gain high reputation. Such false identities are highly unlikely to have high number of incoming and outgoing call requests, as their sole purpose is to enhance reputation of a particular peer.

Social Reliability (SR) of a peer is used to discard such recommendations in the following manner:

$$Rec^{k}_{p_j \rightarrow p_i} = \begin{cases} Accepted & \text{if } I(p_j) > I_{th} and O(p_j) > O_{th} \\ Discarded & \text{otherwise} \end{cases} \quad (4)$$

where $I(p_j)$ and $O(p_j)$ are the number of incoming and outgoing call requests whereas I_{th} and O_{th} are incoming and outgoing call thresholds respectively. We consider peers to be socially reliable if their incoming and outgoing call requests are above a particular threshold. However, this forces to discard some credible feedbacks from peers having very low interaction in the network.

In equation 3, each recommendation is considered equally to evaluate trust for peer p_i. However, not every recommendation is credible enough to be considered. Choosing the correct recommendation is critical to estimate trust accurately. Therefore, we choose to weight each recommendation with the credibility of its recommender as follows:

$$Auth_{t_i}(p_i) = \frac{\sum_{k=1}^{n} w_k \sum_{j=1}^{n^k_{p_i}} Rec^{k}_{p_j \rightarrow p_i} \times Cr(p_j)}{\sum_{k=1}^{n} n^k_{p_i}} \quad (5)$$

where $Cr(p_j)$ is the credibility of the recommender p_j introduced in the subsection 4.2.

4.2. Credibility

Credibility represents the sincerity of a peer in giving correct recommendations. Credibility of a peer is evaluated within the specified time period such that $Cr(p_j) \epsilon [0, 1]$. We introduce four sincerity metrics to determine the credibility of a recommender: reliability, similarity and honesty.

Reliability (R): The reliability metric is based on two assumptions; that legitimate peers are more likely to give correct recommendations, and that malicious peers are more likely to submit false recommendations. Reliability considers the evaluated trust in the following simple manner:

$$Reliability_{t_i} = \begin{cases} Auth_{t_{i-1}} & \text{if } Auth_{t_{i-1}} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

The second assumption is generally true, but the first assumption may not be true at all times, as legitimate peers may occasionally provide false recommendations to other legitimate peers.

Similarity (S): This measures the similarity of each peer with its neighbors (communicating participants) in terms of similar recommendations. To find the similarity for each peer p_j, set of common peers that were rated by peer p_j and its neighboring peers are first obtained. The similarity is then evaluated in the following manner:

$$Similarity = \frac{SR}{SR + DR} \quad (7)$$

where SR is the total number of similar recommendations and DR is the total number of dissimilar recommendations. This metric is based on the fact that any peer is more likely to communicate with other legitimate peers. Therefore, legitimate peers should have high similarity whereas malicious peers should have low similarity.

Honesty (H): This metric indicates the honesty of a recommender by considering the degree in which the recommendations given by the peer are different from the evaluated trust value. A recommendation provided at time t_i is considered as honest if its sign is same as the sign of evaluated authenticity value at t_{i-1} and is considered as a lie otherwise.

$$Honesty = \frac{HR}{\textit{Total number of Recommendations}} \quad (8)$$

where HR are the number of honest recommendations. However, the honesty of a peer is difficult to predict when there are a high number of liars within the network, as the evaluated trust would mostly be based on false recommendations.

4.3. Popularity

The popularity of a communicating peer determines its state of being accepted by other members of the network. We aim on categorizing peers based on their behavior within the network. We observe that malicious peers due to their undesired activities have very low in-degree [25]. Moreover, malicious peers such as spammers cannot avoid making a large number of outgoing calls. However, their call duration is usually very short as callees try to end the communication very quickly after noticing their malicious behavior. On the other hand, legitimate peers have high in-degree and significant talk time [21]. Therefore, we use in-degree, out-degree and talk time to rank peers in the network in order to estimate their popularity.

We use SymRank [26], a modified version of the famous PageRank algorithm which considers both incoming and outgoing links to rank peers. The rank of each peer $RankCall(p_i)$ is evaluated within a specified time period as follows:

$$RankCall_{t_i}(p_i) = Rank_{In}(p_i) - Rank_{Out}(p_i) \quad (9)$$

where $Rank_{In}(p_i)$ computes rank based on incoming links:

$$Rank_{In}(p_i) = \frac{1-d_{f_i}}{N} + d_{f_i} \sum_{p_j \in M(p_i)} \frac{Rank_{In}(p_j)}{L(p_j)} \times tt(p_i, p_j) \quad (10)$$

and $Rank_{Out(p_i)}$ computes rank based on outgoing links:

$$Rank_{Out}(p_i) = \frac{-d_{f_o}}{N} + d_{f_o} \sum_{p_j \in M'(p_i)} \frac{Rank_{Out}(p_j)}{L'(p_j)} \times tt(p_i, p_j) \quad (11)$$

$M(p_i)$ are set of peers that link to peer p_i, and $L(p_j)$ are the number of outgoing links of $L(p_j)$. Where $M'(p_i)$ is the set of peers that p_i links to and $L'(p_j)$ are the number incoming links of node p_j. d_{f_i} is the incoming damping factor set to 0.85 which is the value used in PageRank algorithm. Whereas, d_{f_o} is the outgoing damping factor

TABLE 1. RANKCALL AND POPULARITY

Acceptance	RankCall	Pop
Highly Popular	$LowestRank \leqslant RankCall < Rank_{th2}$	+1
Popular	$Rank_{th2} \leqslant RankCall < Rank_{th3}$	+0.5
Undetermined	$Rank_{th3} \leqslant RankCall < Rank_{th4}$	0
Unpopular	$Rank_{th4} \leqslant RankCall < Rank_{th5}$	-0.5
Highly Unpopular	$Rank_{th5} \leqslant RankCall \leqslant HighestRank$	-1

chosen to be 0.25 to ensure the convergence of the algorithm [26]. In addition, we use total talk time $tt(p_i, p_j)$ between two peers to weight the incoming and outgoing links. The talk time shows the importance of trust relationship between two peers.

In $RankCall$ the incoming links give credit to and outgoing links take credit from a peer. Peers having high in-degree with highly trusted incoming calls are ranked highest. Whereas, peers with large number of out-links having low talk time are more likely to be malicious and thus are ranked the lowest. $RankCall$ can be used to categorize peers based on their popularity. For example, Table 1 shows a service provider using various ranking thresholds between the lowest and highest rank to categorize peer into five popularity sets namely: highly popular, popular, undetermined, unpopular and highly unpopular. The $Pop(p_i)$ is the popularity of peer p_i described as a number in the range $[-1, +1]$.

4.4. Trust Metric

Various formats are used to interpret trust. We choose to express trust as a number between -1 and $+1$. This representation facilitates in illustrating the amount of trust as well as the amount of distrust associated with a peer. The final trust $Tr(p_i)$ for a peer p_i is the sum of its authenticity $Auth(p_i)$ and its popularity $Pop(p_i)$:

$$Tr(p_i) = \alpha \times Auth(p_i) + (1 - \alpha) \times Pop(p_i) \qquad (12)$$

where α is the reputation factor that ranges from $[0, 1]$. This is the weight assigned to the authenticity and popularity parameters. The selection of reputation factor to quantify the influence of each parameter on the evaluated trust itself is a research problem that deserves attention of its own. For instance, a service provider may choose $\alpha = 0$ in order to use popularity parameter to punish highly unpopular peers from the network. A user may choose to completely rely on recommendations to evaluate trust by setting $\alpha = 0$. Moreover, both authenticity and popularity can be used simultaneously to estimate trust.

5. Experimental Evaluation

We conduct five set of experiments to show the feasibility and efficiency of *ProtectCall* Model by using authenticity and popularity parameters. The objective of these set of experiments is to evaluate the robustness of our solution against different malicious behaviors of peers. The first experiment evaluates the feasibility of *ProtectCall* in the presence of traitors and the second compares various sincerity metric in the presence of false recommendations. The third and fourth experiments tests the *ProtectCall* model under Sybil and collusive attacks respectively. Lastly, the fifth experiment uses the popularity parameter to categorize peers based on their call behaviors.

5.1. Simulation Setup

We implemented a simulator to test the feasibility and effectiveness of our model in web calling services. The simulator evaluates trust over a network of communicating peers using authenticity and popularity parameters. The structural properties of telecom call graphs were used to generate a network of communicating peers. The degree distribution of telecom call graphs follows power law $P(d) = d^{-\gamma}$ where d is the degree and γ is the power law exponent [27].

Network model: The simulator uses a BarabasiAlbert model [28] to generate a random scale-free network of communicating peers based on the preferential attachment mechanism. The main parameters are summarized in Table 2. We consider 300 communicating peers to generate a network. Experiments are also conducted by varying the number of communicating peers. However, no major difference in the evaluated results were found. A statistical analysis of call data records [29] was used to set the in-degree and the out-degree power law exponents to be between $1.5 < \gamma < 2.5$. The network is simulated with a synthetic call workload using real world call characteristics [21]. The call duration is generated using normal distribution where legitimate peer have talk time between $124 - 204$ seconds, whereas malicious peers have call duration less than 20 seconds. In the recommendation system legitimate peers are considered to rate other legitimate peers correctly with a probability of 0.8 whereas malicious peers always rate legitimate peers falsely.

Trust computation: We used the *ProtectCall* model to differentiate between legitimate and malicious peers. A peer is considered trustworthy if the evaluated trust value is greater than zero, otherwise it is considered untrustworthy. The performance of *ProtectCall* is tested by evaluating the trust computation error in the presence of malicious peers present in the network. An error occurs when *ProtectCall* incorrectly identifies the behavior of a communicating peer, where either a malicious peer is estimated as trustworthy or

TABLE 2. SIMULATION PARAMETERS

Notation	Description	Value
N	Number of communicating peer	300
γ	In-degree and Out-degree Power Law Exponent	1.5-2.5
C	Clustering coefficient	0.75-0.8
$P(LL)$	Probability legitimate rates legitimate correctly	0.8
$P(ML)$	Probability malicious rates legitimate correctly	0
n_{exp}	# of experiments over results are averaged	10
n	Number of intervals	7
tt_L	Talk time of a legitimate peer (sec)	124-204
tt_M	Talk time of a malicious peer (sec)	$\leqslant 20$

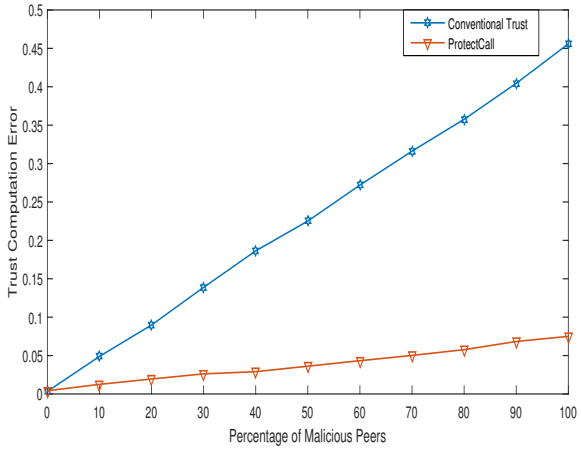


Figure 2. Trust computation error with respect to percentage of malicious peers.

a legitimate peer as untrustworthy. Thus, trust computation error is the total number of errors occurred divided by the total number of communicating peers. Trust computation error is evaluated by varying percentage of malicious peers in the network between 0-100%. All of the results were averaged over 10 runs of the experiment.

5.2. Experiment 1

This experiment was conducted to examine the effectiveness of *ProtectCall* model against traitors.

Evaluation: We consider a time period divided into 7 equal subintervals over which calls are placed. 50% of malicious peers are set to be traitors. Traitors behave well in the initial days of the time period to gain good reputation after which they start acting maliciously. The other 50% behave maliciously throughout the time period. The conventional trust is evaluated using equation 2 compared with the dynamic trust evaluation of *ProtectCall* using equation 3.

Discussion: Figure 2 represents the trust computation error against various percentages of malicious peers in the network. The performance of conventional approach drops as the number of malicious peers are increased. This is due to the fact that traitors can easily fool the conventional trust evaluation by maintaining a respectable reputation value. On the other hand, *ProtectCall* performs very well under increasing number of malicious peers. This is due to the fact that it considers dynamic nature of trust in evaluation and thus is efficiently able to detect traitors in the network.

5.3. Experiment 2

In this experiment, the reliability, similarity and honesty are used as credibility in order to compute authenticity represented by $ProtectCall_R$, $ProtectCall_S$ and $ProtectCall_H$ respectively.

Evaluation: In this experiment, malicious peers rate legitimate peers falsely whereas malicious peers rate other malicious peers correctly.

Discussion: Figure 3a represents the trust computation error with respect to percentage of malicious peers in the network. Firstly, we observe that the conventional approach is very sensitive to peers who provide false recommendations as it does not considers peer's credibility. However, using reliability metric in $ProtectCall_R$ the false recommendations can be filtered out considerately. Honesty metrics in $ProtectCall_H$ proves to be much more effective than $ProtectCall_R$, as it is able to detect liars in the network. We also observe that similarity metric $ProtectCall_S$ is not very effective in estimating peer's credibility. This is due to the fact that in a highly clustered network a large number of calls are placed between malicious and legitimate peers. Therefore, it is difficult to estimate peer's credibility on the bases of similarity with its neighbors. We do not consider cooperation between malicious peers in this experiment therefore no error is detected when all peers in the network are considered to be malicious.

5.4. Experiment 3

In this experiment, the feasibility of *ProtectCall* model is tested under collusive attack where malicious peers cooperate with each other in order to enhance their reputation.

Evaluation: We divide malicious peers into two sets of collusive groups. Malicous peers within the group cooperate with each other by giving false recommendations to each other. However, they provide correct recommendations when communicating with malicious peers outside their group.

Discussion: In the last experiment, honesty performs much better than similarity and reliability. Therefore, we choose to examine *ProtectCall* performance using the honesty metric in the presence of collusive groups. From Figure 3b we can observe that the *ProtectCall* performs very well when the collusive group is small. However, as the number of peers in the collusive group increase the performance of *ProtectCall* decreases substantially. When high number of peers cooperate with each other then the value of authenticity is largely based on false feedbacks. Therefore, it remains difficult for the honesty metric to detect false recommendations. However, a very large group is highly unlikely to occur in communication networks. There can be high number of disjoint collusive groups present in the network but a group containing extremely large number of peers is not probable. Hence, *ProtectCall* provides a reasonable defense against collusive attacks.

5.5. Experiment 4

In this experiment, the effectiveness of *ProtectCall* against Sybil attacks is analyzed.

Evaluation: In this experiment, 50% of the malicious peers present in the network carry out a Sybil attack by creating 30 fake profiles in the network. These fake profiles are considered to have low interaction rate as their sole

TABLE 3. USING POPULARITY TO CATEGORIZE PEERS

Category	Popular	Unpopular	All peers
Size	30	30	300
Average In-Degree	122.37	2.5	40.13
Average Out-Degree	39.7	40	40.13
Average Talk Time	167.79	7.06	85.24

purpose is to provide false recommendation to a particular peer.

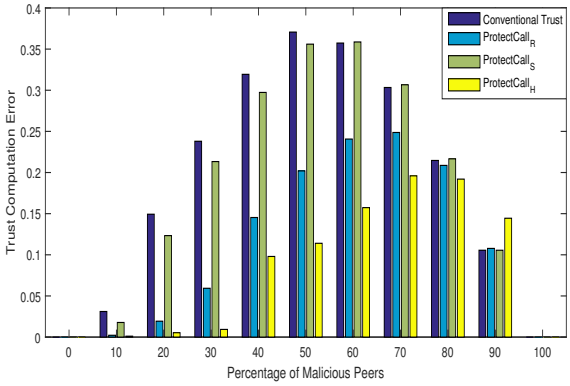
Discussion: We choose to examine *ProtectCall* performance using the honesty metric in the presence of Sybil attack. When comparing Figure 3c with Figure 3a, we notice a considerable decrease in the performance of conventional trust approach. This is due to the presence of large number of fake profiles in the network. On the other hand, *ProtectCall* accepts or discards a recommendation based on the recommender's social reliability as described by equation 4. In this experiment *ProtectCall* considers a peer to be socially reliable if it has in-degree higher than 5. This value is selected based on the average in-degree of the network. Hence, *ProtectCall* is able to nullify Sybil attacks successfully using social reliability parameter. Figure 3c clearly shows that *ProtectCall* provides a effective defense mechanism against Sybil attacks conducted by communicating peers.

5.6. Experiment 5

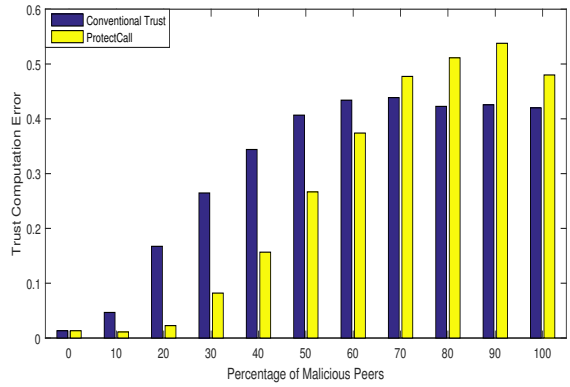
In this experiment, peers are categorized based on their popularity within the network.

Evaluation: In this experiment, the popularity parameter is used to rank and categorize peers into two sets: unpopular and popular. The network consists of 300 peers with 25% malicious peers. We consider 10% of the lowest rank peers to be unpopular whereas 10% of the highest ranked peers to be popular. The Pop value assigned to unpopular peers is -1 and popular peers is $+1$, whereas remaining peers have a Pop value of 0.

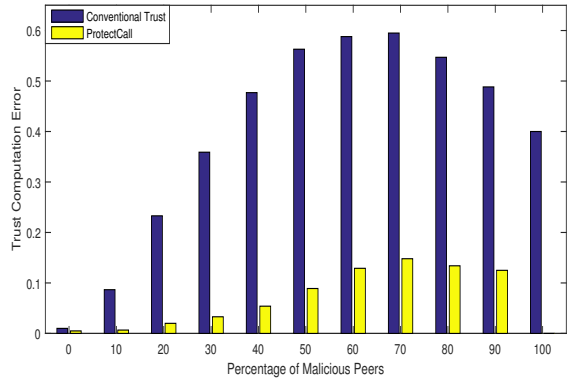
Discussion: Table 3 summarizes the call characteristics of peers in the networks such as average out-degree, in-degree and talk time. For popular peers, in-degree is much higher than the average in-degree of the network which shows their acceptance and importance within the community. On the other hand, unpopular peers have very low in-degree due to their malicious behavior. However, they do stay connected to other peers in the network as they have high out-degree. The out-degree of unpopular peers is close to the average out-degree of the network. This shows that peers having low in-degree and high-out degree are ranked the lowest in the network as they are more likely to be malicious. Moreover, the average talk time of unpopular peers remains to be very low as compared to the average talk time of popular peers.



(a) Trust computation error in the presence of liars



(b) Trust computation error in the presence of collusive grouping



(c) Trust computation error in the presence of Sybil attack

Figure 3. Performance of ProtectCall in the presence of malicious peers

6. Conclusion

WebRTC has emerged as a powerful communication tool that is leading to new innovative ways to communicate over the web. However, it provides an attractive medium to generate Spam calls, distribute malicious content, conduct phishing and fraudulent telemarketing. Therefore, we present '*ProtectCall*' a trust model that protects web communication services from social security threats. To the best of our knowledge, this is the first ever attempt to estimate the trustworthiness of communicating peers in WebRTC calling services. This model enhances user satisfaction and security by minimizing malicious and undesired activities

over web communication services. It reduces the risk involved in establishing a communication session especially when connecting with an unknown caller. The results proves effectiveness of *ProtectCall* model over a network of communicating peers.

Acknowledgments. This work has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 645342, project reTHINK.

References

[1] A. Bergkvist, D. C. Burnett, C. Jennings, A. Narayanan, and B. Aboba, "WebRTC 1.0: Real-time Communication Between Browsers," W3C Working Draft, Tech. Rep., May 2016.

[2] E. Bertin, S. Cubaud, S. Tuffin, N. Crespi, and V. Beltran, "Webrtc, the day after: What's next for conversational services?" in *Intelligence in Next Generation Networks (ICIN), 2013 17th International Conference on*, Oct 2013, pp. 46–52.

[3] S. Becot, E. Bertin, J. M. Crom, V. Frey, and S. Tuffin, "Communication services in the web era: How can telco join the ott hangout?" in *18th conference on Innovations in Clouds, Internet and Networks (ICIN)*, Feb 2015, pp. 208–215.

[4] S. Loreto and S. P. Romano, "Real-time communications in the web: Issues, achievements, and ongoing standardization efforts," *IEEE Internet Computing*, vol. 16, no. 5, pp. 68–73, Sept 2012.

[5] R. L. Barnes and M. Thomson, "Browser-to-browser security assurances for webrtc," *IEEE Internet Computing*, vol. 18, no. 6, pp. 11–17, Nov 2014.

[6] I. T. Javed, K. Toumi, N. Crespi, and A. Mohammadinejad, "Br2br: A vector-based trust framework for webrtc calling services," in *2016 IEEE 18th International Conference on High Performance Computing and Communications; (HPCC)*, Dec 2016, pp. 522–529.

[7] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proceedings of the 12th International Conference on World Wide Web*, ser. WWW '03. New York, NY, USA: ACM, 2003, pp. 640–651.

[8] L. Xiong and L. Liu, "Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities," *IEEE Trans. on Knowl. and Data Eng.*, vol. 16, no. 7, pp. 843–857, Jul. 2004.

[9] R. Zhou and K. Hwang, "Powertrust: A robust and scalable reputation system for trusted peer-to-peer computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 4, pp. 460–473, 2007.

[10] R. Copeland, K. Corre, I. Friese, and S. E. Jaouhari, "Requirements for Trust and Privacy in WebRTC Peer-to-peer Authentication," Internet-Draft, Tech. Rep., September 2016.

[11] V. Beltran, E. Bertin, and N. Crespi, "User identity for webrtc services: A matter of trust," *IEEE Internet Computing*, vol. 18, no. 6, pp. 18–25, Nov 2014.

[12] V. Beltran, E. Bertin, and S. Cazeaux, "Additional Use-cases and Requirements for WebRTC Identity Architecture," Internet-Draft, Tech. Rep., March 2015.

[13] C. Jennings, T. Hardie, and M. Westerlund, "Real-time communications for the web," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 20–26, April 2013.

[14] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decis. Support Syst.*, vol. 43, no. 2, pp. 618–644, Mar. 2007.

[15] S. Marti and H. Garcia-Molina, "Taxonomy of trust: Categorizing {P2P} reputation systems," *Computer Networks*, vol. 50, no. 4, pp. 472 – 484, 2006, management in Peer-to-Peer Systems.

[16] E. Rescorla, "WebRTC Security Architecture," IETF Internet Draft, Standards Track, June 2016.

[17] A. Johnston and D. Burnett, *WebRTC: APIs and RTCWEB protocols of the HTML5 real-time web*. Digital Codex LLC, 2012.

[18] W. Sherchan, S. Nepal, and C. Paris, "A survey of trust in social networks," *ACM Comput. Surv.*, vol. 45, no. 4, pp. 47:1–47:33, Aug. 2013.

[19] S. Nepal, W. Sherchan, and C. Paris, "Strust: A trust model for social networks," in *2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications(TrustCom)*, Nov 2011, pp. 841–846.

[20] K. Zhao and L. Pan, "A machine learning based trust evaluation framework for online social networks," in *2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications*, Sept 2014, pp. 69–74.

[21] N. Chaisamran, T. Okuda, G. Blanc, and S. Yamaguchi, "Trust-based voip spam detection based on call duration and human relationships," in *2011 IEEE/IPSJ International Symposium on Applications and the Internet*, July 2011, pp. 451–456.

[22] A. D. Keromytis, "Voice-over-ip security: Research and practice," *IEEE Security Privacy*, vol. 8, no. 2, pp. 76–78, March 2010.

[23] A. Keromytis, "A comprehensive survey of voice over ip security research," *IEEE Communications Surveys Tutorials*, vol. 14, no. 2, pp. 514–537, Second 2012.

[24] I. Ray and S. Chakraborty, "A vector model of trust for developing trustworthy systems," in *Computer Security–ESORICS 2004*. Springer, 2004, pp. 260–275.

[25] J. Quittek, S. Niccolini, S. Tartarelli, and R. Schlegel, "On spam over internet telephony (spit) prevention," *IEEE Communications Magazine*, vol. 46, no. 8, pp. 80–86, August 2008.

[26] H. K. Bokharaei, A. Sahraei, Y. Ganjali, R. Keralapura, and A. Nucci, "You can spit, but you can't hide: Spammer identification in telephony networks," in *2011 Proceedings IEEE INFOCOM*, April 2011, pp. 41–45.

[27] V. D. Blondel, A. Decuyper, and G. Krings, "A survey of results on mobile phone datasets analysis," *EPJ Data Science*, vol. 4, no. 1, p. 10, 2015.

[28] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Rev. Mod. Phys.*, vol. 74, pp. 47–97, Jan 2002.

[29] A. A. Nanavati, S. Gurumurthy, G. Das, D. Chakraborty, K. Dasgupta, S. Mukherjea, and A. Joshi, "On the structural properties of massive telecom call graphs: Findings and implications," in *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, 2006, pp. 435–444.