A trust framework for real-time web communications

Ibrahim Tariq Javed

PhD DEGREE IN COMPUTER SCIENCE AND TELECOMMUNICATION

Doctorate jointly delivered by

the Telecom SudParis and University Pierre Marie Curie (UPMC)

Specialty:

Computer Science and Networks

National Thesis Number (NNT):

2018TELE0016

Author:

Ibrahim Tariq JAVED

A Trust Framework for Real-time Web Communications

Ph.D defense on 4^{th} October, 2018. Committee in charge:

Thesis Director	Noel Crespi	Telecom SudParis
Jury President & Examiner	Joaquin GARCIA-ALFARO	Telecom SudParis, France
Reviewer	Jeaseung SONG	Sejong University, South Korea
Reviewer	Gyu Myoung LEE	Liverpool John Moores University, United Kingdom
Examiner	Rami LANGAR	University Paris-Est-Marne-La-Vallée, France
Examiner	Joaquin GARCIA-ALFARO	Telecom SudParis, France
Examiner	Tiziana MARGARIA	University of Limerick, Ireland

Dedication

To my honorable mother Iram Tariq
Life doesn't come with a manual. It comes with a mother. I dedicate this work as a token of my deep love for her. Thank you for all your support and prayers.

To my respected father Tariq Javed
I am especially thankful for your love, your understanding and your continuous support. Thanks for always believing in me.

To my beloved wife Bakhtawar Shahid
I am particularly in debt for your unconditional trust and sincere love. Thanks for always standing by my side and for the fun moments I have shared with you.

Ibrahim Tariq JAVED

Acknowledgment

First and foremost, I would like to thank my God for giving me the strength, knowledge, ability and opportunity to undertake this research and complete it satisfactorily. Without His blessings, this achievement would not have been possible.

I would like to thank my thesis director Prof Noel Crespi, who gave me the opportunity to do this research and provided me all the support and freedom one could wish for. I would like to thank him for all his help, suggestions and encouragement.

More thanks goes to Khalifa Toumi who was mentoring me for the last two years of my P.h.D. We have discussed a lot and I really appreciate as well as enjoy the time working with him. His constructive remarks helped me to build my personal point of view about trust management.

My profound love, respect and thanks goes to my family, my mother Iram Tariq, my father Tariq Javed, my sisters Eman Tariq and Ahlam Tariq, and my wife Bakhtawar Shahid. They have always boosted me with encouragement to achieve this new milestone of my life. I hope they find here the expression of my deep gratitude and appreciation.

Lastly, many thanks to all my friends and colleagues who supported me during my PhD.

Abstract

Abstract

Real-time web conversational services allow users to have audio and video calls over the Internet. Over-The-Top operators such as Google and Facebook offer cost-effective communication services with advanced conversational features. With the introduction of WebRTC standard, any website or web application can now have built-in communication capabilities. WebRTC technology is expected to boost Voice-Over-IP by making it more robust, flexible and accessible. Telco operators intend to use the underlying technology to offer communication services to their subscribers over the web. The web-centric communication platforms aims to offer modern methods of contacting and communicating over the web.

However, web operators are unable to ensure the trustworthiness of their subscribers, since identities are based on self-asserted user profiles and credentials. Thus, they remain exposed to many social threats in which the context between communicating parties is manipulated. An attacker usually misrepresents himself to convey false information to the targeted victim. Typical social threats include phishing, spam, fraudulent telemarketing and unlawful content distribution. To ensure user security over communication networks, trust between communicating parties needs to be established. Communicating participants should be able to verify each other's identity to be sure of whom they are talking to. However, authentication alone cannot guarantee the trustworthiness of a caller. New methods of estimating caller's reputation should also be built in web calling services.

In this thesis, we present a novel trust framework that provides information about the trustworthiness of callers in web communication networks. Our approach is organized in four parts. Firstly, we describe the notion of trust in real-time web communication services. A trust model approach is presented to formally introduce the trust computation parameters and relationships in a communication system.

Secondly, we detail the mechanism of identity provisioning that allows communicating participants to verify each other's identity in a Peer-to-Peer fashion. The choice of authentication protocol highly impacts user privacy. We showed how OpenID Connect used for Single-Sign-On purposes can be adopted for provisioning identities while preserving user privacy.

Thirdly, a trust computational model is proposed to measure the trustworthiness of callers in a communication network. The legitimacy and genuineness of a caller's identity is computed using recommendations from members of the network. On the other hand, the popularity of a caller is estimated by analyzing its behavior in the network. Each subscriber will be able to visualize the computed trust of other members before initiating or accepting a call request.

Lastly, the reputation of a caller is used to combat nuisance calls generated over communication networks. Nuisance calls are described as unsolicited bulk spam phone calls generated for marketing and deceptive purposes. Caller's reputation is computed using the diversity of outgoing calls, call duration, recommendations from

called participants, reciprocity and repetitive nature of calls. The reputation is used to differentiate between legitimate and nuisance calls generated over the network.

Keywords: Trust Computation, Real-time Web Communication, WebRTC, Spam Over Internet Telephony.

Résumé:

Les services de conversation Web en temps réel permettent aux utilisateurs d'avoir des appels audio et vidéo et de transférer directement des données sur Internet. Les opérateurs OTT (OTT) tels que Google, Skype et WhatsApp proposent des services de communication économiques avec des fonctionnalités de conversation évoluées. Avec l'introduction de la norme de Web Real Time Communication (WebRTC), n'importe quelle page Web peut désormais offrir des services d'appel. WebRTC est utilisé comme technologie sous-jacente pour déployer de nouvelles plateformes de communication centrées sur le Web. Ces plates-formes visent à offrir de nouvelles méthodes modernes de contact et de communication sur le web. Contrairement aux réseaux de télécommunication traditionnels, les identités sur le Web sont basées sur des profils d'utilisateur et des informations d'identification auto-affirmés. Par conséquent, les opérateurs Web sont incapables d'assurer la fiabilité de leurs abonnés. Les services de communication Web restent exposés à des menaces dans lesquelles le contexte social entre les parties communicantes est manipulé. Un attaquant se définit comme une entité de confiance pour transmettre de fausses informations à l'utilisateur ciblé. Les menaces typiques contre le contexte social comprennent la fausse représentation d'identité, le hameçonnage, le spam et la distribution illégale de contenu. Afin d'assurer la sécurité sur les services de communication Web, la confiance entre les parties communicantes doit être établie. La première étape consiste à permettre aux utilisateurs d'identifier leurs participants communicants afin de savoir avec qui ils parlent. Cependant, l'authentification seule ne peut garantir la fiabilité d'un appelant. De nouvelles méthodes d'estimation de la réputation de l'appelant devraient également être intégrées dans les services d'appel Web. Par conséquent, dans cette thèse, nous présentons un nouveau cadre de confiance qui fournit des informations sur la fiabilité des appelants dans les réseaux de communication Web. Notre approche est organisée en quatre parties. Premièrement, nous décrivons la notion de confiance dans la communication web en temps réel. Un modèle de confiance est présenté pour identifier les relations de confiance nécessaires entre les entités d'un système de communication. Les paramètres requis pour calculer la confiance dans les services de communication Web sont officiellement introduits. Deuxièmement, nous montrons comment les protocoles Single-Sign-On (SSO) peuvent être utilisés pour authentifier les utilisateurs d'une manière Peer-to-Peer (P2P) sans dépendre de leur fournisseur de service. Nous présentons une comparaison entre trois protocoles d'authentification appropriés (OAuth, BrowserID, OpenID Connect). La comparaison montre que OpenID Connect est le meilleur candidat en termes de confidentialité des utilisateurs. Troisièmement, un modèle de calcul de confiance est proposé pour mesurer la fiabilité des appelants dans un réseau de communication. La légitimité et l'authenticité d'un appelant sont calculées à l'aide de recommandations, tandis que la popularité d'un appelant est estimée en utilisant son comportement de communication. Un abonné d'un service de communication sera capable de visualiser

la confiance calculée d'autres membres avant d'initier ou d'accepter une demande d'appel. Enfin, la réputation d'un appelant est utilisée pour lutter contre les appels nuisibles générés sur les réseaux de communication. Les appels de nuisance sont décrits comme des appels de spam non sollicités en masse générés sur un réseau de communication à des fins de marketing et de tromperie. Les enregistrements de données d'appel et les commentaires reçus par les parties communicantes sont utilisés pour déterminer la réputation de l'appelant. La réputation évaluée est utilisée pour différencier les spammeurs et les appelants légitimes du réseau.

Publications

Thesis Publications

Journal papers

- Ibrahim Tariq Javed, Rebecca Copeland, Noel Crespi,..., Ricardo Lopes Pereira, *"Cross-Domain Identity and Discovery Framework for Web Calling Services"*, Springer Annals of Telecommunication Journal, June 2017.

- Ibrahim Tariq Javed, Khalifa Toumi and Noel Crespi, *" TrustCall: A Trust Computation Model for Web Conversational Services "*, in IEEE Access, vol. 5, pp. 24376-24388, 2017.

Conference papers

- Ibrahim Tariq Javed, Khalifa Toumi, Noel Crespi,*" N-Combat: A Nuisance Call Combating Framework for Internet Telephony "*, IEEE TrustCom 2018, (Accepted).

- Ibrahim Tariq Javed, Khalifa Toumi, Noel Crespi *" ProtectCall: Call Protection based on User Reputation "*, IEEE TrustCom 2017, 1-4 August, Sydney Australia.

- Ibrahim Tariq Javed, Khalifa Toumi, Noel Crespi, A. Mohammadinejad *" Br2Br: A Vector-based Trust Framework for WebRTC Calling Services "*, IEEE HPCC 2016, 12-14 December, Sydney Australia.

- Ibrahim Tariq Javed, Khalifa Toumi, Noel Crespi, *" Browser-to-Browser Authentication and Trust Relationships for WebRTC "*, The Tenth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies UBICOMM, October 9 - 13, 2016, Venice, Italy. [Best Paper Award]

- Ibrahim Tariq Javed, Rebecca Copeland, Noel Crespi,..., Nuno Santo, *" Global Identity and Reachability Framework for Interoperable P2P Communication Services "* 19th International ICIN Conference - Innovations in Clouds, Internet and Networks - March 1-3, 2016, Paris, France.

Standardization

- Rebecca Copeland, Keven Corre, Ingo Friese, Saad El Jaouhari, Ibrahim Tariq Javed, Noel Crespi, Ahmad Bouabdallah, Simon Becot, *" Requirements for Trust and Privacy in WebRTC Peer-to-peer Authentication "*, IETF Internet Draft, Sep 2016.

Contents

List of Figures

Part I

Introduction

" The beginning is the most important part of the work. "
Plato, The Republic

Background and Motivation

Telecom operators that offer traditional voice telephony services are unable to compete with the web-based Over-The-Top (OTT) service providers. OTT communication services use the Internet infrastructure to provide Voice-over-the-Internet-Protocol (VoIP) services to their subscribers. For instance, Skype being one of the most prominent OTT service provider offers free calling between its subscribers and low-cost calling services to non-subscribers. Skype has more than 300 million monthly active users who spent nearly 3 billion minutes per day communicating [SKY03]. Whatsapp is an another popular OTT communication application that allows its subscribers to send and receive voice/video calls, texts, photos, videos, files, and location without paying for the service. OTT communication applications are platform and device independent allowing subscribers to communicate using different devices and operating systems.

The ever growing market of VoIP call services are expected to increase to US $194.5 billion [Per]. With the introduction of Web Real-Time Communication (WebRTC) standard, developers can now embed VoIP capabilities into their websites or web applications. WebRTC allows making phone calls, video calls, text chats and file transfers, directly through web browsers or mobile applications. WebRTC standard is expected to boost VoIP into novel communication platforms that will introduce new modern methods of contacting and communicating over the web. It will allow VoIP market to grow faster in terms of subscribers, revenues and traffic. Telecom operators have started investing into the web paradigm to compete with existing OTT web operators. They intend to develop novel web-centric communication platforms using the underlying Peer-to-Peer (P2P) WebRTC technology to deploy their services. These platforms aim to provide features that current OTT service providers do not offer such as cross-domain interoperability, identity portability, enhanced security and Quality-of-Service (QoS) beyond best effort.

The web-centric communication platforms face two technical challenges related to identity management. The first challenge is user discovery which involves an efficient identity resolution system that maps user identities to the currently available web address of user's device. The second challenge is to ensure the trustworthiness of user identities. In traditional communication networks user identities are considered to be trustworthy, since a subscriber has to proof its identity in person when registering to a service. User's identity is always linked to a secure element such as SIM card or a fixed line identifier. Malicious activities can easily be tracked down by the operator. Regulatory authorities can take necessary actions against malicious callers to secure communication networks. However, identities are managed quite differently by web operators. OTT operators allow anyone to access their services globally by creating a user profile with self asserted user information. The user profile is managed and maintained by the service provider in a centralized manner.

Web communication services remain exposed to several social security threats in which the context between communicating participants is manipulated. Typical threats agaisnt social context include identity misrepresentation, phishing and spam. Identity misrepresentation facilitates an attacker to present fraudulent information, such as a false name, organization, email address, or presence information. Phishing is the most common way to illegally obtain somebody's personal information such as password, bank account number, credit card information over a communication network. Moreover, the free of cost web communication services have attracted telemarketers to generate spam calls. Spam calls can be manually or automatically generated bulk phone calls for marketing and advertisement purposes. Voice spam is considered to be much more disruptive in nature than email or social spam as they require immediate response from the recipient.

WebRTC architecture allows communicating participants to validate user identity during a call. This is facilitated by the use of third party independent Identity Provider (IdP). The use of IdP allows communicating participants to identify each other independent from their service. Identity provisioning is the generation, exchange and verification of identity assertions between communicating participants. Efficient and

secure authentication mechanisms are required for this purpose of idenitfing provisioning. However, authentication alone cannot guarantee the trustworthiness of callers in a communication network. New methods to compute caller's reputation should also be built in web calling services. The information about trustworthiness of callers will enhance user security by protecting them from different social security threats present over communication networks.

Thesis objectives and contributions

In this subsection, we present the main objectives of this thesis. To address each objective we provide one or more contributions. The main aim of this thesis is to design a novel trust framework for real-time web communication services. The framework aims to provide information about trust in a communicating party. The main objectives are as follows:

- To define the notion of trust in real-time web communication. The aim is to provide a formal definition of trust, identify the required trust relationships and specify necessary parameters required to compute trust in web communication services.
- To securely authenticate communicating participants in a privacy enabled manner. The aim is to allow communicating participants to securely verify each other's identity while preserving their privacy.
- To design a trust computational model that is able to estimate the trustworthiness of callers over the network. The aim is to compute caller's reputation by analyzing its behavior and information provided by other members of the network.
- To use caller's reputation to combat nuisance calls over communication network. The aim is to compute callers reputation in order to differentiate between nuisance and legitimate call.

Our approach to design the trust framework is organized into four major parts. We discuss the major contributions in this thesis as follows:

First Contribution: In our first contribution, we present a vector based trust model for representing trust in WebRTC security architecture. The model formalizes the notion of trust, distrust and mistrust. A trust vector is used to represent trust relationships between a truster and trustee. The model defines three trust relationships in real-time web communications. The first vector is used represent trust between user and its service provider. The second vector represents trust between user and identity provider whereas third vector represents trust between communicating participants. In order to compute trust, the model defines three parameters namely experience, reputation and identification. The experience parameter is based on the past performance of the trustee in a given context over a specified period of time. The identification parameter determines the strength in authentication process of the communicating participant. Whereas the reputation parameter is the weighted aggregate of the average recommendation received about a trustee.

Second Contribution: Subscribers of communication services want to be certain that they are speaking to the person that he/she claims to be. For this purpose, communicating participants should be able to verify each other's identity before establishing a communication session. Identity provisioning is the generation, exchange and verification of identity assertions between communication participants. RTCWEB working group propose the use of existing SSO protocols for identity provisioning. The selection of a particular authentication protocol profoundly affect the overall security and privacy of user identities. In this contribution, we propose the use of OpenID Connect (OIDC). We compared OIDC with OAuth and Browserid in terms of privacy properties such as anonymity, unlinkability, audience control etc. The comparison shows that OIDC is the best candidate in terms of user privacy for the exchange of identity assertions between communicating participants.

Third Contribution A trust computational model is proposed to compute the reputation of callers over web conversational services. The trust model is comprised of three components namely, information collection, trust computation and trust dissemination. The data is collected using recommendations from other members of the network and call data records. The reputation of the caller is determined by evaluating authen-

ticity and behavioral trust. Authenticity trust describes the legitimacy of a caller by collecting recommendations from other members of the network, whereas behavioral trust determines the caller's popularity based on its communication behavior. The trust dissemination unit allows any user to visualize the computed trust of other members of the network before initiating or accepting a call request. A network of communicating peers is simulated to test the feasibility and effectiveness of our proposed trust model. We compare the performance of our model with the popular recommendation based trust model PeerTrust. The comparison shows that TrustCall performs better in terms of user satisfaction and trust computational error.

Fourth Contribution: In our fourth contribution, we propose a nuisance call combating mechanism to effectively mitigate manually and automatically generated nuisance calls over communication networks. The caller's reputation is computed using total call duration, out-degree, reciprocity and repetitive nature of calls, feedback and the reliability of communicating participants. To address the dynamic behavior of callers, the concept of a dual time window is used. White washing allows spammers to shed their bad reputation and re-enter the network with a new identity. Therefore, a watchdog mechanism is proposed to combat whitewashing attacks in communication services. We evaluate the performance of our proposed solution in the presence of four types of callers namely ordinary, specific, telemarketers and auto-dialers. We compare our approach with two existing spam combating models namely Progressive Multi Gray-leveling (PMG) and Discrete Event System Specification (DEVS). Our approach is able to detect spammers while having a very low false negative rate when compared to existing threshold based spam detection methods.

Figure 1 summarizes our thesis plan. The organization and content of each chapter are as follows:

Chapter 1: This chapter presents the literature review on real-time web communication and trust management. The chapter's introduction is presented in Section 1.1. Section 1.2 presents a comparison of telco and web operated communication services. This section also introduces WebRTC standard and novel web-centric communication platforms be-

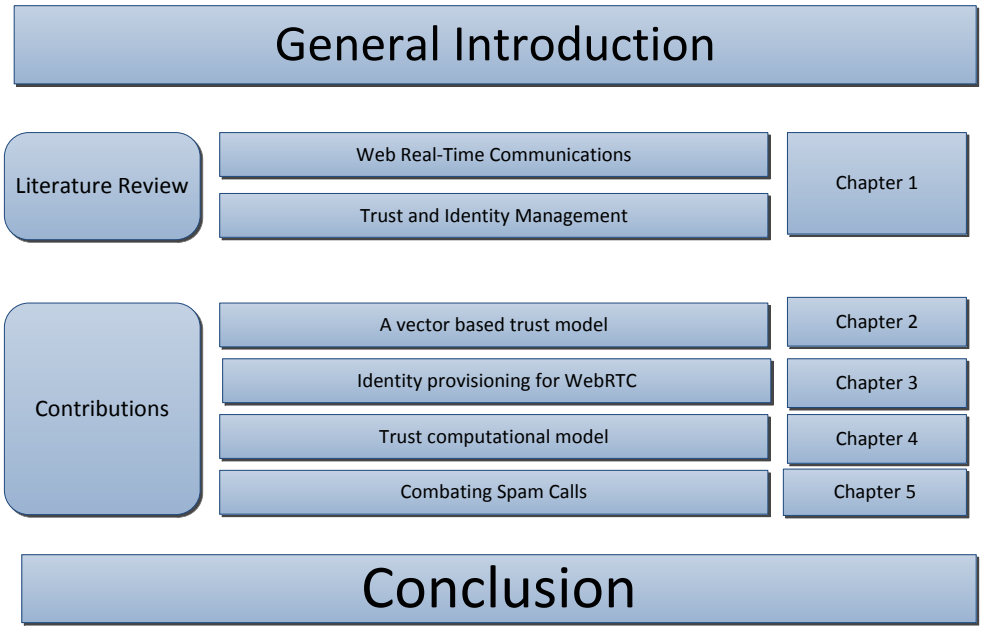

Figure 1: Thesis Plan

ing deployed. Section 1.3 presents the threat taxonomy for real-time web communication applications. The threat taxonomy include threats against availability, confidentiality, integrity and social context. This state of the art motivates the use trust to combat social security threats. Therefore, Section 1.4 presents the literature review for the computation of trust in online applications. It also presents the challenges for trust computation in real-time web communication.

Chapter 2: This chapter presents a trust model "Br2Br" to define the notion of trust in WebRTC architecture. The chapter is structured as follows: Section 2.1 presents the introduction. Section 2.2 describes the WebRTC security architecture presented by RTCWEB working group. The vector based trust model " Br2Br " is presented in Section 2.3. The parameters used for trust evaluation are introduced and formalized in Section 2.4. Three trust relationships of a user in WebRTC are identified in Section 2.5. Section 2.6 presents a user scenario utilizing our trust model. Finally, in Section 2.7 we provide our conclusion for this chapter.

Chapter 3: This chapter studies the WebRTC call model in detail and presents a comparison between suitable authentication protocols for

user identification. The chapter is structured as follows: Section 3.1 presents the introduction of the chapter. Section 3.2 gives a brief introduction of WebRTC call model and Section 3.3 explains the process of user identification in P2P fashion. Section 3.4 describes the architecture of three SSO authentication protocols namely OAuth, BrowserID and OpenID Connect. Section 3.5 presents a comparison between the three authentication protocols in terms of user privacy when used for WebRTC call model. The chapter concludes with Section 3.6.

Chapter 4: In this chapter, a novel trust computational model "*TrustCall*" is introduced. The chapter is structured as follows: the introduction is present in Section 4.1 the related work is described in Section 4.2. A threat taxonomy for real-time web conversational services is presented in Section 4.3. The three components of '*TrustCall*' model: information collection, trust computation and trust usage are described in Section 4.4. In Section 4.5, various experiments are conducted to prove the feasibility and effectiveness of the TrustCall model. Finally the conclusion is provided in Section 4.6

Chapter 5: In this chapter, a nuisance call combating framework "*N-Combat*" is introduced to mitigate automatically and manually generated spam calls. The chapter is structured as follows: Section 5.1 presents the introduction of chapter. The related work is provided in Section 5.2 and the nuisance call detection model is described in Section 5.3. The framework with the details of its components is presented in Section 5.4. In Section 5.5, experiments are conducted to prove the feasibility and robustness of our proposed framework. Finally, we offer our conclusions and recommendations for future work in Section 5.6.

Overview of my Publications

During this thesis 7 papers (5 conferences and 2 Journals) were published in well reputed conferences and journals. The author is listed as the first author in all the papers mentioned. The author also contributed to IETF Internet draft titled as " Requirements for Trust and Privacy in WebRTC Peer-to-Peer Authentication " published on September 26,

2016.

The title of paper I is *" Global Identity and Reachability Framework for Interoperable P2P Communication Services "*. The paper was published in the proceeding of 19th International Innovations in Clouds, Internet and Networks (ICIN) 2016 held on March 1-3, 2016, Paris, France. The paper is a joint contribution with partners of H2020 EU reTHINK project partners. This paper consists of an initial prototype of a web-based global identity and reachability framework that allow users of a specific domain to discover, locate and identify communication participants from different domains.

The title of paper II is *" Browser-to-Browser Authentication and Trust Relationships for WebRTC "*. The paper was published in the 10th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM) held on October 9-13, 2016, Venice, Italy. This paper details the WebRTC identity architecture and how identity provisioning can be conducted in an efficient and secure manner. The paper received ***Best Paper Award*** in the conference.

The title of paper III is *" Br2Br: A Vector-based Trust Framework for WebRTC Calling Services "*. The paper was published in the proceeding of IEEE High Performance Computing and Communications (HPCC) 2016 held on 12-14 December, Sydney Australia. The conference is ranked B according to CORE ranking 2018. The paper consist of a trust framework presenting different trust relationships and trust parameters that can be used to compute trust.

The title of paper IV is *" Cross-Domain Identity and Discovery Framework for Web Calling Services "* published in Springer Annals of Telecommunication Journal, June 2017. The paper provides an cross-domain identity and reachability framework for web communication platforms. The identity framework allows global discovery; cross-domain interoperability and identity portability which current identity management systems do not offer. The Journal has an impact factor of 1.412.

The title of paper V is *" ProtectCall: Call Protection based on User Reputation "*. This paper was published in the proceeding of IEEE TrustCom 2017 held on 1-4 August, Sydney Australia. The conference is ranked A according to CORE 2018. The paper proposes three param-

eters namely authenticity, credibility and popularity to compute trust between communicating peers in web communication service.

The title of paper VI is *" TrustCall: A Trust Computation Model for Web Conversational Services "* published in IEEE Access, vol. 5, pp. 24376-24388, 2017. This paper consists of a trust computation model for web conversational services. IEEE Access Journal received an impact factor of 3.244 in the 2016 JCR release. It is an award-winning, multidisciplinary, all-electronic archival journal.

The title of paper VII is *" N-Combat: A Nuisance Call Combating Framework for Internet Telephony "*. This paper is published in the proceeding of IEEE Trustcom 2018. The paper provides solution to mitigate spam calls by using caller's reputation. The conference is ranked A by the Computing Research and Education Association of Australasia.

Project Contribution

The work in this thesis has been performed as a part of European Union's Horizon 2020 research and innovation project titled as " re-THINK Trustful hyper-linked entities in dynamic networks ". The work is performed under grant agreement No 645342 by a consortium of 3 global telecom, 2 SMEs, 2 Universities and 2 leading research institutes. The partners include Orange, Deutsche telekom, Portugal Telecom, Eurescom, Quobis, Apizee, IMT, TU Berlin, Fokus Frauhofer and INESCID.

The main goal of the reTHINK project was to design and prototype a new, non telecom centric, but web-centric P2P service architecture. A novel communication platform is proposed that provides solutions to manage real-time communication capabilities over the open Internet. The platform allows telecom operators to deploy their communication services over the web in order to compete with existing OTT services. It facilitates cross-domain interoperability by allowing subscribers of different domains to communicate with each other. The framework decouples authentication from the service providers by using third party independent identity providers. A trust engine component is introduced to make sure conversation can take place at the right level of confidence. The trust engine provides information about trust in a communication party.

The contributions of this thesis were used as part of developing the trust engine.

Part II

State of the Art:

Chapter 1

Background and Literature Review

" An efficient telecommunications network is the foundation upon which an information society is built."
Talal Abu-Ghazaleh

Contents

1.1 Introduction

Web communication applications allow users to have voice/video calls, web conferencing, and direct data transfers in a real-time fashion. OTT operators such as Google, Microsoft and Facebook offer communication services that are globally accessible and cost-effective. Over the past few years enterprise and personal communication have shifted from traditional dedicated networks towards VoIP platforms. VoIP is cloud-based technology that allows calls to be sent over digital data using Internet. VoIP offers simpler and cost effective technology than traditional phone services. Majority of people and businesses have switched to VoIP in order to save money and use enhanced calling features. Moreover with the introduction of WebRTC standard any website can now provide voice/video calling and file sharing facilities. Thus WebRTC acts as an modern catalyst for VoIP that will allow web communication services to become more robust, user friendly, and flexible.

In this chapter, we present our literature review on real-time web communication services. We detail OTT services, WebRTC standard and emerging web-centric communication platforms. We also list the major security threats present over VoIP communication services. This includes threats against availability, confidentiality, integrity and social context. In this thesis, we particularly focus on combating social security threats by computing trust. For this purpose we present the major concepts of trust management. We also detail the challenges for trust computation in web communication services.

1.2 Real-Time Web Communication Services

Web communication applications allow users to have real-time voice/video calls, web conferencing, and direct data transfers over the Internet. In this section, we compare web communication services with traditional telephony services. We present the major advantages and disadvantages of OTT communication services. We also introduce the WebRTC standard and emerging communication platforms using the underlying technology.

Telco Federated Model	Walled Garden Model
Limited innovation, not flexible enough	Much more competitive and agile
Access controlled communication services	Not constrained by standards
Geographically constrained	Globally available services
Reliable service with guaranteed QoS	Best effort service
Identity portability between service providers	No portability of identity or user data
Well defined standards to enable universal interoperability	Can't interoperate with users from other domains

Figure 1.1: Comparison of Telecommunication models

1.2.1 Telecommunication Architectures

Traditional operator-enabled services such as voice telephony are loosing their significance due to the presence of web communication services [BBC+15a]. VoIP have transformed the traditional well-established services and business models of telecommunication providers. VoIP is a group of technologies that makes communication possible over data networks [KP09]. The transmission of data is conducted over a general purpose packet switched network instead of the traditional dedicated circuit-switched network. Currently, telecommunication architectures can be categorized into Telco-Federated Model or on a Walled-Garden Distribution Model. The main features of both models are summarized in Figure 1.1.

- **Telco-Federated Model:** Telco operators follow a vertical service distribution model in which services are bundled together with identity management, application platform and network access. Communication services are tightly bound to their access network which is deployed over a specified geographical area under the enforcement of regulation bodies. However, Telco-Federated Model offer

reliable communication services by managing guaranteed QoS. The well defined standards also allow universal interoperability between different service providers. It further facilitates identity portability where users can switch their service provider without losing their identity.

- **Walled-Garden Model:** Web communication services such as Whatsapp and Skype follow a private walled garden approach. These services create their own communication standards and protocols which results in silos of users. Users of a particular service provider are restricted to only communicate with subscribers of their own domain. This results in the creation of isolated communication platforms over the web. This model relying on loosely coupled applications, device-side platforms and data-centers. This allows developers to create innovative and competitive communication services with global reachability and non-standardized interfaces. The service delivery of these model is unregulated and relies on best-effort delivery.

Over-the-top (OTT) is a general term that refers to services that a subscriber use which operates on top of a network. OTT communication services use the VoIP technology to deliver traffic over Internet. They allow subscribers to communicate by making audio/video calls or sending instant messages. In addition they provide services such as watching videos, play games and advertising. These services are highly accessible to users as their applications can be operated over different hardware and software platforms. OTT operators use Internet infrastructure to provide their services. They only have to invest in the development and maintenance of their applications/websites. Therefore, OTT services are usually provided for free to their subscriber. However they may use advertisements or payed for value added services to generate revenue. Some examples of OTT service providers are as follows:

1. **Skype:** Skype [SKY03] is regarded as an IP telephony service that provide free calling from Skype to Skype and low cost calling from Skype to Public Switched Telephone Network (PSTN). Skype has more than 300 million monthly active users of Skype. It is available

on different platforms such as Windows, Linux, MacOS, Android etc.

2. **WeChat:** WeChat [WEC11] is a application and a website which allows mobile text and voice communications services. In January 2017, Wechat had 846 million monthly active users. In addition to making free calls and sending free messages, users of WeChat's services may play games, send money, make video calls, order food, read the news, book a doctor appointment and many other things.

3. **Facebook Messenger:** Facebook [FAC04] was initially started as a social networking website. Currently, Facebook provides a unique platform that allow people to communicate with each other by sending messages and making calls. In January 2017, there were around 1,871 billion monthly active users of Facebook.

4. **Whatsapp:** WhatsApp [WHA09] was founded as a cross-platform mobile messaging company and is now operating as a subsidiary of Facebook. Whatsapp offers simple, secure, reliable messaging and calling services. In January 2017, there were nearly 1 billion of monthly active users of Whatsapp in over 180 countries.

5. **Viber:** Viber [VIB10]is the name of application for making free voice and video calls as well as sending free text and voice messages over the Internet. In January 2017, there were nearly 247 million monthly active users of Viber. Viber application provides its users with a possibility to communicate within the platform as well as making calls to any fixed/mobile telephony numbers all over the world.

1.2.2 WebRTC standard

WebRTC [BBJ+16] is an open source web-based application technology that allows exchange of media and data in a real-time fashion. WebRTC allows browsers and mobile applications to have voice/video calls, chats and P2P file transfers without the installation of any plugins. WebRTC communication is directly controlled by web server using simple JavaScript APIs. With the introduction of WebRTC, any Hypertext

Markup Language (HTML) compatible mobile device can now be used to communicate ubiquitously. Any website can now provide their own communication features without the need to install Skype client or load Flash plugins.

WebRTC is currently being standardized by the combined effort of Internet Engineering Task Force (IETF) and World Wide Web Consortium (W3C). RTCWEB Working Group [RTC] in IETF is responsible for developing the protocols for real-time communication. Whereas, WEBRTC Working Group [WEBb] in W3C defines the APIs required by the JavaScript web applications. Most widely used web browsers such as Google Chrome, Mozilla Firefox and Opera are now supporting WebRTC.

WebRTC offers new capabilities that existing web-based communication systems do not provide [JB12]. Some of them are listed as follows:

1. **Platform independence:** WebRTC allow developers to create applications that can function across different operating systems, browsers and devices.

2. **Secure media:** Mandatory use of encryption for voice and video transfer makes the media channel secure.

3. **Enhanced quality:** Built in voice and video codecs enables interoperability and avoids the need to download codecs.

4. **Reliable session establishment:** The load on server is reduced by establishing direct media connection in order to have better quality connection.

5. **Adaptive to network conditions:** It provides feedback about network conditions.

6. **Interoperability with VoIP systems:** Existing VoIP can interwork with WebRTC using standard protocols.

7. **Signaling is not standardized:** Developers are free to choose their own signaling mechanism for their communication application.

WebRTC breaths new life into VoIP communication services. It allows widespread adoption of VoIP by allowing it to exist inside web browsers

and web applications. With WebRTC web developers can make VoIP capabilities right into their website and web-based applications.

1.2.3 Web-Centric Communication Platforms

In order to compete with existing OTT players such as Google, Facebook and Whatsapp, telecommunication operators must resolve their current limitations [BCT+13]. They should renew the key aspects of their operational structure that includes signaling, identity management and QoS management. This cannot be achieved with existing Telco operated infrastructure such as IMS. Therefore, a new generation of web-centric communication framework is required to renew the architecture of telecommunication services. The new communication platform should follow target service distribution model in which identity management, signaling and network management are decoupled from each other. The architecture should be without standardized interconnection interfaces. The framework should also facilitate interoperability between different service providers allowing subscribers of different domains to communicate freely.

In [BBC+15a] different challenges for the development and deployment of renewed web-based Telco architecture are explained. The three major challenges are listed as follows:

1. **Trustworthy Identity Framework:** The first challenge is the development and deployment of a trustworthy identity framework. This requires removing identity management from the signaling plane by using IdPs to authenticate and verify user identities. The user identities should be globally searchable, cross domain interoperable and portable.

2. **Reliable real-time platforms:** The second challenge is to develop and deploy a platform that offers integrated communication as a feature over the web. The platform should be web developer friendly by using widely available and adopted technologies such as JavaScript and HTML5. The framework should also rely on Software-as-a-Service (SaaS) approach to avoid the developers from the additional burden of managing and maintaining various operations.

3. **End-to-end network QoS:** The third challenge is to offer QoS that is beyond the Internet best effort approach. The best effort delivery is not sufficient for future communication services over the web.

The European founded reTHINK project [retc] describes a new communication framework providing solutions to manage real-time communication capabilities over the open Internet. The project involves three major telecom operator: Orange, Deutsche telecom and Portugal telecom. reTHINK framework uses the underlying WebRTC technology to facilitate P2P communication in a secure and efficient manner [RETb]. reTHINK framework applies to set of use cases [RETa], such as machine-to-machine communication for Internet of Things, OTT services for audio/video chats and smart homes. It allows Telco operators to compete with large OTT web companies by de-perimetrising their communication services. It introduces interoperability between service providers, ecosystem-agnostic development environment, decentralized service delivery and establishes user control over its data and privacy. Some of the main features of reTHINK framework includes:

- **Decentralized session control**: A module of software is dynamically deployed in end user devices to execute session control and media flow management in a peer to peer manner.
- **Global reachability with de-perimeterised services**: The web-centric platform allows services to be accessed globally
- **Non-service-bound identities**: User identities are decoupled from the services by allowing the use of trusted third party identity providers.
- **Measuring confidence level of identities**: Service present information about trust in user identities.
- **Cross domain interoperability**: Users across domains are able search and contact each other.
- **QoS beyond best effort**: The framework offers managed quality of service for reliable and efficient service delivery.

1.2.4 Identity Management

In this subsection we define the basic concepts related to user identities. We compare the identities management of traditional telephony network with web communication services. We also list the challenges for a trustworthy identity management for future web-centric communication platforms.

Identity is an instrument used by an entity in order to provide information about itself to the system. An *identity* is always associated with an entity or generally formed by an unique identifier. A *User* is an entity that uses a service furnished by a service provider. *Identifier* is used to prove ownership of the identity, through credentials which allows a system to make decisions about the associated entity. An identifier is a exclusive index for an identity which is always unique to the system. A *Credential* is used to prove an identity to a system. Credentials ensure a system that an entity truly has the right to use a particular identity.

Identity Management (IdM) is used to manage user identities and control entities access to system resources [SG09]. It provides information about user profile, service features and access policies [SDS10]. It allows the right user to access the right resources at the right time and for the right purpose. IdM systems allows to establish trust between entities such as users, applications, services and devices. *Identity Provider* is an entity that controls user's credentials and provides authentication services. *Identity Provider* can be part of the service provider or independent entity. *Identity Provider* manages user identities, their authentication, authorization and the profile related to their identities [JFH+05].

In traditional telephony world, identities are linked to a publicly known phone numbers that follows international standardized rules. Phone numbers are structured by country, operator and user in order to maintain unique global identification. Identities in traditionally telecommunication services are considered trustworthy as they are based on a secure element such as a SIM card or fixed line phone numbers. User identities in telephony networks are not only used for identification but also for routing purposes. Identities are unique and owned by the service provider. However, number portability across different service providers can be achieved through mutual agreements. Communicating participants trust their own service provider while service providers trust each

other in order to facilitate interoperability.

The situation drastically changes over web communication services since identities over web is simply a combination of user profile and credentials. A user profile is associated with the identity which is stored in a centralized directory. OTT services have service specific formats and authentication procedures which are only applicable within their own administrative domains. This has led to isolated communication platforms dominated by some big players such as Google or Facebook. The abundance of independent services have resulted in multiple unrelated identities of one person over the web. Single Sign On (SSO) solutions relieve the strain on users of managing and maintaining different identities. SSO systems allow users to login to one service which acts as login to another. User authentication is decoupled form the service and delegated to a third-party IdP.

Potential adopters of the WebRTC standard face two technical challenges related to user identities: i) user discovery and ii) identity provisioning. User discovery involves an efficient identity resolution system that maps user identities to the currently available web address of user's device. This is essential in order to establish a communication session for the exchange of media. However, before establishing a connection it is necessary to ensure that users know who they are talking to. For this purpose, WebRTC architecture facilities end-to-end peer authentication using IdP. IdP can be managed by the CSP itself or delegated to a trusted third party. Identity provisioning in WebRTC standard involves the retrieval, delivery and verification of identities between communicating participants. WebRTC proposes the use of SSO protocols for the purpose of identity provisioning such as BrowserID [BA13] and OAuth [Har]. However, SSO protocols are designed for service client authentication. Therefor they require certain modifications in order to facilitate P2P authentication between communicating participants.

One of the main challenge for future web-centric communication platform is the development of a trustworthy identity framework. For future web communication services user identities should be trustworthy and identifiable across different domains. The main challenges of a trustworthy global web identity framework are as follows:

1. **Cross domain inter-operable:** Subscribers of different domains

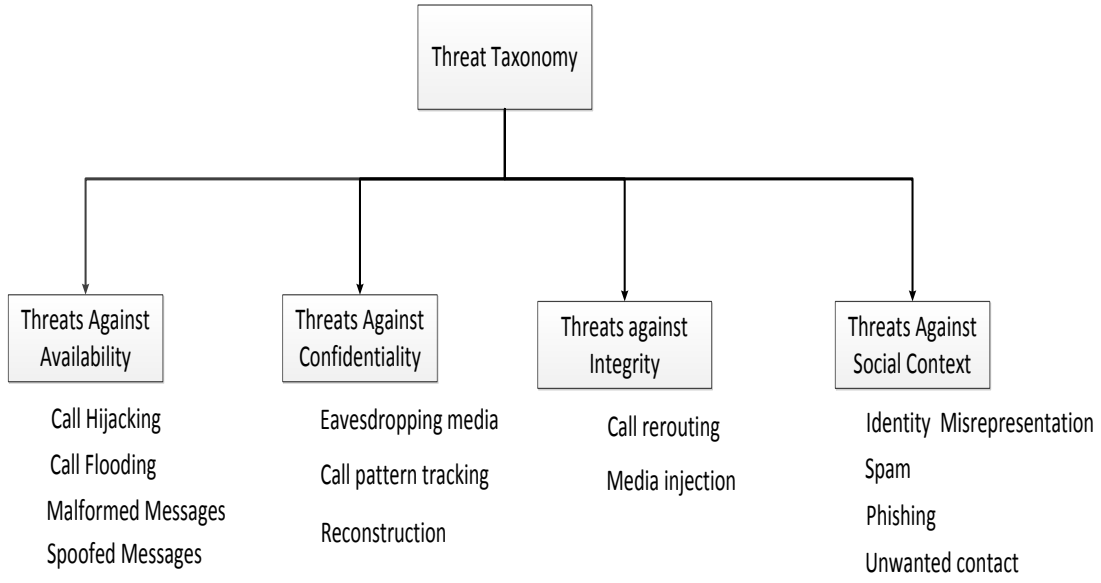

Figure 1.2: Threat Taxonomy

should able to search, identify and communicate with users of other domains.

2. **Identity decoupled from service providers:** Identities should be decoupled from the service by using third party IdPs.
3. **Identity portability:** Users are able to migrate between different service providers without losing their identity information
4. **Trust-enhanced identity:** Beyond user authentication, communication between communicating participants require trust which should be computed and displayed to users.

1.3 Threat Taxonomy for Web Communication

In this section we present the threat taxonomy of real-time web communication service. The taxonomy defines potential security threats to VoIP deployments, services, and end users. We categorize the security threats into four major categorizes: 1) threats against availability, 2) threats against confidentiality, 3) threats against integrity and 4) threats

against social context as shown in Figure 1.2. Each of them are described as follows:

1.3.1 Threats Against Availability

Threats against availability aims at service interruption that are typically in the form of Denial of Service (DoS) attacks. Communication services are supposed to be available for user when it needs it. DoS attack is an attack on the network or device denying it of a service or connectivity. The typical threats against availability include:

1. *Call Flooding:* A heavy flood of signals or media traffic is generated in order to drop the performance of the communication system.
2. *Malformed Messages:* Protocol message with wrong syntax are transmitted to the target server or client.
3. *Spoofed Messages:* Fake spoofed messages are inserted into a communication session in order to suspend the service.
4. *Call Hijacking:* In call hijacking the control of transactions such as registration, call setup or media flow between a user and the network is attained.

1.3.2 Threats Against Confidentiality

Threats against confidentiality include unauthorized means of capturing user information such as media, credentials, identities and call patterns. Confidentiality implies that the user information is protected from unauthorized access. The most popular types of confidentiality threats are:

1. *Eavesdropping:* Eavesdropping include sniffing of the entire signaling or media traffic between two or more users.
2. *Call pattern tracking:* The traffic between different users are analyzed in order to know who is communicating with who, and when.
3. *Reconstruction:* The conversation including voice, video or text between communicating parties is collected, manipulated and reconstructed without their consent.

1.3.3 Threats Against Integrity

Threats against integrity includes the capturing and altering of signaling or media messages. The alteration include deletion, injection or replacement of certain information in the established communication session. Integrity implies that user information remains unaltered by any unauthorized access. Threats against integrity include:

1. *Call rerouting:* The messages are intercepted in the middle of communication path in order to alter the protocol message and reroute the call.

2. *Media injection:* The media traffic is intercepted in the middle of communication to alter, inject unauthorized media and delete certain media information.

1.3.4 Threats Against Social Context

Threat against social context is the manipulation of the social context between communicating parties. This manipulation allows an attacker to misrepresent himself and convey fraudulent information. The typical threats against social context are as follows:

1. *Identity Misrepresentation:* Identity misrepresentation is the intentional presentation of a false identity. Identity misrepresentation may include false information such as false name, profession, organization, number and email address.

2. *Spam:* Spam over communication networks are bulk of unsolicited prerecorded automatically dialed voice or video messages transmitted for marketing and advertisement purposes.

3. *Fraudulent Telemarketing:* Communication networks are used to commit frauds by advertising customers to buy products or services.

4. *Phishing:* Phishing is an illegal attempt to obtain target's personal information such as user-name, password, bank account number, credit card information over the communication session.

5. *Unwanted communication:* Unwanted communication includes cases of harassment, extortion and unlawful content distribution.

1.3.5 WebRTC Security Considerations

WebRTC has big advantage over existing VoIP services in terms of ensuring security. WebRTC has a strong focus on secure communication and thus is currently regarded as one of the most secure VoIP solutions [Weba]. WebRTC enforces important security concepts in all main area [Res16]. Most importantly, it mandates the use of encryption for communication which most existing VoIP services do not. Thus, users of WebRTC enabled applications are assured the safety and privacy of their data. These security consideration are detailed in the IETF Internet draft [Res15]. We briefly describe the major security considerations of WebRTC standard:

1. *Installation risk:* WebRTC technology is installed as part of downloading a suitable WebRTC compatible browser. WebRTC requires no setup, installation or plugins to operate. Thus, there is no risk of installation of malware or viruses when used for communication.

2. *Signaling traffic protection:* WebRTC does not specify any particular signaling protocol that is used to establish media and data channels. It allows the developer to choose the signaling protocol for the application. If the signaling channel is compromised, an attacker could interfere with the session. Therefore, securing the signaling channel by choosing an appropriate protocol is crucial for WebRTC applications.

3. *Media traffic protection:* Eavesdropping is a major security risk which allows an attacker to intercept media. In WebRTC, encryption is enforced on all components of the communication system. Media is encrypted using Secure Real-time Transport Protocol (SRTP) while data streams are encrypted using Datagram Transport Layer Security (DTLS).

4. *Access to local resources:* The browser can access local resources including camera, microphone and files. This allows web applications to record video or audio without user's knowledge. Therefore, in WebRTC application a browser is always required to verify user consent before providing application the access to user's microphone or camera.

5. *P2P authentication:* WebRTC defines an alternative approach for user authentication. It allows communicating participants to authenticate each other without relying on the service provider. This is done using independent Identity Provider (IdP). Users can authenticate themselves to the IdP and use the identity assertions generated by the IdP to identify themselves to their communicating participant.

Until now, most VoIP services provide communication services without encryption. But as WebRTC forbids unencrypted communication, users can be assured that their data remains safe and private. Therefore, WebRTC is considered to be a secure VOIP solution. Although WebRTC facilitate P2P authentication it will still be exposed to threat against social context. Authentication alone cannot guarantee the trustworthiness of communication participant. To combat social context trust in each communicating participant needs to be communicating. Therefore, in this thesis we present solutions to establish trust between communicating participants.

1.4 Trust Management

Trust is used to cope with uncertainty about the intentions of other agents. Trust management systems are used to compute trust so that two agents can build trust relationship in a particular situation. We focus on establishing trust between communicating participants in real-time web communication services. In order to understand how trust is being computed we present the general definition of trust.

We also list the challenges for developing a trust computation.

1.4.1 The notion of Trust

Trust exhibits in many different forms, thus it remains very challenging to define it clearly and precisely. The term trust is being used in variety of meaning. It is widely acknowledged that trust is complex and multidimensional. The concept of trust has been studied in many areas such as psychology, sociology, philosophy, history, law, business and economics.

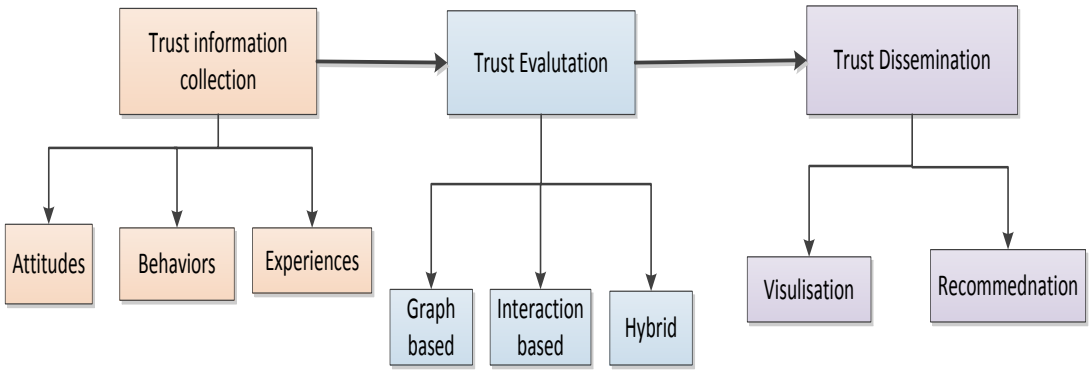

Figure 1.3: Trust system Classification

We find several definitions of trust in literature. Some of them are cited as follows:

1. According to the Oxford dictionary [OXF84], " Trust is a firm belief in the reliability, truth, ability, or strength of someone or something ".

2. Researchers in [MDS95] suggest that " Trust can be viewed as an attribute of risk-taking behavior ".

3. Deutsch [Deu73] defined trust as " The confidence that an individual will find what is desired from another, rather than what is feared ".

4. Grandison and Sloman [GS00] proposed trust to be defined as " The firm belief in the competence of an entity to act dependably, securely and reliably within a specified context ".

5. Chang et al. [CHD05] has specified that trust is " The belief the trusting agent has in the trusted agent's willingness and capability to deliver a mutually agreed service in a given context and in a given time slot ".

1.4.2 Trust System Classification

A trust system is largely composed of three major components: Trust Information Collection, Trust Evaluation and Trust Dissemination as shown in Figure 1.3.

1. **Trust Information Collection:** The information regarding trust can be extracted from attitudes, experiences and behaviors. Attitudes are generally positive or negative belief of an individual and is derived from user's interactions. Experiences describe the perception of an individual which they create while interacting with each other. Whereas, behaviors are patterns of interactions which describe the manner in which a particular entity conducts itself.

2. **Trust Evaluation**: Trust computation models can be categorized into network-based or interaction-based trust models. Network-based trust models are based on the concepts of Friend-of-A-Friend (FoAF) where trust is considered to be transitive in nature. Various techniques are then used to transverse the network and determine trust between two entities. On the other hand, interaction-based trust model capture actual interactions between users. They are applied to networks where volume, frequency and type of interaction are useful in computing trust. Models that use both interactions and social network structure are hybrid trust models.

3. **Trust Dissemination:** Trust can be disseminated by visualization techniques or providing recommendation. Many visualization tools are used to represent the trust value between two entities. Systems can also use the computed trust to generate personal recommendations for users.

1.4.3 Trust Computation in Online Applications

In literature, trust has been applied to various on-line applications. We detail the trust literature in three major on-line applications such as social network, e-commerce and P2P Networks.

Trust in Social Networks

Social network applications allow individuals to disseminate information and connect with their friends, families, governments and enterprises. Privacy remains a major concern for social network users. Therefore, the level of trust that users have with each other critically important in social network applications. In social networks the concept of FoAF is

used to compute trust. The underlying assumption of FoAF network is that trust is transitive in nature. Golbeck et al. [GPH03] used FoAF relationship to indicate level of trust between individuals. The author used ontology to represent trust levels ranging from " trust absolutely " to " distrust absolutely ". In [Gol05b], Golbeck presented TidalTrust that computes trust relationship between two individual who are not directly connected. TidalTrust is based on the assumption that neighbors having high trust relations usually have same opinion about the trustworthiness of a third member. Nepal et al [NSP11] presented a social trust model based on the computation of popularity and engagement trust. The popularity trust refers to the popularity of an individual member in the community whereas the engagement trust refers to the involvement of an individual member in the community. In [HWS09], Hang et al. proposed an algebraic approach for the propagating of trust in social networks using concatenation operator, an aggregation operator and a selection operator. Kuter et al. [KG07] proposed a Bayesian trust model for estimating confidence on the trust information obtained from different social chains. In [VCCdS09], Victor et al. presented a trust propagation model based on fuzzy logics using unavailable and contradictory trust information simultaneously.

Trust in e-commerce Environments

E-commerce refers to on-line transactions where information exchange about purchase and sale is conducted. In e-commerce there is a high level of uncertainty in the transactions itself or the system handling transactions. Trust plays a major role in making satisfaction in on-line transactions. Trust in e-commerce environments are largely based on user feedbacks. eBay [EBA95] is an on-line shopping website in which a buyer gives " positive ", " neutral " or " negative " feedback to the system after each transaction. The feedback score is calculated and displayed on web pages. In [ZM00], the Sporas system is introduced to compute trust for e-commerce applications. In Sporas, the ratings of later transactions are given higher weights as they are more important in trust evaluation. The Histos system proposed in [ZM00] is a more personalized reputation system compared where reputation of a user depends on who makes the query, and how that person rated other users.

In [SHZK05], Song et al. apply fuzzy logic by dividing the sellers into multiple classes of reputation ranks. In [WL08a], Wang et al. propose an approach to evaluate situational transaction trust which includes service specific trust, service category trust, transaction amount, category specific trust and price trust. The trust ratings of a forthcoming transaction are binded with previous transactions to provide specific trust information to buyers. In [WL08b], Wang and Lin presented reputation-based trust evaluation mechanisms to depict the trust level of sellers on forthcoming transactions and the relationship between interacting entities.

Trust in P2P Information Networks

P2P systems allow users to share their resources such as files, audios and videos to other peers in the network without the use of any central point of control. P2P systems are used to create many large-scale content sharing systems at low cost. Due to the open and anonymous nature of P2P network they are vulnerable to attacks by malicious peers that have an incentive to spread viruses and in authentic data. Trust have been proposed to distinguish malicious peers and ensure the quality of services in P2P networks. There are three major trust models for P2P networks namely: *EigenTrust* [KSGM03], *PeerTrust* [XL04] and *PowerTrust* [ZH07]. All of the three trust models shows reasonably good performance for P2P networks under different circumstances. *EigenTrust* adopts a binary rating system where local ratings are collected in order to assign a global trust value for each peer. *EigenTrust* assumes the presence of some pre-trusted peers that help the model to converge. *PeerTrust* model is a reputation based trust model for P2P networks which uses three basic trust parameters and two adaptive factor to compute the trustworthiness of a peer. The basic trust parameters include the received feedback, the total number of transactions and the credibility of the feedback. The two adaptive factors are transaction context factor and the community context factor. *PowerTrust* model leverages the power-law distribution of feedbacks. The *PowerTrust* model is based on the selection of a small number of power nodes that are the most reputable in the system using a distributed ranking mechanism.

1.4.4 Trust in Real-Time Web Communications

There are two major types of trust relationships in communication services: (a) trust between a user and the service provider, and (b) trust between communicating participants.

The trust that exists between user and service provider depends on many factors. For instance in [WL10] it has been suggested that the service interface design plays role is establishing trust. Whereas the easiness of use of online application enhances member's trust in service provider [SH02]. In [JG05] authors found that service performance lead to cognitive trust. Coulter [CC02] finds that the strength of trust relationship between a member and a provided service is dependent upon the length of their relationship. Authors in [MP09a] classifies trust as system trust, which includes security, privacy, and all the logic aspects, and relationship trust, which includes associated entities and application interface.

The second type of trust is between communicating participants. The first step to establish trust between communicating participants is user identification [Res16]. Users usually have to rely on their service provider for authenticating their communicating participant. However the service provider can not necessarily be trusted for this purpose. Therefore WebRTC standard allows communication participants to authenticate each other independent of the service they are using. It is also important to note that trust cannot be built merely on the basis of authentication [CCFJ16]. For example, if Bob is able to reliably and securely verify that Alice@gmail.com is owned by Alice, it does not mean that Bob can trust Alice. Mutual authentication allows communicating participants to identify each other but does not guarantee their trustworthiness. Therefore in order to determine trust between communicating participants methods of estimating trustworthiness and reputation should be built into web calling services.

Therefore in this thesis we aim to design a trust framework that provides information about the trustworthiness of communicating participants. In order to compute trust there are several issues and challenges which needs to be addressed. We list them as follows:

- *Defining trust*: Choosing and defining a relevant trust definition is

critical in trust framework as different definitions of trust exist in the literature that do not converge.

- *Selecting parameters*: It is important to choose the right parameters or combination of parameters to compute trust. Different parameters such as experience recommendation and knowledge are used in the literature to compute trust.

- *Choosing forgetting factor*: For accurate computation of trust ignoring or discarding information that remains no longer relevant in the computation of trust is necessary.

- *Presentation of trust*: Choosing how to represent trust is another issue to resolve. The computed trust can be presented in various forms including a continuous or discrete variable, binary, vector or matrix form.

- *Trust bootstrapping*: Trust bootstrapping is the setting of initial trust values when there is no information about trust available.

- *Time*: Time is a critical parameter that must be defined while evaluating trust as trust levels usually change with time.

- *Robustness*: Trust computational methods are prone to several security. The trust computation method should be robust agaisnt different mechanisms used by malicous users to enhance their trust.

- *Recommendations*: In order to use feedbacks to compute trust there are several questions that needs to be addressed. For instance, How to avoid false feedback problem? How to combine different feedback values? How to understand and share this value?

1.5 Conclusion

In this chapter, we present the literature review related to real-time web communication and trust management. We present a comparison between Telco-Federated and Walled-Garden telecommunication architectures. We introduced WebRTC standard and novel communication platforms using the underlying technology. We list the challenges for

a trustworthy identity management for future web communication. We present the threat taxonomy for VoIP that includes threats against availability, confidentiality, integrity, social context. In order to combat social security trust between communicating participants needs to be evaluated. Therefore, we also present the state of the art related to trust computation. Finally, we present various challenges for trust computation in real-time web communication.

Part III

Trust Framework for Real-Time Web Communications

Chapter 2

Trust Model for WebRTC

" Love all, trust a few, do wrong to none. "
William Shakespeare

Contents

2.1 Introduction

WebRTC standard [BBJ+16] has a different communication paradigm than traditional VoIP communication services, since it is directly controlled by a web server [BBC14]. The RTCWEB working group [RTC] defines the WebRTC security architecture [Res16] The architecture follows a binary trust model based on authentication that will categorize trust value to " no trust " or " complete trust ". The above observation prompt us to propose a new model of trust in which different degrees of trust are presented. Parameters other than authentication are also introduced that can be used to estimate trust efficiently.

In this chapter, we present " Br2Br " a vector based trust model to define the notion of trust in WebRTC architecture. The model formalizes three trust relationships of WebRTC: User-IdP, User-CS and User-User. To define these relationships, the concepts of trust evaluation, trust policy, trust context and parameters influencing trust are presented. The model formalizes dependence of trust on time and on a particular context. The notion of different degrees of trust are introduced, differentiating between trust, distrust and mistrust adopted from Jϕsang's opinion model [JØs98]. The evaluation of trust depends upon three parameters: experience, reputation and identification.

2.2 WebRTC Security Architecture

WebRTC standard is an open source web technology that provides real-time communication capabilities to browsers and web applications via simple APIs. It is envisioned to allow existing telecom operators and OTT players the incentive of having free, open, global and inter-operable communication flows over the web [BCT+13]. A new communication framework using the underlying WebRTC technology is being developed [JCC+16] whereas 3GPP offers the interconnection of WebRTC with IMS [3GP]. WebRTC is expected to bring a wide range of possibilities for corporate and personal communications over the web.

In WebRTC, the calling site is a web server that enables communicating participants to exchange information by providing JS client that executes on the browser. The CS is responsible for providing signaling

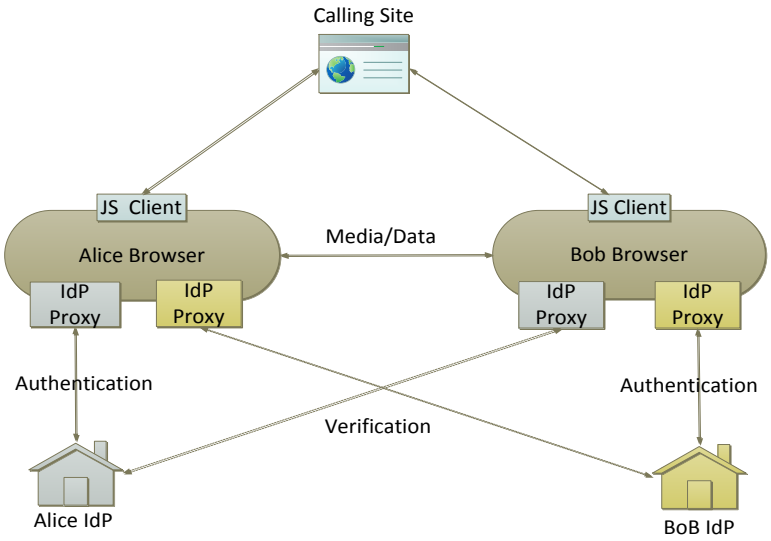

Figure 2.1: WebRTC Security Architecture

between the two parties for the exchange of session parameters, identities, call answer/offer request and user reachable addresses. WebRTC aims in having minimum level of trust in CS by decoupling the authentication procedures from the signaling. Authentication of communicating participants is managed by service-independent IdP [PM03] using existing Single Sign On protocols such as OAuth2.0, OpenID Connect, SAML etc.

Figure 2.1 presents WebRTC security architecture [Res16] in an Alice-Bob call scenario. The CS provides a calling interface for Alice to discover Bob and initiate a call request. To authenticate Alice, Alice's browser downloads an IdP Proxy from Alice's IdP. Upon successful authentication, the IdP server returns an identity assertion containing Alice's identity information. The assertion is attached to the call request sent to Bob via the CS. When Bob receives the call request, Bob's browser instantiates Alice's IdP Proxy and passes on this assertion in order to verify Alice's Identity. Upon successful verification the authentication result is shown to Bob.

Figure 2.2 presents a WebRTC call model where Alice and Bob are subscribers of 'TalkNow' service. They choose to identify themselves via their trusted IdPs. If Alice wants to talk to Bob, she uses the services of Talknow to discover Bob's web address of a currently available user device. Talknow is responsible for providing signaling between Alice and

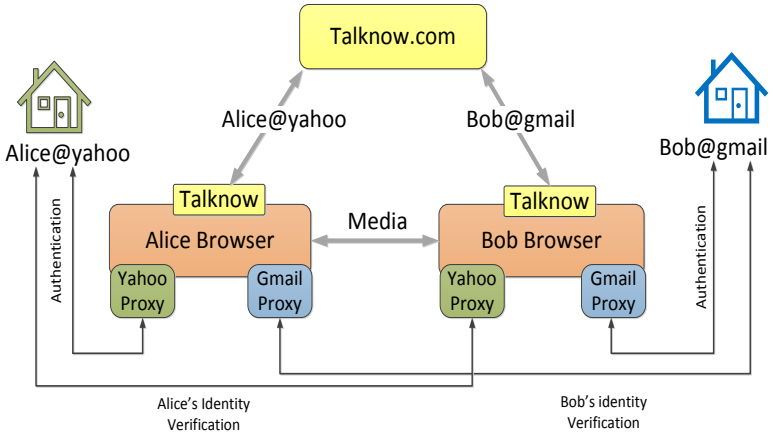

Figure 2.2: User identification in WebRTC Call Model

Bob in order to establish a communication session for the exchange of media. However, before establishing a session, Alice and Bob identify each other using IdP proxy mechanisms. The IdP-Proxy downloaded from an IdP's URL provides an interface between IdP and browser for user authentication and verification purposes.

In WebRTC users trust their calling services to connect them to authorized parties and treat their personal data and accumulated call history confidential. On the other hand IdPs are trusted to store and manage their personal profile information in a secure and efficient manner while preserving their privacy [JHW13]. However, users trust their communication participants to access media/data streams based on the level of identification they provide. In WebRTC, web browser is the only entity that user trusts completely. Therefore it initiates the authentication process for each entity on behalf of the user. However, trust cannot established by merely validating the identities of each entity. An efficient trust management system to estimate the trustworthiness of communicating participants and the service provides is essential.

2.3 Br2Br: A Vector based Trust Framework

We introduce the concept of trust in order to manage the security of information exchanged in WebRTC services by proposing a new trust framework " Br2Br ". Figure 2.3 illustrates the basic concept of the trust framework, which includes three types of trust relationships: User-

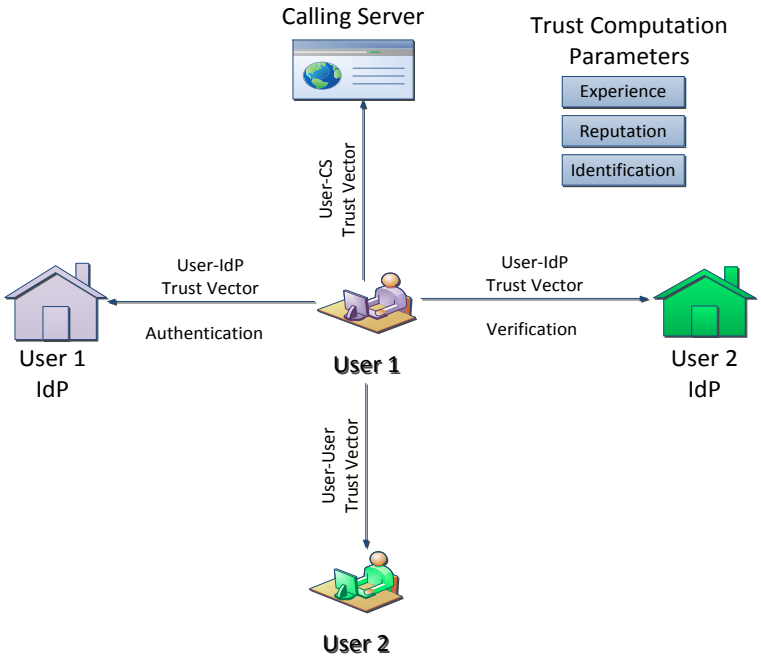

Figure 2.3: WebRTC Trust Model

CS, User-User and User-IdP. In our model trust is influenced by three parameters: experience, identification and reputation.

In this framework, entities are characterized into one of the two types, either a truster, the entity which establishes trust, or a trustee, the entity which is being trusted. A user's browser is the only entity that is considered as a truster, whereas CS, IdP and the communicating participants browsers are considered as trustees. We represent trust as a relation between user U and an entity E within a context c at time $\dot{t}$ such that:

$$(U \xrightarrow{c} E)_{\dot{t}} \tag{2.1}$$

where time $\dot{t}$ is used to characterize the dynamic behavior of a trust relationship over a specific time period $[t_0, t_n]$. The time period is divided into n subintervals $[t_0, t_1], [t_1, t_2], ..., [t_{n-1}, t_n]$. The k^{th} interval is represented as $[t_{k-1}, t_k]$ where $k = 1, 2, ..., n$. The context c is the information that characterizes the situation of entities involved. The notion of context is defined by combining the concepts of trust objectives and trustee aspects.

Definition 1: Trust objective is the purpose to form a trust rela-

tionship, whereas aspects are the characteristics of trustee considered by the truster. Therefore, let CONTEXT, OBJECTIVE and ASPECTS be the set of all possible contexts, objectives and aspects respectively, where each $c \in$ CONTEXT is a tuple (o, a) and where $o \in$ OBJECTIVES and a $\in$ ASPECTS.

We consider two types of trust objectives in WebRTC: to access resources and to provide services. Services include communication and authentication whereas resources include media stream and identity assertions. The aspects of trustee considered are security, reliability, confidentiality and honesty. Therefore the trust relationship in WebRTC is never absolute. A truster will always trust a trustee with respect to the set of specific objectives and aspects defined by the trust context. For example, user U trusts trustee E's security and confidentiality to provide authentication.

Example 1: Alice uses a web calling site " example.com " to place calls from her browser. She trusts the CS to provide communication services in a secure and reliable manner. This does not mean that the CS will also be trusted to access Alice's identity information and media streams. Meanwhile, Alice trusts her friend Bob to access her identity information and media streams in a confidential manner.

In our model, trust is represented in the form of a trust triple (t, d, m), where t represents trust, d represents distrust and m represents mistrust. Unlike single trust values this vector representation of trust allows us to show the amount of trust, distrust and uncertainty within each WebRTC relationship.

Definition 2: We represent trust using a trust triple (t, d, m) where $t, d, m \in [0, 1]$ and $t + d + m = 1$. Trust t is the expectation that an entity will perform reliably, securely and confidentially within a specific context. Distrust d is the expectation that an entity will not perform reliably, securely and confidentially within a specific context and Mistrust m is a level of doubt that an entity will perform reliably, securely and confidentially within a specific context.

The trust relation is a 3×3 matrix. The rows of the matrix correspond to three parameters, experience, recommendation and identification. The formal definition and evaluation of each parameter is provided in Section 2.4. Each of these parameters are represented in the rows of a trust

matrix, where each term of the trust triple represents the columns of trust matrix.

$$\begin{pmatrix} t^E & d^E & m^E \\ t^R & d^R & m^R \\ t^I & d^I & m^I \end{pmatrix} \tag{2.2}$$

The three parameters may not be of equal importance in evaluating trust. For example, a truster U may place more significance on the identification parameter rather than experience and reputation. Therefore, we present a weight scheme vector that specifies the relative weights for each parameter to evaluate trust triples. The user's trust evaluation policy will define the weight scheme vector.

Definition 3: The weight scheme is a vector of the form $(S^E, S^R, S^I)_{U \to E}$. The elements of vector are the weights assigned to the parameters in the trust matrix such that $S^E + S^R + S^I = 1$ and $S^E, S^R, S^I \in [0, 1]$.

U's trust on E within a specific context c is thus represented by a single trust triple, as follows:

$$(t^c, d^c, m^c)_{U \to E} = (S^E, S^R, S^I) \times \begin{pmatrix} t^E & d^E & m^E \\ t^R & d^R & m^R \\ t^I & d^I & m^I \end{pmatrix} \tag{2.3}$$

where $t^c = S^E \times t^E + S^R \times t^R + S^I \times t^I$, $d^c = S^E \times d^E + S^R \times d^R + S^I \times d^I$ and$m^c = S^E \times m^E + S^R \times m^R + S^I \times m^I$

However the trust relationship should not only depend on the current values evaluated, it should also depend on the old values of trust. For example, if truster U completely trusts the trustee E then negative factors will be often overlooked when trust is re-evaluated. Therefore we present the final trust vector at time $\dot{t}$ as a linear combination of the previous time-dependent trust (t_i, d_i, m_i) and the trust evaluated at the present time (t^c, d^c, m^c). The weights assigned to old and current trust vectors is a matter of a user's trust evaluation policy.

Definition 4: To evaluate the final trust vector the relative weight α is assigned to the trust obtained at the present time and $1 - \alpha$ to the previous time-dependent trust vector, where $\alpha \in [0, 1]$.

Thus the final trust evaluated between a truster U and trustee E at time $\dot{t}$ in a particular context c is defined as:

$$\begin{aligned}(U \xrightarrow{c} E)_i &= \alpha \times (t_i, d_i, U_i) + (1-\alpha) \times (t^c, d^c, m^c)) \\ &= ({}_U t^c_E, {}_U d^c_E, {}_U m^c_E)\end{aligned} \tag{2.4}$$

where ${}_U t^c_E = \alpha \times t_i + (1-\alpha) \times t^c$, ${}_U d^c_E = \alpha \times d_i + (1-\alpha) \times d^c$ and ${}_U m^c_E = \alpha \times m_i + (1-\alpha) \times m^c$

2.4 Trust Evaluation Parameters

In this section, we formally define the three parameters, experience, reputation and identification, along with their respective evaluation. The Br2Br framework is easily extend-able for the inclusion of other parameters, such as Knowledge.

2.4.1 Experience

The experience parameter is based on the past performance of the trustee in the given context [TACM12]. In our trust model the performance is evaluated based on the behavior of trustee. We consider four types of behaviors encountered by the truster: good, bad, neutral and undisclosed.

Definition 5: Experience parameter (t^E, d^E, m^E) is defined as the computation of the aggregate performance of a trustee based on its behavior detected in a particular context over a specified period of time .

We model experience in terms of the number of behaviors encountered by a truster in a context over n subintervals of time period $[t_0, t_n]$. Let G_k, B_k, N_k, U_k be set of all good, bad, neutral and undisclosed behaviors that occur in the k^{th} interval $[t_{k-1}, t_k]$ of the time period. The experience acquired in the k^{th} interval is represented by (t_k, d_k, m_k) and evaluated as follows:

$$t_k = \frac{|G_k| + |\frac{N_k}{2}|}{|G_k| + |B_k| + |N_k| + |U_k|}$$

$$d_k = \frac{|B_k| + |\frac{N_k}{2}|}{|G_k| + |B_k| + |N_k| + |U_k|}$$

$$m_k = \begin{cases} 1 & if G_k = B_k = N_k = U_k = 0 \\ \frac{|U_k|}{|G_k|+|B_k|+|N_k|+|U_k|} & otherwise \end{cases} \quad (2.5)$$

The intuition behind the evaluation of experience is that each good, bad and undisclosed behavior contributes to the trust, distrust and mistrust components respectively by a factor of $\frac{1}{|G_k|+|B_k|+|N_k|+|U_k|}$, whereas, the neutral behavior contributes to both trust and distrust components by a factor of $\frac{0.5}{|G_k|+|B_k|+|N_k|+|U_k|}$. However, if no behavior occurs in k^{th} time interval then the mistrust component is equal to 1 and $t_k = d_k = 0$.

Naturally, the behaviors that occur in the older intervals should be weighted less than the behaviors in recent intervals. Each interval $[t_{k-1}, t_k]$ is thus weighted based on its position. We use the position weight p_k for each interval calculated, using $p_k = \frac{k}{S}$ where $S = \frac{n(n+1)}{2}$ [RC04]. Therefore the experience parameter is evaluated as $t^E = \sum_{i=1}^{n} p_k \times t_k$, $d^E = \sum_{i=1}^{n} p_k \times d_k$ and $m^E = \sum_{i=1}^{n} p_k \times m_k$.

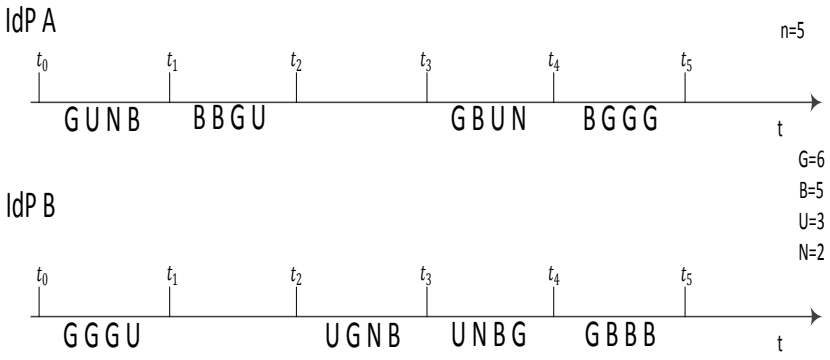

Figure 2.4: Set of Behaviors

Example 2: Bob uses the services of two different IdP's to authenticate himself over various web calling sites. To put trust in IdP he only considers the experience parameter. Figure 2.4 shows the set of IdP A and IdP B behaviors that Bob has experienced over a time period $[t_0, t_5]$ where $n = 5$. The position weights assigned to each interval are $p_1 = \frac{1}{15}, p_2 = \frac{2}{15}, p_3 = \frac{3}{15}, p_4 = \frac{4}{15}, p_5 = \frac{5}{15}$. Both sets have the same number of good, bad, neutral and undetermined behavior. However, the trust triple for IdP A is (0.4,0.28,0.32) whereas for IdP B it is

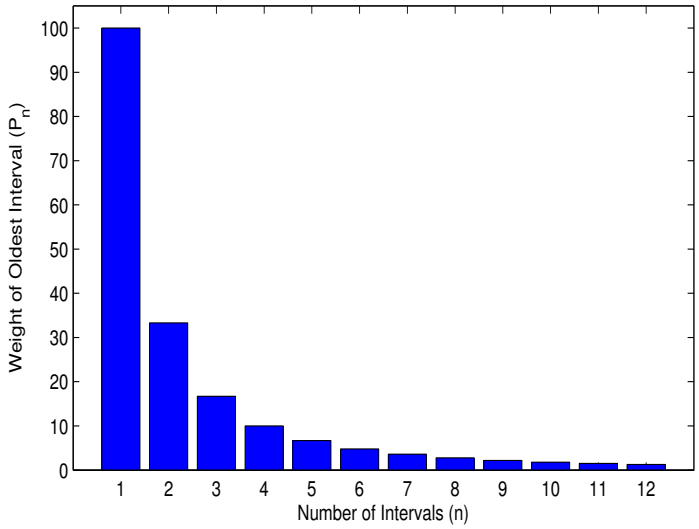

Figure 2.5: Oldest Interval Impact

(0.31,0.42,0.27). IdP A has a higher level of trust value only because it has more good behaviors that have occurred more recently than those of IdP B.

The experience parameter not only depends upon the weight-age of each interval but also on the total number of intervals n considered by the truster. Figure 2.5 represents the weight-age of the oldest interval with varying n. It can be observed that as the number of intervals increases, the weight of the oldest interval gets so small that it has no significant impact on the current value of trust. The selection of total number of intervals for computing the experience is again a matter of user's trust evaluation policy. Users may choose to forget the behaviors that are older than a particular amount of time. However, the decision should depend on the requirement of the accuracy of trust and the storage cost per interval.

2.4.2 Identification

The Identification parameter measures the amount of trust that a user can place in a digital identity received to authenticate the communicating participant. Several characteristics of the identity assertion are considered, wherein each characteristic consists of various identification levels. The identification levels are provided by the IdP during the iden-

tity verification process.

Definition 6: The identification parameter (t^I, d^I, m^I) determines the strength in the authentication process of the communicating participant. It is the aggregate of all satisfactory, unsatisfactory and unproven identification levels of the digital identity transaction weighted with the amount of trust in the IdP providing the authentication information.

The characteristics are represented by alphabets such as "X" and consists of various identification levels such as $X0, X1, X2, X3....etc$ further explained in Section 2.5. Each level is considered to be satisfactory, unsatisfactory or unproven attribute of the identity assertion. This categorization of identification levels are based on user trust evaluation policy.

Let IdP 'i' be the entity that provides the authentication information for the communicating participant p to user U. Where as Sat, $Unsat$ and $Unprov$ are the set of satisfactory, unsatisfactory and unproven identification levels considered by user U. Then the identity trust triple (t_p, d_p, m_p) for the communicating participant p is defined as the average aggregate of the number of satisfactory, unsatisfactory and unproven identification levels such that $t_p = \frac{|Sat|}{|Sat|+|Unsat|+|Unprov|}$, $d_p = \frac{|Unsat|}{|Sat|+|Unsat|+|Unprov|}$ and $m_p = \frac{|Unprov|}{|Sat|+|Unsat|+|Unprov|}$.

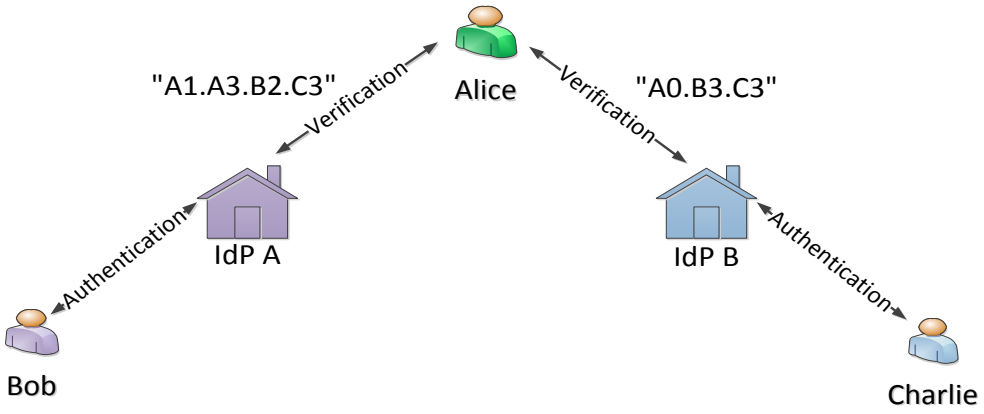

Figure 2.6: Identification Scenario

Example 3: Alice communicates with Bob and Charlie who are authenticated from IdP A and IdP B, respectively, as shown in Figure 2.6. Three characteristics "A", "B" and "C" are considered to evaluate the strength in the identity assertion. During identity verification, the IdPs provide the identification levels as "A1.A3.B2.C3" and "A0.B3.C3" for

Bob and Charlie respectively. The trust evaluation policy of Alice considers the sets $Sat = \{A2, B2, B3, C3\}$ $Unsat = \{A1, A3, B1, C1, C2\}$ and $Unprov = \{A0, B0, C0\}$. Using the formulas of t_p, d_p and m_p the identity trust triples for Bob and Charlie are calculated to be $(\frac{2}{4}, \frac{2}{4}, 0)$ and $(\frac{2}{3}, 0, \frac{1}{3})$ respectively.

However to evaluate the identification parameter the amount of trust in the IdP providing the authentication information should also be taken into account. Therefore if the trust triple between user U and IdP i is denoted by $({}_Ut_i, {}_Ud_i, {}_Um_i)$ then the identification trust triple (t^I, d^I, m^I) is evaluated by weighting it with the user's trust in the IdP :

$$\begin{aligned} t^I &= {}_Ut_i \times t_p \\ d^I &= {}_Ut_i \times d_p \\ m^I &= {}_Ud_i + {}_Um_i + {}_Ut_i \times m_p \end{aligned} \tag{2.6}$$

The intuition behind the weight-age assessment is that the user considers the authentication information trustworthy only if that user trusts the IdP otherwise it ignores the information making the mistrust factor of the identification parameter even higher.

Example 4: From the previous example it seems that Alice will put more trust in the authentication process of Charlie compared to that of Bob. However, this may not necessarily be the case. Let us suppose that the triple for Alice's trust in IdP A is $(0.9, 0.1, 0)$ whereas the triple for Alice's trust in IdP B is $(0.1, 0.7, 0.2)$. Using Equation 2.6, the identification parameters (t^I, d^I, m^I) for Bob and Charlie are evaluated to be $(0.45, 0.45, 0.1)$, and $(0.067, 0, 0.933)$, respectively making Bob's authentication more trustworthy. This is due to the fact that Alice ignores IdP B's authentication information about Charlie increasing the uncertainty in Charlie's identification.

Using the Electronic Authentication Guideline [BDP04] we present three characteristics of identity assertion: Identity Proofing, Credential Strength and Assertion Endurance, to estimate the trustworthiness of communicating participants. Each characteristic is further represented by different identification levels. These levels can be used to evaluate trust triple or can be indicated in plain text/symbols to the user.

Identity Proofing: This characteristic defines how strongly the set

of identity information representing a person has been verified by the IdP. This characteristic is represented by the following levels:

$P0$ No information about proofing is provided by the IdP;

$P1$ A pseudonymous identity is used;

$P2$ Identity information is self proclaimed;

$P3$ Identity information is proofed using social proofing;

$P4$ Identity information is proofed using signed/notarized documents;

$P5$ Identity information is proofed in person.

Credential Strength: This characteristic defines how strong user credentials are and how easily they can be spoofed or stolen. The characteristic is represented by the following levels:

$C0$ No information about credentials is provided by the IdP;

$C1$ No credentials are used;

$C2$ Credentials having user-name/password combination;

$C3$ Shared secret using symmetric key encryption;

$C4$ Cryptographic proof using asymmetric key;

$C5$ Hard tokens employed using trusted biometrics.

Assertion Endurance: This characteristic shows how well the identity assertion is protected against unauthorized access. The characteristic is represented by the following levels:

$S0$ No information about assertion is provided by the IdP;

$S1$ The identity assertion is neither protected nor signed;

$S2$ An access token is used to retrieve identity assertion;

$S3$ Identity assertion is signed and verifiable by the IdP;

$S4$ Identity assertion is encrypted;

$S5$ Identity assertion is audience protected.

Example 6: A bank provides remote financial assistance using WebRTC calling server. The bank requires customers to authenticate from a set of trusted IdPs. However, to provide security and confidentiality the bank representative limits financial information based on the strength of customers identification. Let's suppose $Customer1$ and $Customer2$ have identification levels as "$P5.C4.S3.S4.S5$" and "$P2.C2.S1$" respectively. Due to strong identification characteristics the bank representative allows $Customer1$ to receive sensitive information regarding personal account transactions. However, it restricts $Customer2$ to only obtain general information about bank services due to fragile identification.

2.4.3 Reputation

The reputation parameter aggregates the endorsements received about an entity from user's various communicating participants. An endorsement about an entity E is a trust triple $({}_pt_E, {}_pd_E, {}_pm_E)$ provided to the user by a communicating participant p. However, each endorsement should be weighted with the amount of trust in the communicating participant. Therefore we consider reputation to be collective measure of the endorsements from members of a particular community where each community is weighted according to the trust of the user in that community.

Definition 7: Reputation parameter (t^R, d^R, m^R) is the weighted aggregate of the average endorsements about a trustee received by each communicating participant of a particular community in a specific context.

We define 7 levels for endorsements in Table 2.1, where each endorsement level corresponds to a specific trust triple $({}_pt_E, {}_pd_E, {}_pm_E)$ provided to the user U by a communicating participant p about an entity E. However each participant belongs to a particular community of the user's contact list such as friends, classmates, relatives, co-workers etc. Therefore we consider the aggregate community trust triple $({}_ct_E, {}_cd_E, {}_cm_E)$ as the average of all endorsements received by the communicating participants of the community c such that ${}_ct_E = (\frac{\sum_{i=1}^{\bar{n}} e_i t_E}{\bar{n}})$, ${}_cd_E = (\frac{\sum_{i=1}^{\bar{n}} e_i d_E}{\bar{n}})$

Table 2.1: Endorsement levels

Endorsement Levels	Trust triple
Uncertain	$(0, 0, 1)$
Trusts absolutely	$(1, 0, 0)$
Trusts moderately	$(\frac{3}{4}, \frac{1}{4}, 0)$
Trusts neutrally	$(\frac{1}{2}, \frac{1}{2}, 0)$
Distrusts moderately	$(\frac{1}{4}, \frac{3}{4}, 0)$
Distrusts absolutely	$(0, 1, 0)$
No response	$(0, 0, 1)$

and ${}_cm_E = (\frac{\sum_{i=1}^{\bar{n}} e_i m_E}{\bar{n}})$ where $\bar{n}$ are the total number of endorsers in the community. However, each community trust triple should be weighted with the amount of user trust in that community.

Definition 8: Let $\hat{n}$ be the total number of communities set by the user, then the corresponding community weight vector is $(W_{c_1}, W_{c_2}, \cdots, W_{c_{\hat{n}}})$ such that $(W_{c_1} + W_{c_2} + \cdots + W_{c_{\hat{n}}}) = 1$ and $W_{c_1}, W_{c_2}, \cdots, W_{c_{\hat{n}}} \in [0, 1]$.

Therefore the community trust matrix consists of $\hat{n}$ rows where each row correspond to the trust triple of a particular community. The reputation parameter (t^R, d^R, m^R) is a multiplication of the community trust matrix and the corresponding community weight vector of the user:

$$(t^R, d^R, m^R) = (W_{c_1} W_{c_2} \cdots W_{c_{\hat{n}}}) \times \begin{pmatrix} {}_{c_1}t_E & {}_{c_1}d_E & {}_{c_1}m_E \\ {}_{c_2}t_E & {}_{c_2}d_E & {}_{c_2}m_E \\ \vdots & \vdots & \vdots \\ {}_{c_{\hat{n}}}t_E & {}_{c_{\hat{n}}}d_E & {}_{c_{\hat{n}}}m_E \end{pmatrix} \tag{2.7}$$

where

$$\begin{aligned} t^R &= W_{c_1} \times c_1 t_E + W_{c_2} \times c_2 t_E + \cdots + W_{c_n} \times c_n t_E \\ d^R &= W_{c_1} \times c_1 d_E + W_{c_2} \times c_2 d_E + \cdots + W_{c_n} \times c_n d_E \\ m^R &= W_{c_1} \times c_1 m_E + W_{c_2} \times c_2 m_E + \cdots + W_{c_n} \times c_n m_E \end{aligned}$$

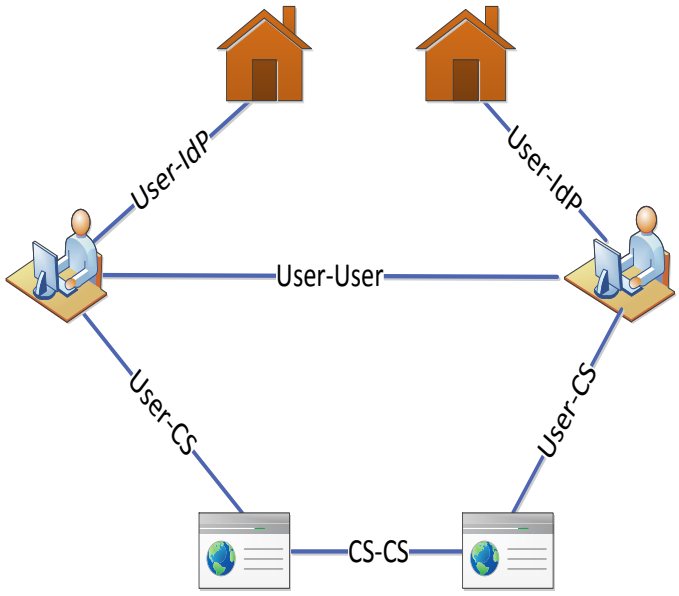

Figure 2.7: Trust Relationships in WebRTC

Example 5: Alice has set up two communities in her contact list $(family, friends)$. The corresponding community weight vector is $(0.8, 0.2)$. Alice usually avoids picking up calls that are not in her contact list, however before rejecting a call request she considers caller's reputation. If the trust value t^I of the caller is very high she accepts the call because that makes her feel that the person calling is very well known and trusted by her family members.

2.5 Trust Relationships

In Br2Br, each relationship is represented by a trust vector. We identify three trust relationships: User-User, User-CS and User-IdP as shown in Figure 2.7. The fourth trust relationship is CS-CS in case of inter-domain WebRTC call model. The experience, reputation and identification parameter or a combination of these parameters can be used to compute trust.

1. **User-CS**

$$(User \xrightarrow{c} CS)_i \tag{2.8}$$

Trust between user and CS is represented by Equation 2.8. CS provides JS application that allows browser to communicate in a P2P fashion. CS is also responsible for implementing signaling for the exchange of session parameters, identities, call answer/offer request and user reachable addresses between communicating parties. The purpose of trust in a User-CS relationship reflects the CS ability to provide communication services whereas CS aspects that needs to be considered are security and reliability. The trust context in User-CS relationship is the user's trust of a CS's security and reliability to provide communication services.

Utilizing the well established WebRTC security requirements [Res15] we provide a set of behaviors that should be considered to evaluate the experience parameter for web calling services.

- *Mixed Content:* In WebRTC, user interconnection with CS is considered to be secure if data is transferred over HTTPS [Res16]. However, the CS may produce mixed content during the duration of call by loading JS from an HTTP origin over its HTTPS page. The JS from HTTP might redirect media to location controlled by the attacker.
- *IdP Selection:* Current WebRTC specifications allow a CS to enforce the selection of a particular IdP. If the *setIdentityProvider* method has been called by the CS, then the user is bound to authenticate from a particular IdP [BBJ+16] set by the CS. This may lead to privacy and security concerns as a user may not trust the IdP to which it is forced to authenticate.
- *JS Client Load Time:* This indicates the time in seconds required to receive all the elements from the CS while loading the JS client. The user will only be able to place or receive calls from the browser on successful loading of JS client. The reliability of CS will depend on the time it takes for loading the JS client to be loaded on to the browser.
- *Response Waiting Delay:* This delay specifies the time in seconds spent by the browser waiting for a response message from the server. This depends un the processing time the CS requires for performing various tasks such as user discovery.

- *Malware Detection:* The security of the relationship will highly depend upon any malwares, errors, software vulnerabilities and undesirable software installations while running the JS client on the browser.

2. **User-IdP**

$$(User \xrightarrow{c} IdP)_t \tag{2.9}$$

The trust relationship between user and IdP is represented by Equation 2.9 IdP provides users the functionality of storing and managing their identity information while allowing them to authenticate themselves to their communication participants. The purpose of trust in User-IdP relationship is to trust an IdP's ability to provide authentication services while considering an its ability to preserve privacy.

Based on the additional trust requirements for IdP [BBC15b], we present set of essential behaviors that should be considered while evaluating the experience parameter for IdP.

- *Identity Encryption:* In WebRTC standard, the assertions are exchanged between the communicating parties via the CS. This allows CS to extract user identity information and track user activities [BBC14]. In order to have identity confidentiality from CS, the IdP must provide encrypted identity assertions.
- *Audience Protection:* During P2P authentication process of WebRTC, the IdP is unable to verify the party receiving the identity assertion. This allows any unauthorized party capturing the assertion to impersonate. Authentication protocols such as OIDC may be used which has the audience protection feature to verify that the authorized party is accessing the identity assertion [BdMM].
- *IdP Proxy Load Time:* This indicates the time in seconds required to receive all elements from the IdP web server while loading the IdP Proxy. Delay in loading IdP proxy will lag the authentication procedure required before establishing the connection.

- *Information Control:* This IdP feature allows a user to select the information presented in the identity assertion generated by the IdP. User can achieve confidentiality and enhance privacy by controlling the amount of information shared to their communicating participants in the identity assertions.
- *Authentication Delay:* This delay specifies the time required for an IdP to authenticate the user and generate identity assertion. The user requires to attach the identity assertion in order to initiate a call request.
- *Malware Detection:* Detection of any malwares, errors, software vulnerabilities and undesirable software installations while running the IdP proxy on the browser.

3. **User-User**

$$(User \xrightarrow{c} User)_i \tag{2.10}$$

Equation 2.10 represent trust between communicating participants. Before an exchange of real-time media, each user needs to verify the identity of its communicating participant. The purpose of trust relationship is to allows users to identify each other before establishing a communication session. Subjective user aspects are considered such as user's honesty, accuracy and integrity. The result of verification process is displayed to user which enables it to decide whether to accept or reject the call based on user verified identity.

For establishing trust between users the following should be considered

- *User Identification:* Before establishing communication session communicating participants should be able identify each other. Users needs to be certain that they are speaking to the person they believe that they are speaking to. Therefore, it would be desirable for a user to verify the identity of its communicating participant without relying on the service provider.
- *User Behavior:* Verifying user identity cannot guarantee the trustworthiness of communicating participant. Beyond sharing

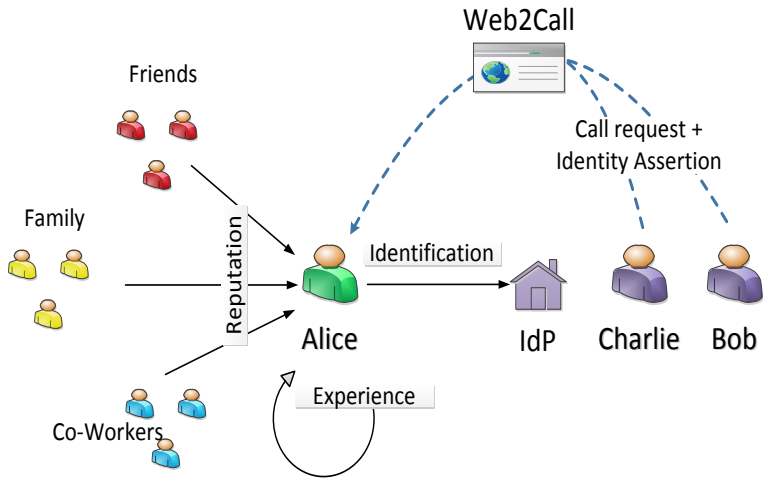

Figure 2.8: User Scenario Example

and verifying user identities the reputation of the caller should also be disseminated. User behavior in the network can be used to compute caller's reputation.

Table 2.2: Alice Trust Policy Defining Weight Scheme Vector

Truster	Trustee	S^E	S^R	S^I
Alice	User	0.1	0.4	0.5
Alice	CS	0.7	0.3	0
Alice	IdP	0	1	0

2.6 User Scenario

Br2Br manages the security in WebRTC calling services by evaluating trust for web browsers. The user scenario in Figure 2.8 illustrates how Br2Br framework helps in enhancing browser's security and user privacy. Bob and Charlie authenticate themselves to a particular IdP in order to initiate a call request to Alice via " Web2Call " calling service. Br2Br will allow Alice's browser to evaluate the amount of trust that can be invested in Bob and Charlie before accepting their call requests. The

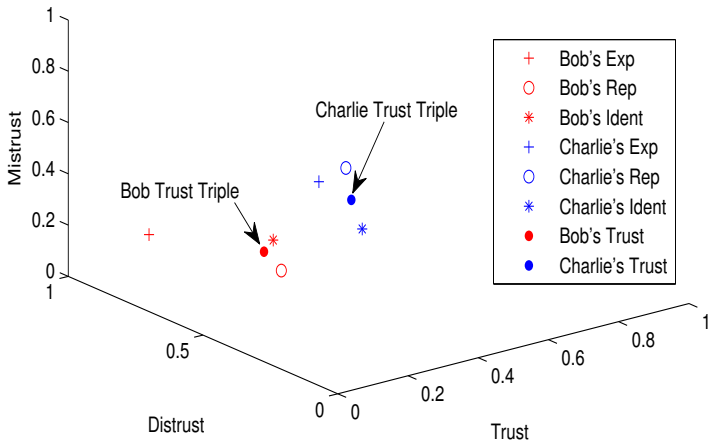

Figure 2.9: Bob and Charlie Trust Representation

trust triples for experience, reputation and identification parameters for Bob and Charlie are presented in the 3D plot of Figure 2.9.

Let's suppose Alice's browser blocks call requests from users having distrust value higher than 0.5. Using Alice's trust scheme vector in Table 2.2, the final trust vector evaluated for Bob and Charlie are $(0.22, 0.56, 0.22)$ and $(0.2, 0.21, 0.59)$ respectively. Both vectors have almost same trust values however, the browser blocks Bob call request whereas allows Charlie call request. This is due to the fact that the uncertainty factor for Charlie makes the distrust value lower than 0.5.

For the same user scenario, Figure 2.10 speculates the dynamic behavior of the CS " Web2Call ". Experience and reputation parameters are used to compute trust as per the trust policy in Table 2.2. Alice's browser by default terminates connection with any CS having trust levels below a particular threshold. At time $t = t_0$, Alice's browser detects mixed content and several attacks from the " Web2Call ". This type of behavior decreases the experience parameter which leads the trust value to fall below the threshold at time $t = t_1$. Therefore at t_1 browser will disconnect the services of " Web2Call " and display it to be unsafe for communication.

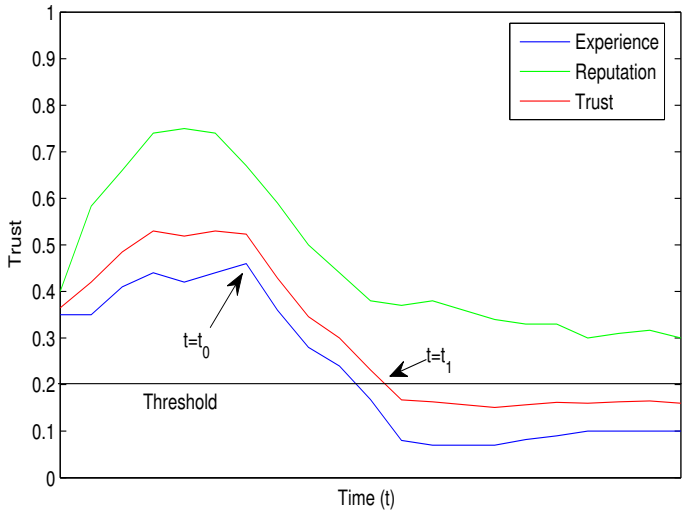

Figure 2.10: Dynamic Trust Evaluation for " Web2Call "

2.7 Conclusion

In this chapter, we have presented a new model for defining trust in WebRTC calling services. Our model formalizes the notion of trust, distrust and mistrust and presents three trust vector representing User-CS, User-IdP and User-User relationship in WebRTC. The framework uses three parameters namely experience, recommendation and identification to evaluate each trust vector. We propose expressions for each parameter to formalize trust in WebRTC. In our model the dependence of trust on time, context and trust policy is taken into account. To the best of our knowledge, this model is the first where formal definition of trust is presented for WebRTC security architecture.

Chapter 3

Identity Provisioning in WebRTC

" Your identity is your most valuable possession. Protect it. "
Elastigirl

Contents

3.1 Introduction

WebRTC allows communicating participants to validate each other's identity independent of their service provider [Res16]. Identity provisioning allows the generation, exchange and verification of identity assertions between communicating participants. This process is facilitated by the use of third party IdPs [Lyn11]. Users having an account on reputable IdP can use that IdP to prove to other services that they are who they claim to be. RTCWEB working group [RTC] propose the use of SSO authentication protocols (OAuth [Har] and BrowserID [BA13]) for the purpose of identity provisioning. We observed that the selection of underlying authentication protocol will strongly affect the privacy of user identities. User privacy in WebRTC deals with user identities and their associated profile information.

In this chapter, we study the concept of identity provisioning in WebRTC call model. We propose the use of OIDC protocol for the exchange of identity assertions between communicating participants. We mapped OIDC protocol over the WebRTC call model and compared it with OAuth and BrowserID in terms of user privacy properties. We conclude that OIDC is a better candidate than OAuth and BrowserID. The feature of encryption and audience control protects unauthorized parties to obtain user identity information. It further gives user control over their identity information.

3.2 WebRTC Call Model

WebRTC identity architecture [Res16] aims to provide maximum amount of authentication with the minimum possible level of trust in web CS. WebRTC architecture can be categorized into two basic call models: simplistic and multi-domain call models [JB12].

Figure 3.1 represents the multi-domain WebRTC call model that allows users from two different websites to communicate with each other. The main challenge in this model is the discovery of users across different domains. This requires each CS to share availability status, identity information and reachable address of its subscribers to other domains. Several efforts have been made to achieve cross domain interoperabil-

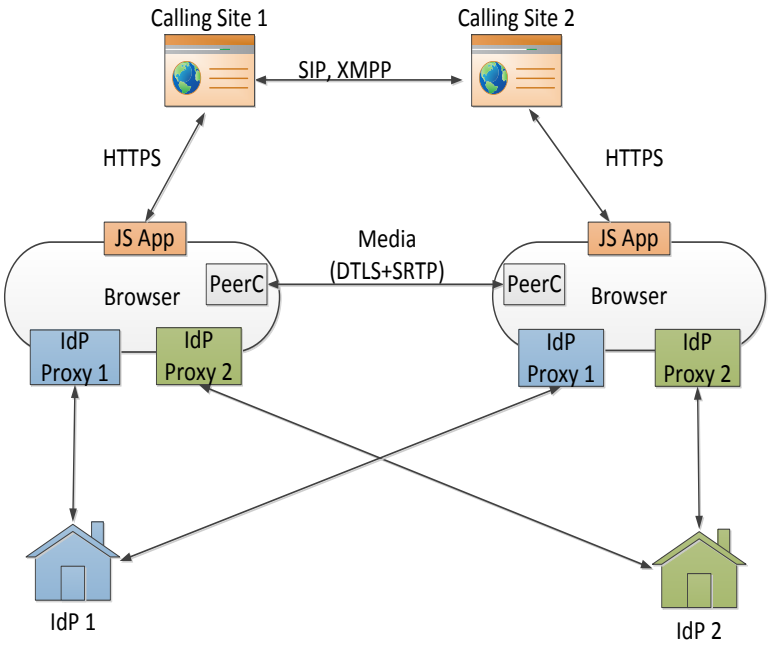

Figure 3.1: WebRTC Communication Model

ity [BBC+15a], [LCQC14], [JCC+16]. The second call model is the simplistic version where users communicate with each other using the same CS.

Irrespective of the call model, each CS is responsible for providing a JS application that operates over the browser and initiates PeerConnection component [BBJ+16]. By calling appropriate JS APIs PeerConnection (PeerC) establishes direct media connection between browsers as demonstrated in Figure 3.1. The P2P connection is established using DTLS extension to establish keys for SRTP [MR]. CS is also responsible for implementing signaling, Session Description Protocol (SDP) is used to exchange reachable addresses and session parameters. WebRTC does not mandate the use of any particular signaling mechanism to allow developers to implement their own choice of signaling method [JB12].

In order to authenticate user from IdP, PeerC downloads JS code " IdP proxy " from a specific location defined in the IdP domain. Browser is responsible for segregating JS codes into sandboxes where each script is only allowed to interact with resources from the same origin. Thus IdP proxy is only able to communicate to its IdP in order to authentication user. In response IdP generates user Identity Assertion (IA) which is in-

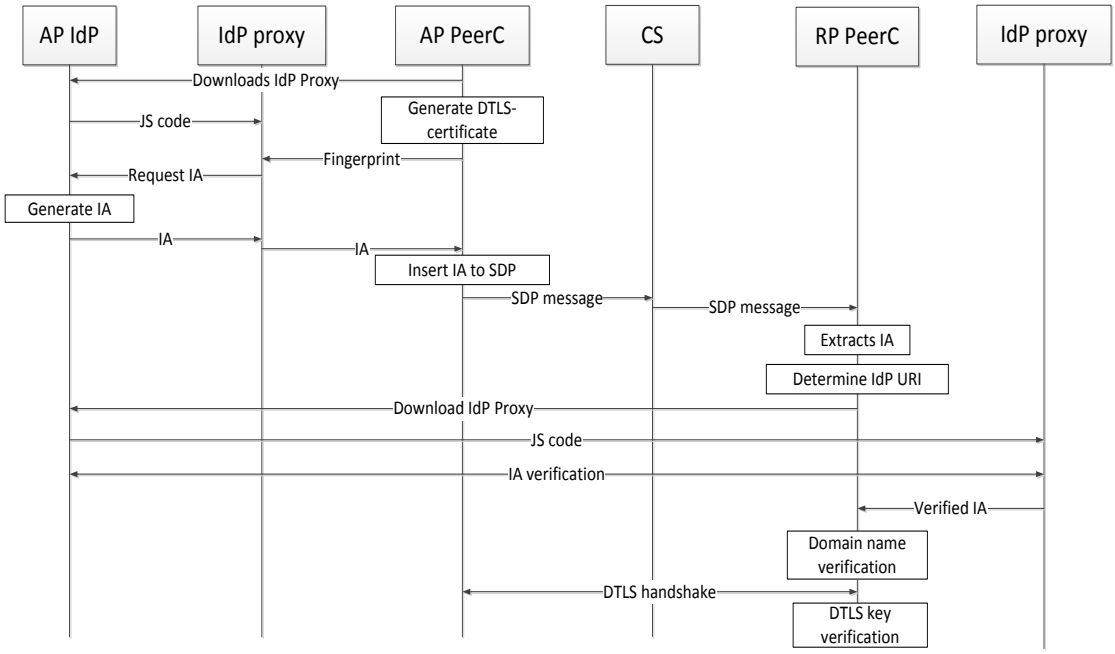

Figure 3.2: End-to-End Authentication Flow Diagram

cluded in the identity attribute of SDP descriptor message. The concept of IdP proxy allows browser to support any type of IdP or authentication protocol as long as it is able to download and run the JS code from IdP.

The browser that establishes user identity by authenticating itself with the IdP is Authenticating party (AP) whereas the browser which verifies the AP identity from the IdP is called the Relying Party (RP) [Res16]. In order for communicating parties to authenticate each other both browsers will act as an AP and as an RP in the process of end-to-end authentication.

3.3 Authentication in WebRTC

There are two types of authentication that apply to WebRTC communications. First is the authentication of the CS/IdP in which browser validates the ownership of origin. This is done by verifying the received digital certificate from the issuing certificate authority analogous to the authentication of any other website over Internet. The major drawback over here is that browser will trust any certificate that is validated by the trusted authority and has no means of verifying that the certificate

truly belongs to the owner. Identification of CS and IdP is very crucial in browser to browser communications as they are responsible for providing functionalities by running JS codes inside browsers. Thus for WebRTC efficient authentication mechanisms should be introduced that allow browsers to accurately verify that a digital certificate received is the correct certificate used by that website.

The second type of authentication is between communicating peers. User identity information in the form of IA are exchanged between peers via the CS and are verified from the same IdP that generated them [BB15]. There are two major drawbacks of WebRTC identity provision process. Firstly the IA are sent unencrypted which allows CS to extract user identity information and track user activities based on identities across communications. This shows that identity unlinkability from CS may never be achieved. Secondly the standard allows CS to force the selection of IdP which does not allow user to select its own choice of IdP. In order to use the services of CS user will have to authenticate to CS defined IdP which it may not trust.

The identity provision procedure presented in Figure 3.2 for end-to-end authentication in WebRTC is explained as follows:

Identity Assertion Generation AP PeerC generates request for assertion by attaching fingerprint of DTLS-SRTP certificate. The request also contains the origin of CS which allows IdP to be always aware of the CS user is using to communicate. IdP proxy is able to access user cookies which allows IdP to check whether user is already logged in or not. If user is not authenticated then the IdP proxy returns an error including the URL for entering user credentials. This error is handled by the JS Application or the CS as IdP proxy is sandboxed and cannot directly demand user to login. After successful authentication IdP generates and returns IA. The IA includes user identity information and DTLS fingerprint. The received IA is attached to the SDP message by PeerC and is sent to the remote party via CS as shown in Figure 3.2.

Identity Assertion Verification The RP PeerC extracts the IA from received SDP message. The domain name of IdP from IA is used to construct URL in order to downland the IdP proxy. For identity verification user is not required to authenticate itself. Upon successful verification the verified IA is returned. PeerC verifies IdP by comparing

name-space of received identifier in IA with domain name of IdP. In case of non-authoritative IdP where the name-space of identifier is not same as domain name, the IdP should be explicitly configured as trusted in browser. Before establishing connection PeerC matches fingerprint in IA with DTLS certificate received over the media channel. This is to ensure that the party establishing peer connection is same as the one which provided the IA.

3.3.1 Requirements for User Identification

We derive new set of trust requirements based on the weaknesses of identity architecture identified in previous section. These requirements address the privacy, security and trust concerns raised during end-to-authentication. In order to fulfill these requirements new solutions/modifications to the architecture and procedures of WebRTC should be proposed.

1. ***Identity Unlinkability:*** IdP needs to be able to provide user identity confidentiality against CS.
2. ***Identity Encryption:*** IdP needs to be able to provide encryption to user IA.
3. ***IdP Selection:*** User needs to select its own choice of IdP without being forced by CS.
4. ***CS Unlinkability:*** User needs to be able to hide the information about origin of CS from IdP.
5. ***Identity Information Control:*** User needs to be able to select the identity information included in IA.
6. ***Certificate Verification:*** User needs to be able to verify that the digital certificate provided by CS for authentication is the correct certificate used by it.
7. ***User Anonymity by IdP:*** User needs to be able to acquire anonymous identity from IdP.
8. ***Privacy awareness:*** IdP needs to inform user about how privacy will be handled during P2P authentication.

3.4 SSO Authentication protocols

WebRTC proposes the use of existing SSO [PM03] protocols for end-to-end authentication. SSO systems are designed for client server login where IdP allows applications to access user authentication information. As these protocols are not particularly designed for P2P authentication implementing them for WebRTC identity provisioning may require certain modifications. RTCWEB working group [RTC] propose the use of BrowserID and OAuth2.0 for identity provisioning. In order to map these protocols over the WebRTC security architecture we firstly need to study their architecture. In this section we detail the mechanism of OAuth and BrowserId. We also present OIDC protocol which constitutes a set of extensions on top of OAuth.

3.4.1 BrowserID

BrowserID allows any website to receive assertion of email address ownership from the user [FKS14]. The website is the RP whereas the browser is considered to be the client. In BrowserID specifications [BA13] client send Backed Identity Assertion (BIA) to the RP. BIA is combination of User Certificate (UC) and IA. UC carries user email address and user public key which is digitally signed by the IdP to certify the ownership of email address and public key of the user. Whereas IA contains the request to login into specific RP is signed by the user private key. As shown in Figure 3.3 the authentication procedure can be divided into three distinct processes as shown below:

- User certificate provisioning (steps 1-4): Client generates asymmetric cryptographic key pair and sends the public key along with the user email address to IdP. Upon user valid authentication the IdP generates UC by signing user email address and public key.
- Assertion generation (steps 5-6). Client generates IA that containing the origin of RP signed by user private key and sends BIA to RP which contains IA and UC.
- Assertion verification (step 7-8): The RP validates signature in the IA with public key inside UC to ensure that the user requesting

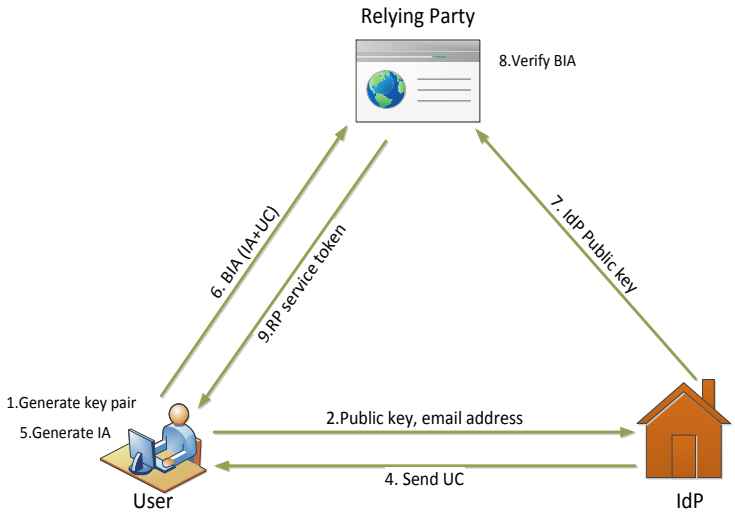

Figure 3.3: BrowserID Authentication Overview

to login is the same that requested UC from the IdP. RP requests IdP for the public key and matches it with the signature inside UC. Upon successful verification RP allows client to use its services.

3.4.2 OAuth

OAuth 2.0 [Har] being an authorization protocol allows client applications to access resource hosted on a Resource Server (RS) owned by Resource Owner (RO) as shown in Figure 3.4. The authorization to access resource is provided by the Authorization Server (AS) on behalf of the RO. However before accessing the resource client has to register with AS using the client-id [Lei12]. The process of authorization in Figure 3.4 is described in brief as follows:

- Steps 1-3: RO accesses the client application which redirects it to the AS to authenticate itself
- Steps 4-5: After successful authentication it is redirected back to the client application with the authorization code.
- Steps 6-7: The client receives access token by providing the authorization code to the AS. Before receiving access token the client has to verify itself by providing the client id and client secret.

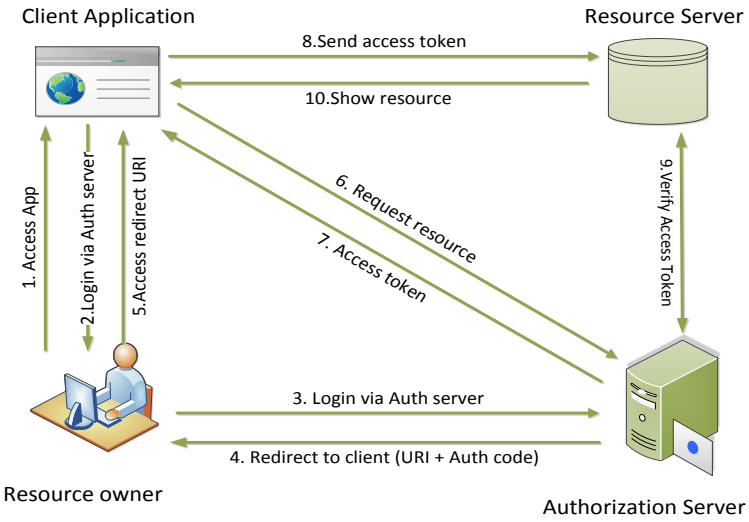

Figure 3.4: OAuth Authorization Overview

- Steps 8-10: The access token is used by the client to request resource from the RS. The RS grants access to the resource by verifying the access token from the AS.

3.4.3 OIDC

OIDC adds an identity layer on top of the OAuth 2.0 protocol. It enables client to verify the identity of user based on the ID Token [BdMM] that contains claims about the user authentication. The ID Token incorporates user (AP) identity information, the IdP identifier and the audience (RP) for which the token is intended for. The ID token is signed by the IdP and can optionally be encrypted. The authentication procedure for OIDC implicit flow in Figure 3.5 is described as follows:

- Step 1-2 The client application requires user to authenticate and sends authentication request containing the desired request parameters to the IdP.
- Step 3-4 The IdP requires user to authenticate and sends back an ID Token upon successful authentication.
- Step 4-5 The user forwards the ID Token to the client which allows it to retrieve information about the user after validating the ID Token.

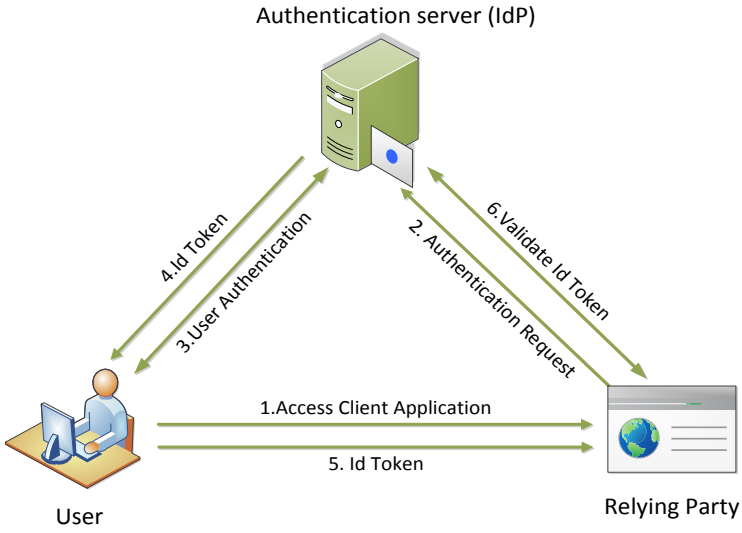

Figure 3.5: OIDC Authentication Overview

3.5 Identity Provisioning using SSO Protocols

In this section we map the previously introduced SSO protocols over the WebRTC call model. We demonstrate how these protocols designed for user multiple login purposes can be used for the exchange of identity assertions between peers. Although the use of IdP proxy makes WebRTC to be protocol independent, the selection of a particular authentication protocol will profoundly affect the security and privacy of user identities. User privacy involves the protection of user identities and their profile information. To ensure user privacy we propose the use of OIDC for identity provisioning. In order to prove it is better candidate than BrowserID and OAuth we compare the three protocols in terms of user privacy properties.

3.5.1 BrowserID

BrowserID is considered to be a browser centric protocol that relies on the browser to provide the final identity assertion. This makes it consistent with the identity provision architecture. However when applied to WebRTC instead of the website browsers will authenticate each other. BrowserID can be mapped to WebRTC architecture as follows:

Identity Assertion Generation A public private key pair is generated by the AP browser for asymmetric encryption. PeerC downloads the IdP proxy and requests IdP to generate UC by including the user public key. After valid authentication of user the IdP generates UC by signing the user identity (public email address) and public key. The PeerC generates the DTLS-SRTP key for establishing the media connection and sends the fingerprint to IdP proxy. The IA is generated by IdP proxy by signing the fingerprint with user private key. It should be noted that in BrowserID browser is not required to send the fingerprint to IdP as it generates the final assertion. Lastly PeerC generates the final assertion BIA and includes it into the identity attribute of the SDP.

Identity Assertion Verification The RP PeerC receives the SDP message and extracts IA and UC. The domain name is used to download IdP proxy to request public key from the IdP. PeerC performs two checks. First it matches the received public key with the signature inside UC, secondly it verifies user public key inside UC with the signature inside IA.

3.5.2 OAuth

To map OAuth protocol onto WebRTC architecture the client application can be considered as RP browser, the RO as the AP browser whereas the IdP acts as AS as well as RS. For WebRTC the AP has to register its identity and the fingerprint as an identity resource at the IdP. Instead of the assertion the AP provides authorization code to the RP though CS which allows the RP to access the identity resource from the IdP. OAuth protocol is particularly designed for authorization but may be applied for P2P authentication as follows:

Identity Assertion Generation The AP PeerC using the IdP proxy authenticates the user to the IdP and registers an identity resource with IdP including the fingerprint of DTLS certificate. The IdP in return sends the IA which contains the authorization code. PeerC attaches the authorization code to the SDP and sends it to RP browser via the CS.

Table 1: Comparison of Authentication Protocols for WebRTC

Authentication Protocols	Identity Verification	Anonymity	Unlinkability from CS	Unlinkability from IdP	Pseudonymity	Identity Encryption	Browser Centric	Information Control	Audience Verification	IdP Centric
BrowserID	✓			✓			✓			
OAuth2.0	✓	✓		✓	✓					✓
OIDC	✓	✓	✓	✓	✓	✓		✓	✓	✓

Identity Assertion Verification The RP PeerC receives the SDP message and extracts the authorization code from the identity attribute of SDP. The IdP proxy sends the authorization code and receives access token from the IdP. Upon receiving the access token the IdP proxy retrieves the identity and fingerprint. The RP PeerC verifies the fingerprint with DTLS certificate received over the media channel.

3.5.3 OIDC

The use of ID Token as IA makes OIDC more complaint with the browser centric approach of WebRTC as compared to OAuth. The OIDC protocol can be applied to WebRTC identity architecture as follows:

Identity Assertion Generation AP PeerC sends authentication request to IdP containing fingerprint and the audience (RP identity) to which the ID Token is intended for. The request may also indicate the type of identity information to be returned in the ID Token. The ID Token is generated and signed by the IdP which contains AP identity information, fingerprint, RP identity and the IdP identifier. PeerC includes the ID Token to the SDP and sends it to the RP.

Identity Assertion Verification The RP PeerC receives the SDP message and downloads IdP proxy by using identifier domain name. It also extracts the ID Token and fingerprint from the identity attribute. The IdP proxy requests the IdP to validate the ID Token. IdP verifying the signature and returns the verified Identity to the RP. The RP PeerC then verifies the fingerprint with DTLS certificate received over the media channel.

3.5.4 Comparison

Privacy is the individual interest in sustaining a personal space, free from interference by other people and organizations [Cla13]. Privacy protection can be defined as user control about what, when and with whom its information is shared. In WebRTC user privacy mainly deals with protection of user identity and associated profile information. Table 1 provides a quick comparison of authentication protocols in terms of user privacy properties [PT10] and features defined as follows:

1. *Identity Verification*: User ability to verify the identity of remote party.

2. *Anonymity*: The inability of remote party and CS to learn user identity.

3. *Unlinkability from CS*: The inability of CS to track user activities based on user identities.

4. *Unlinkability from IdP*: The inability of IdP to track user activities across different CS.

5. *Pseudonymity*: The ability of IdP to provide user with pseudonyms as anonymous identities.

6. *Identity Encryption*: The ability of IdP to encrypt user identity to achieve confidentiality from CP.

7. *Browser Centric*: The ability of browser to generate the identity assertion

8. *Information Control*: The ability of user to control the type of information IdP includes in the IA.

9. *Audience Verification*: The ability of IdP to disclose identity information exclusively to the person which it was intended for.

10. *IdP Centric*: The ability of user to enforce rules and policies through a trustworthy IdP.

Browser centric approach of BrowserID makes it the simplest protocol that can be applied to WebRTC architecture for identity provisioning. When compared with OAuth and OIDC the considerable drawback of this protocol is the adoption of public email address as user identity. Firstly it does not allow user to stay anonymous/unidentifiable during the communication. Secondly unlinkability from CS can never be achieved as public email address will always allow it track user activities. When having BrowserID for authentication users will have to trust their CS with their identity information.

However the fact that final IA is generated by the browser without the need of sending DTLS fingerprints to IdP makes BrowserID more reliable in case of distrusted IdP. In contrast to BrowserID protocol, OAuth and OIDC operate in an IdP centric format where IdP is responsible for generating and verifying the IAs. IdP centric nature will allow users to enforce policies and rules through their IdPs. Other than this Anonymity in both these protocols can easily be achieved by the user of pseudonyms.

In OAuth protocol redirection between browsers is impossible to achieve as browsers do not have the capability to accept HTTP connection from other browsers. Thus when using OAuth for P2P authentication AP browser is never aware of who is accessing its identity resource whereas IdP is unable to verify that RP has the authority to access AP identity. This brings about serious security concerns for WebRTC communication as any unauthorized party having access to authorization code will be able to obtain user identity information.

OIDC seems to be a better candidate than OAuth in terms of identity confidentially and unlinkability. The feature of encryption and audience in ID Token does not allow any unauthorized party such as Man-in the middle or CS to obtain user identity information. The audience field in OIDC allows the AP to specify the identity of RP to which the information is intended for. This requires AP to be aware of RP identity before P2P authentication which may be communicated though the CS. Lastly OIDC gives user much more control over their identity information to be shared by indicating it in the authentication request.

3.6 Conclusion

In this chapter, we study the WebRTC call model to define the process of identity provisioning. We map SSO authentication protocols over WebRTC architecture to allows communicating participants to authenticate each other in a P2P fashion. WebRTC is considered to be very secure however it has not considered user privacy. We identify different privacy properties in relation to user identities. We compare the three authentication protocols in terms of privacy properties. We conclude that OIDC is the best protocol in terms of user privacy.

Chapter 4

Trust Computational Model

" Trusting is hard. Knowing who to trust, even harder ".
Maria V. Snyder

Contents

4.1 Introduction

Web-based communication services are exposed to several threats in which the social context between communicating participants is manipulated. Cybercrimes based on identity misrepresentation to obtain sensitive information are on the rise. Various scams and frauds are conducted by distributing malicious content, viruses and Spam over web communication services. User identification is the first step on recognizing the communication participant. However user authentication over web is not sufficient to protect communication services. Authentication remains a static parameter which does not show the overall behavior of the user. In order to protect communication services from social security threats, methods of estimating trustworthiness and reputation should be built into web calling services.

In this chapter, we present a detailed description of the potential social security threats that exist in real-time web communication services. We propose a novel reputation-based trust model "*TrustCall*" to estimate the trustworthiness of communicating participants. The computed trust is used to differentiate between legitimate and malicious callers over web communication services. *TrustCall* is a hybrid trust computational model based on the evaluation of *Authenticity Trust* and *Behavioral Trust*. *Authenticity Trust* describes the legitimacy and genuineness of a caller's identity whereas *Behavioral Trust* determines the popularity and acceptance of a user in the network. *Authenticity Trust* is based on recommendations received by other members of the network, while *Behavioral Trust* is computed by examining the communication behavior of the user. The feasibility and effectiveness of the model is shown using a simulated network of communicating peers.

4.2 Related Work

This section provides a comprehensive literature review of the WebRTC standard, as well as an analysis of the existing trust computational models.

The identity specifications of WebRTC are highly flexible when compared to the closed ecosystems of existing VOIP solutions. The WebRTC

architecture [Res16] allows communicating participants to identify each other before establishing a communication session [Res15]. Each user verifies the authenticity of a communicating participant's identity independent of the service provider [LR12]. Different models for provisioning user identity in an end-to-end manner are defined in [BBC14]. Authors in [DGSJ+16] have proposed several mitigating techniques and security improvements for WebRTC identity specifications, and new requirements for WebRTC identity architecture are highlighted in [BBC15b] and [CCFJ16].

A considerable amount of literature has been published on identifying and authenticating users over WebRTC-enabled services. For instance, the researchers in [LFGG+14] provide authorization models based on access control lists and capability-based security. A novel identity mapping and discovery system based on DHT-based directory service is proposed in [IRCaK+17]. This system enables users of web-based communication applications to discover and authenticate other users in the network. In [LCQC14], a mirror-presence mechanism is used to locate, identify and authenticate users on web calling services. While all these solutions facilitate user authentication and identification in WebRTC, they do not provide a method that ensures the legitimacy of users. Therefore, new mechanisms are still needed to screen and scrutinize callers over web communication services.

In order to anticipate user behavior, reputation-based techniques are one the most practical and effective solutions used over the Internet. Most of the reputation-based models used over P2P networks leverage on the collection of recommendations about the trustworthiness of a peer by other members of the network. The most popular reputation-based trust models for P2P networks include EigenTrust [KSGM03], PeerTrust [XL04] and PowerTrust [ZH07]. A comprehensive comparison of these models is provided in [MP09b]. EigenTrust is one of the most well-known and cited trust model for P2P file sharing networks. PowerTrust is considered as an enhancement of EigenTrust. However, these models are based on the assumption of some pre-trusted peers in the network. In contrast, PeerTrust is a very simplistic model that determines peer's trustworthiness by taking into account several important factors such as feedback source credibility, transaction context and community context.

On the other hand, reputation over on-line social networks is based on user interactions. Interaction-based trust models are usually applied to networks where the size, frequency and type of interaction are important indicators of trust. This is evident in the case of STrust [NSP11], where trust is evaluated based on the popularity and engagement of users in social networks. A novel behavior-based trust model for on-line social networks is presented in [AEG+10]. These models completely ignore the structure of networks that provide important information about how members in a community relate to each other. In contrast, network-based trust models exploit the propagative nature of trust in the network to determine the trust between any two nodes. TidalTrust [Gol05a] provides a good illustration on how the network structure can be used to establish trust between peers having no direct connection.

In WebRTC, the CSP facilitates direct connection between communicating participants. However, the WebRTC based communication differs from P2P networks. P2P systems are usually deployed using a distributed hash table that allow peers to efficiently search the network for a resource. On the other hand, WebRTC standard requires a central server to discover and locate peers in order establish a communication session between them [Res16]. With WebRTC services, the centralized functionality of a server is used to maintain decentralized clusters of peers. Therefore, trust in WebRTC services needs to be computed in a centralized manner. The call graphs in communication services show how peers relate to each other. The frequency, duration and nature of their calls are important behavioral indicators that can be used to estimate their trustworthiness [COBY11]. We choose to explore this area of research to present the firs hybrid trust model for real-time web communication services.

4.3 Threat Taxonomy

In this section, we detail the threat taxonomy for real-time web conversation services, with a focus on social threats that directly target users.

The security threats in web communications are categorized into: confidentiality, integrity and social threats. Potential threats against user confidentiality includes unauthorized means of capturing information

such as voice, data, identities, credentials and call patterns. Threats against integrity involve the alteration of signaling or media messages by intercepting them in the middle of the network. However, threats against social contexts are distinctive, as they are directly aimed against humans. In social threats, the context between communicating parties is manipulated in order to transfer false or malicious content to the target victim. Identity over web conversational services is commonly a combination of self-created user profiles and credentials. Therefore, an attacker can present fraudulent information, such as a false name, organization, email address, or presence information to misrepresent himself over the network. Identity misrepresentation over the telephone network facilitates various security threats.

We identify five social security threats that are present over web conversational services:

1. ***Phishing:*** Phishing is an illegal attempt to obtain some one's confidential information such as their identity, password, bank account number, credit card information etc. During a phone call, the attacker usually pretends to be from a trustworthy organization (such as a reputed bank or recognized office) in order to trick their victims in revealing private and confidential information.

2. ***Spam:*** Spam over Internet Telephony (SPIT) or robocalls are automatically dialed unsolicited pre-recorded bulk phone calls that are broad-casted for marketing purposes. SPIT are much more disruptive than other kinds of spam, as they require immediate response from the recipient. The low cost and open nature of Internet telephony provides an attractive medium for attackers to generate Spam.

3. ***Undesired Content Distribution***: Communication services are used to distribute corrupted or virus-infected files such as spy-wares, viruses, Trojans, malwares etc. Illegal and unlawful content may also be distributed such as sexually explicit images, content promoting crime or violence, copyright violation and illegal trading. This type of content can also be used to deliver false or misleading information, which can in turn be used for phishing attacks.

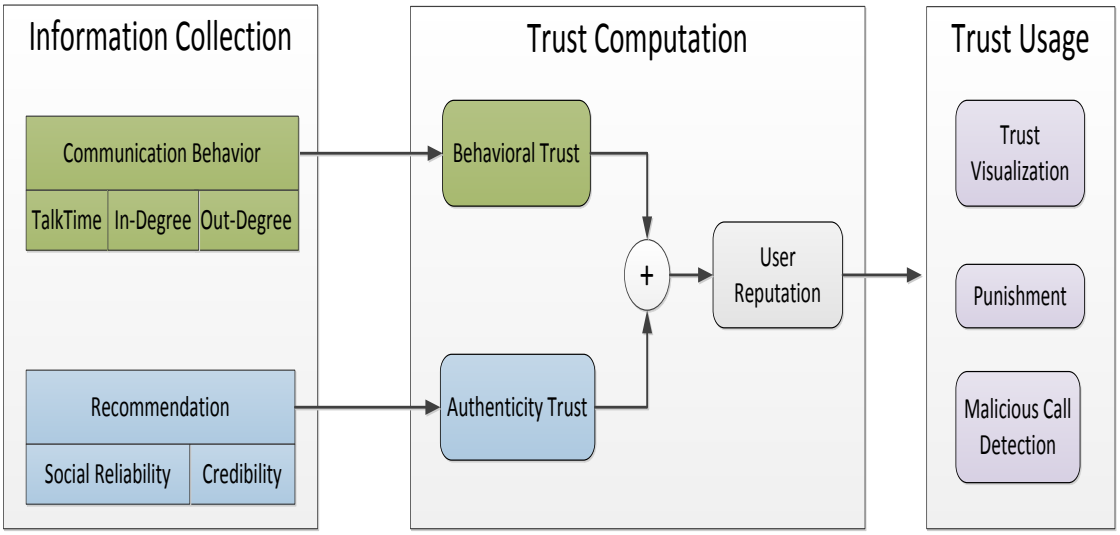

Figure 4.1: TrustCall Architecture

4. ***Nuisance Marketing***: Telemarketers use high pressure sales techniques over communication services to pursue customers in buying their products over phone calls. These advertising tactics are considered to be unethical. Telemarketers have also been involved in various frauds and scams while selling products over the phone.

5. ***Unwanted Contact***: Unwanted communication includes acts of harassment, extortion, blackmail and abuse that are all against the law. While a communication system may not be able to detect unwanted ot undesired communication, it may be able to detect the users involved in such activities.

4.4 TrustCall Model

In this section, we present '*TrustCall*', the first hybrid trust computational model that evaluates trust between communicating participants.

We define trust between communicating peers as the belief that they will act legitimately and securely over the communication session. Trust is dynamic in nature as it increases with positive experiences and decreases with negative ones. Trust should therefore be modeled with respect to time and expressed in a continuous variable. Since older experiences might become irrelevant with time, recent experiences are more

important in determining trust. We consider $\hat{T}$ the time period over which communication occurs. The time period is further divided into n intervals, $[t_0, t_1], [t_1, t_2], ..., [t_{n-1}, t_n]$. A particular interval $[t_{k-1}, t_k]$, is referred to as the k^{th} interval, where for any interval $[t_i, t_j], t_i < t_j$. Experiences that occur within the specific time period are weighted based on their positioning in time. Experiences outside the time period are ignored. The computed trust for a communicating peer p_i is denoted as:

$$Tr_t(p_i) \tag{4.1}$$

Figure 4.1 illustrates the *TrustCall* architecture comprised of three components: information collection, trust computation and trust dissemination.

Information Collection: The information regarding user reputation is collected from two sources, (i) recommendations and (ii) user's communication behavior. Recommendations regarding user legitimacy are collected from other members of the network based on their experience. Each recommendation is selected and weighted based on the member's trustworthiness in give correct recommendations. In order to determine whether a communicating peer is worthy enough to be accepted as recommender, we introduce the social reliability parameter. Whereas, credibility parameter represents the sincerity of a recommender in giving correct recommendations. For each recommender facts about its social reliability and credibility are collected. On the other hand, a user's communication behavior is observed in order to determine their popularity and acceptance in the network. Three attributes of call graphs (incoming calls, outgoing calls and talk time) are observed to describe the behavior of a user in a communication network.

Trust Computation: *TrustCall* is based on the evaluation of two types of trust: *Authenticity Trust* and *Behavioral Trust*. *Authenticity Trust* is used to describe the legitimacy of a communicating peer's identity. *Behavioral Trust* is computed to determine a user's acceptance and recognition by other members of the network. *TrustCall* is a hybrid trust model in which trust is expressed as a linear weighted sum of *Behavioral Trust* and *Authenticity Trust*. Each type of trust owns a weight that indicates its influence over the computed trust. The computed trust has a value between -1 and $+1$. This facilitates illustrating the amount of

trust as well as distrust associated with any peer. The computed trust $Tr_t(p_i)$ for a peer p_i can be expressed as follows:

$$Tr_t(p_i) = \alpha \times Tr_{Auth}(p_i) + (1 - \alpha) \times Tr_{Beh}(p_i) \tag{4.2}$$

where $Tr_{Auth}(p_i)$ is the *Authenticity Trust* and $Tr_{Beh}(p_i)$ is the *Behavioral Trust*. α is the weight that ranges from $[0, 1]$. Quantifying the influence of each type of trust depends upon its usage in web communication networks.

Trust Usage: *TrustCall* allows CSPs to introduce the feature of trust visualization. Trust visualization is the presentation of trust in a pictorial, graphical or textual format. The visualization of a caller's reputation can be used to advise and assist whether and how much a particular peer can be trusted over the communication network. Any user will be able to visualize the computed trust of other members of the network before initiating or accepting a call request. This will help subscribers to identify and communicate with legitimate callers. However, the decision to accept or reject a call request is very personal and is left up to the user to decide. The evaluated trust can further be used to enhance confidentially and security over a communication session. For example, a user may limit the amount of information or refrain from accepting any image or file from callers that are doubtful and suspicious. This trust value can also be used to block call requests that originate from the least-trusted members of the network. In addition to trust visualization, *TrustCall* allows CSPs to detect malicious callers over their networks. CSP may punish malicious callers by blocking their calls or by banning them from the network.

In *TrustCall* the reputation of a user is based on the computation of *Authenticity Trust* and *Behavioral Trust* as shown in Figure 4.1. We formally define *Authenticity Trust* in Section 4.4.1 followed by *Behavioral Trust* in Section 4.4.2.

4.4.1 Authenticity Trust

Authenticity Trust describes the legitimacy and genuineness of user identity. *Authenticity Trust* of a peer is evaluated based on the recommendations received from its communicating participants. In *TrustCall* a

recommendation is bound to each call where both participants rate each other based on their experiences. If the communicating participant has a genuine identity, it is rated as legitimate. If the communicating participant uses a false identity to conduct malicious activities as described in Section 4.3, it is rated as malicious. Any peer p_j can rate its communicating participant p_i as follows:

$$Rec_{p_j \rightarrow p_i} = \begin{cases} +1 & \text{if legitimate} \\ -1 & \text{if malicious} \end{cases} \tag{4.3}$$

In traditional recommendation systems trust is commonly computed as the average aggregate of all recommendations received over a peer's communication lifespan. If n_{p_i} are the total number of communicating participants of peer p_i, then the *Conventional Trust* $Tr_{Conv}(p_i)$ can be computed as follows:

$$Tr_{Conv}(p_i) = \frac{\sum_{j=1}^{n_{p_i}} Rec_{p_j \rightarrow p_i}}{n_{p_i}} \tag{4.4}$$

Traditional recommendation systems are prone to several attacks and strategies used to unfairly enhance reputation [MGM06]. In Table 4.1, we summarize adversarial powers that are accessible to malicious peers. For instance, peers may behave as a traitor, give false recommendations, conduct a Sybil attack, form a malicious collective group, or simply shed their bad reputation by re-entering the network with a new identity. Malicious peers adopt such strategies to avoid their detection in recommendation systems. In *Conventional Trust* recent ratings play an insignificant role in altering a peer's trust value. Moreover, each recommendation is considered equally to evaluate trust for peer p_i. Therefore, malicious peers can easily deceive or mislead traditional recommendation systems.

In *Authenticity Trust* we introduce mechanisms to combat the attacks and strategies defined in Table 4.1. In order to capture peer's recent behavior, we weight recommendations based on their positioning in time. This method helps in detecting traitors presence in the network. We model peer's *Authenticity Trust* in terms of the number of recommendations received over n subintervals of a specified time period $\hat{T}$ (for

Table 4.1: Adversarial powers

Behaviour	**Description**
Traitors	Traitors are users that behave properly for a period of time to maintain a respectable reputation before behaving maliciously.
Sybil Attack	Multiple false identities are forged by a user in order to enhance its reputation.
False Rating	Users may provide false recommendations. Malicious peers are morelikely to provide false recommendations in order to hide their bad reputation.
Malicious Spies	Malicious spies are peers who behave legitimately in the network but give false rating to peers who behave maliciously.
Collusive Group	Peers in the network may form a collusive group in order to cooperatewith each other by providing false rating.
White Washing	Users shed their bad reputations by purposely the network with a new identity.

instance 3 weeks or 3 months). The *Authenticity Trust* at time t_i is represented as follows:

$$\frac{\sum_{k=1}^{n} w_k \sum_{j=1}^{n_{p_i}^k} Rec_{p_j \rightarrow p_i}^k}{\sum_{k=1}^{n} n_{p_i}^k} \tag{4.5}$$

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. Recommendations older than the specified time period are discarded. We use the position weight w_k defined in [RC04] for each interval, using $w_k = \frac{k}{S}$ where $S = \frac{n(n+1)}{2}$. The choice of time period $\hat{T}$ and number of intervals n is a matter of trust evaluation policy that is set by the CSP.

Furthermore, in *Authenticity Trust* we introduce two parameters i) social reliability and ii) credibility to choose and weight each recommendation received. Social reliability determines whether a communicating peer is worthy enough to be accepted as a recommender, while the cred-

ibility parameter represents the sincerity of a recommender in giving correct recommendations. In *Authenticity Trust* each recommendation received is weighted with both the social reliability and credibility parameters. The *Authenticity Trust* $Tr_{Auth_{t_i}}(p_i)$ of a peer p_i can be computed as follows:

$$Tr_{Auth_{t_i}}(p_i) = \frac{\sum_{k=1}^{n} w_k \sum_{j=1}^{n_{p_i}^k} Rec_{p_j \rightarrow p_i}^k \times Sr_{p_j} \times Cr(p_j)}{\sum_{k=1}^{n} n_{p_i}^k} \tag{4.6}$$

where Sr_p is the social reliability and $Cr(p_j)$ is the credibility of peer p_j.

Social reliability is a binary parameter that shows whether a user is reliable enough to be considered as a recommender. The recommendation from any peer is considered if its social reliability parameter is equal to 1. Social reliability is based on the number of interactions in the network. A peer's interaction rate is represented by the amount of calls made and received in the network. The thresholds can be set by examining the average number of incoming and outgoing calls in the network. Social reliability is introduced to detect fake profiles that are injected into the network. Fake profiles are highly unlikely to have a reasonable amount of interactions, as their sole purpose is to falsely recommend a particular peer. The social reliability parameter is defined as follows:

$$Sr_{p_j} = \begin{cases} 1 & \text{if } interactions \geq threshold \\ 0 & \text{otherwise} \end{cases} \tag{4.7}$$

On the other hand, the credibility parameter helps in determining the sincerity of a peer in giving correct recommendations. It is the weight given to the recommendation based on the user's sincerity. The credibility parameter has a value between 0 and 1. Credibility close to 1 shows that the peer is sincere in giving correct recommendations, while credibility close to 0 shows that the peer provides false recommendations. Therefore, the credibility parameter helps to detect users who provide false ratings. Furthermore, it also helps to counter collusive group formation by selecting peers that provide correct recommendations. To determine user credibility we introduce three metrics: i) reliability, ii) similarity and iii) honesty.

Reliability (R): Reliability is based on the assumption that legitimate peers are more likely to give correct recommendations whereas malicious users are more likely to give false recommendation. Therefore, in the reliability metric we use the authenticity trust to determine a peer's credibility. Reliability at time t_i is determined using the authenticity parameter in the following manner:

$$Reliability_{t_i} = \begin{cases} Auth_{t_{i-1}} & \text{if } Auth_{t_{i-1}} \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.8}$$

Similarity (S): This metric is based on the assumption that a legitimate user is more likely to communicate with legitimate peers over the network. The similarity metric measures the similarity of each peer with its neighbors (communicating participants) in terms of similar recommendations. Therefore, legitimate peers should have high similarity, whereas malicious users should have low similarity. To find the similarity for each peer p_j, a set of common peers that were rated by peer p_j and its neighboring peers are obtained. Similarity is then evaluated in the following manner:

$$Similarity = \frac{Number\ of\ Similar\ Recommendations}{Total\ Number\ of\ Recommendations} \tag{4.9}$$

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

$$Honesty = \frac{Number\ of\ Honest\ Ratings}{Total\ number\ of\ Ratings} \tag{4.10}$$

4.4.2 Behavioral Trust

Behavioral Trust describes the trustworthiness of a user based on its popularity in the communication network. Popularity is an important indicator that illustrates user recognition and acceptance by other members in the network. The behavior of a user provides important information that can be used to determine its popularity. We use three basic

attributes of call graphs to describe the communication behavior of a user:

i) Talk time: Talk time (Tk) is the total duration of the calls placed between two participants. The average talk time of a user is the total duration of calls divided by the number of calls placed or received. The frequency and duration of calls are important aspects that define trust relationship between two peers. Repeated calls and long call duration implies that peers have a strong trust relationship. Malicious peers usually have low average talk time as the called party tries to end the communication shortly after noticing malicious behavior [QNTS08]. On the other hand, legitimate peers are more likely to have a respectable average talk time as they are expected to have strong trust relationships with at least a few of their communicating participants [BSG+11].

ii) In-degree: In a call graph of a communication network the in-degree value represents the number of calls received by a user. In-degree is an important attribute that can be used to determine the acceptance of a user within a network. Malicious peers are unlikely to have high in-degree values as they are the least popular members of the network. However, in-degree value alone cannot be used to determine the popularity of a peer, as the number of incoming calls also depends upon a caller's profile. For example, a travel agent will always receive a high number of in-coming calls compared to am accountant.

iii) Out-degree: The out-degree of a user in a call graph represents the number of calls made by the user to other members of the network. Malicious peers are expected to have a high number of outgoing calls as the nature of their activities (Spam, malicious content distribution and Phishing) requires them to make a high number of outgoing call requests.

These attributes can be used to describe the behavior of users in a network. We intend to use these attributes to rank and categorize peers in a communication network. Malicious peers are characterized by their high number of out-going calls, low number of in-coming calls and low average call duration [QNTS08]. On the other hand, legitimate peers usually tend to have high in-degree values and significant talk time [COBY11].

Ranking algorithms are used in social networks to rank nodes using link analysis. The famous PageRank algorithm [PBMW99] determines

the importance of a node based on the number of incoming links. We use the PageRank algorithm to rank communicating peers using their in-degree values. The incoming links are weighted by the talk time between two peers. This ensures that more importance is given to the links that have longer talk time. We define the page rank PR of communicating peer p_i using its in-degree and talk time as follows:

$$PR(p_i) = \frac{1-d}{N} + d \times \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)} \times Tk(p_i, p_j) \tag{4.11}$$

where $M(p_i)$ are a set of peers that link to peer p_i, and $L(p_j)$ are the number of outgoing links of peer p_j. d is the damping factor. In order to consider the outgoing links we use the inverse PageRank algorithm. In inverse PageRank the nodes having high number of outgoing links are ranked the lowest. By weighting the links with their talk time $Tk(p_i, p_j)$, less importance is given to the links that have low call duration. We define the inverse page rank PR' of peer p_i using its out-degree and talk time as follows:

$$PR^{'}(p_i) = \frac{1-d}{N} + d \times \sum_{p_j \in M'(p_i)} \frac{PR^{'}(p_j)}{L'(p_j)} \times Tk(p_i, p_j) \tag{4.12}$$

where $M'(p_i)$ is the set of peers that p_i links to and $L'(p_j)$ are the number of the incoming links of node p_j. In order to consider both incoming and outgoing links we introduce *RankCall* $RC(p_i)$ algorithm which aggregates PageRank and inverse PageRank as follows:

$$\begin{aligned} RC(p_i) = \frac{1 - d_{f_i} - d_{f_o}}{N} + d_{f_i} \times \sum_{p_j \in M(p_i)} \frac{RC(p_j)}{L(p_j)} \times Tk(p_i, p_j) + \\ d_{f_o} \times \sum_{p_j \in M'(p_i)} \frac{RC(p_j)}{L'(p_j)} \times Tk(p_i, p_j) \end{aligned} \tag{4.13}$$

where d_{f_i} and d_{f_o} are the incoming damping factor and outgoing damping factor respectively. In order to ensure the convergence of the algorithm, we use $d_{f_i} = 0.85$ and $d_{f_o} = 0.25$ as noted in the Symrank algorithm [BSG+11]. Symrank algorithm uses in-degree and out-degree values to detect SPIT. However, it does not consider the talk time between communicating peers.

The rank defined by *RankCall* algorithm describes the popularity of a peer in the network. This rank can further be used to categorize peers in order to assign them trust values. This process is illustrated in the example below:

Table 4.2: Communication Network Parameters

Notation	Description	Value
N	Number of Communicating Peers in the Network	300
m_0	number of initially connected peers	20
γ	Power Law Exponent	1.5-2.5
C	Clustering Coefficient	0.75-0.8
n_{exp}	# of experiments over results are averaged	5
Tk_L	Talk time of a legitimate peer (sec)	124-204
Tk_M	Talk time of a malicious peer (sec)	$\leqslant 20$
n	Number of intervals	7

Example: A CSP uses *RankCall* algorithm to categorize its subscribers into sets of highly popular, popular, neutral, unpopular and highly unpopular peers. Various ranking thresholds are set by the CSP in order to categorize peers. The ranking thresholds depend upon the network characteristics such as average talk time etc. The rank of a peer in the network can be used to determine its popularity as follows:

$$Pop(p_i) = \begin{cases} \text{Highly Popular} & \text{if } Rank_{th1} \leqslant RC < Rank_{th2} \\ \text{Popular} & \text{elseif } Rank_{th2} \leqslant RC < Rank_{th3} \\ \text{Neutral} & \text{elseif} Rank_{th3} \leqslant RC < Rank_{th4} \\ \text{Unpopular} & \text{elseif } Rank_{th4} \leqslant RC < Rank_{th5} \\ \text{Highly Unpopular} & \text{elseif } Rank_{th5} \leqslant RC \leqslant Rank_{th6} \end{cases} \tag{4.14}$$

The popularity of a peer p_i is further used to assign *Behavioral Trust*:

$$Tr_{Bev}(p_i) = \begin{cases} +1 & \text{if Highly Popular} \\ +0.5 & \text{if Popular} \\ 0 & \text{if Neutral} \\ -0.5 & \text{if Unpoular} \\ -1 & \text{if Highly Unpopular} \end{cases} \tag{4.15}$$

4.5 Experiments and Results

The performance of *TrustCall* is analyzed in terms of *Authenticity Trust* and *Behavioral Trust* in this section. A network of communicating peers was generated to test the feasibility and effectiveness of our proposed

trust model. Firstly, we show the effectiveness of *Authenticity Trust* against the various types of adversaries that prevail in recommendation systems. Secondly, we demonstrate *Behavioral Trust* by categorizing peers based on their popularity in the network. Lastly, the performance of the *TrustCall* model is compared with that of the *PeerTrust* model.

4.5.1 Experimental Setup

We generated a network of communicating peers to test the feasibility and effectiveness of *TrustCall*. Table 4.2 summarizes the main parameters of the network. The structural properties of telecom call graphs [NGD+06] were used to incorporate the real characteristics of a communication network. In order to simulate a call graph, the BarabasiAlbert algorithm [AB02] was used to generate a random scale-free network of 300 communicating peers. The network was generated using 20 initially connected peers. New peers connect to existing peers with a probability proportional to their communication links. The BarabasiAlbert model uses a preferential attachment mechanism in which new peers introduced into the network prefer to communicate with already heavily linked peers. Therefore, the degree distribution of the network follows a power law distribution. The in-degree and out-degree power law exponents of the network is between $1.5 < \gamma < 2.5$, whereas the clustering coefficient of the network is between $0.75 - 0.8$.

The communicating peers in the network are divided into two sets: Legitimate peers and Malicious peers. The synthetic call workload from [COBY11] is used to set the duration of communication between participants. The call duration between communicating participants is generated using a normal distribution. The talk-time of calls originated by legitimate peers usually are between $124 - 204$ seconds whereas the talk-time of calls originated by malicious peers are generally less than 20 seconds. In the recommendation system built over the communication network we consider two assumptions: legitimate peers provide correct ratings, and malicious peers provide false ratings. The second assumption is generally true as malicious peers usually tend to give false ratings in order to hide their malicious behavior [XL04]. However, the first assumption may not necessarily be true as legitimate peers may provide false rating. Therefore, legitimate peers are considered to rate correctly

with a probability of 0.8. On the other hand, malicious peers always rate other legitimate peers falsely. In the case of collusive group formation malicious peers cooperate with each other in order to enhance their reputations by rating each other falsely.

TrustCall recommends whether a caller is trustworthy or untrustworthy in order to differentiate between malicious and legitimate peers. If the computed trust of the caller is greater than 0, the caller is considered as trustworthy, otherwise it is considered untrustworthy. The decision whether to communicate or not is very personal and is left up to the user to decide. For experimentation purposes, we consider that a user always rejects calls originating from untrustworthy callers and accepts calls originated by trustworthy callers. We used two performance metrics to demonstrate the efficiency of our proposed model :

i) Trust Computation Error: Trust computation error is the total number of errors occurred divided by the total number of communicating peers. An error occurs when *TrustCall* declares a malicious peer as trustworthy or declares a legitimate peer as untrustworthy.

ii) User Satisfaction: User satisfaction in the network is the overall number of satisfied call transactions divided by the total number of calls placed within the network. Users are considered satisfied when they accept a legitimate call or reject a malicious call.

4.5.2 Performance Evaluation of Authenticity Trust

Authenticity Trust evaluates the genuineness and legitimacy of a user's identity. It is based on recommendations received from other members of the network. However, recommendation systems are vulnerable to several threats and adversaries. Therefore, we evaluate the effectiveness and robustness of *Authenticity Trust* against typical adversaries present in the recommendation system. We conducted four different experiments to demonstrate the performance of *Authenticity Trust* in the presence of traitors, false ratings, Sybil attacks and collusive groups. The objective of these experiments is to evaluate the robustness of the *TrustCall* model against different behaviors of malicious peers. Therefore, in each experiment we tested our model such that malicious nodes make up between 0% and 100% of all nodes in the network.

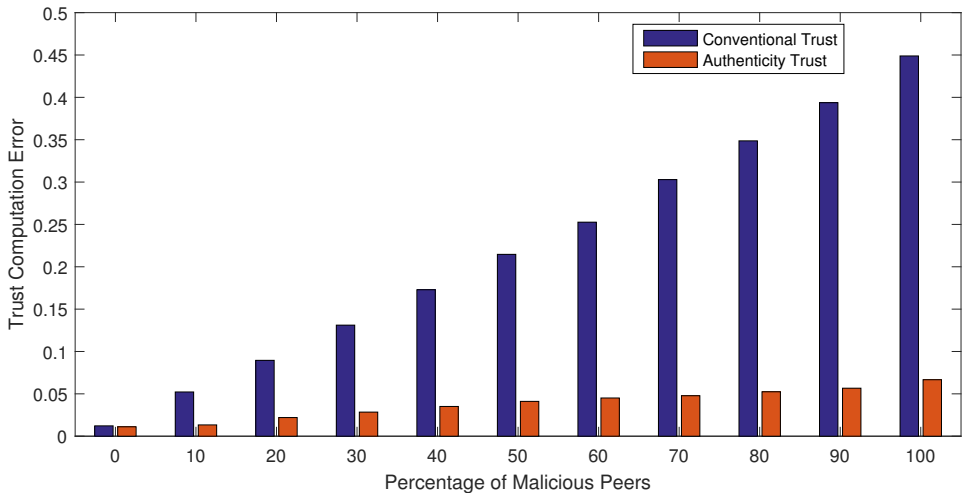

Figure 4.2: Trust computation error in the presence of traitors

Experiment 1: The first experiment shows the effectiveness of *Authenticity Trust* against traitors. In this experiment, we consider a time period divided into 7 equal subintervals over which calls are placed. The trust computation error is computed against an increasing number of malicious peers. We consider 50% of the malicious peers present in the network as traitors. Traitors behave legitimately in the initial subintervals of the time period to earn good reputation after which they start acting maliciously. The other 50% behave maliciously throughout the time period.

Analysis: Figure 4.2 compares the performance of *Authenticity Trust* described by Equation 4.4 with *Conventional Trust* represented by Equation 4.5. It can be observed that the performance of the conventional approach decreases significantly as the number of malicious peers in the network increases. This is due the presence of traitors which remain undetected in the conventional approach. In *Conventional Trust* recent ratings play an insignificant role in altering a peer's trust value. Therefore, traitors can maintain a respectable reputation value by shifting their behaviors. However, in *Authenticity Trust* each rating is weighted based on their positioning in time. Recent ratings are considered more important than old ratings and ratings beyond a specific time period are discarded. This dynamic evaluation allows to detect the behavior of traitors in the network.

Conclusion: Authenticity Trust performs consistently over an increas-

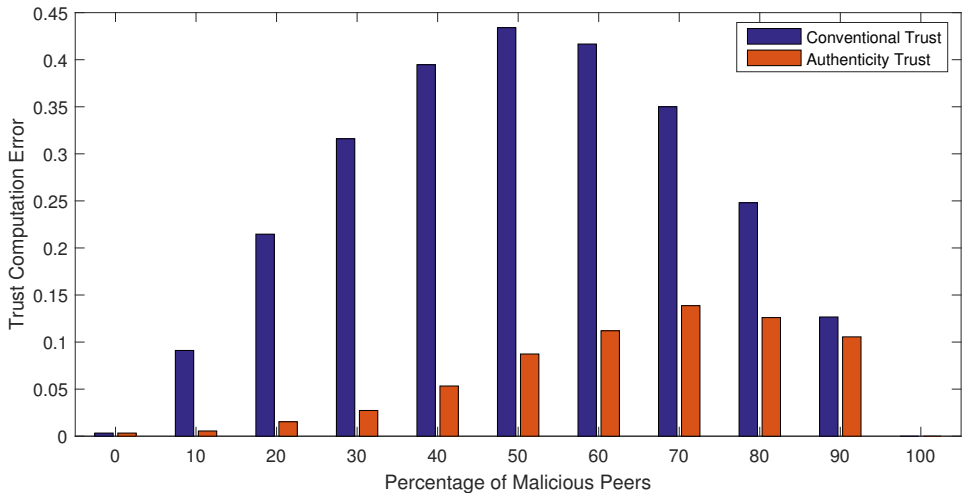

Figure 4.3: Trust computation error in the presence of Sybil Attack

ing number of malicious peers in the network. The dynamic evaluation of *TrustCall* provides an effective mechanism against traitors present in the network. However, peers who consistently behave in an acceptable manner but decide to act maliciously once in a while will still remain undetected.

Experiment 2: The second experiment was conducted to show the robustness of *Authenticity Trust* against Sybil attacks. In this experiment, we consider that 50% of the malicious peers present in the network will carry out a Sybil attack. A Sybil attack is conducted by creating and introducing 30 fake peers in the network. Therefore, for each malicious peer conducting a Sybil attack we inject 30 fake peers into the network. These peers communicate and provide false rating to the user conducting the Sybil attack. As their sole purpose is to enhance user's reputation, such peers have fairly low interaction rate in the network.

Analysis: In Figure 4.3 we compare *Authenticity Trust* with *Conventional Trust*. It can be observed that *Authenticity Trust* provides an effective defense mechanism against Sybil attacks. This is due to the fact that it relies on Social Reliability parameter described by Equation 4.7 to combat Sybil attacks. Social Reliability parameter allows *Authenticity Trust* to consider ratings received from socially reliable peers. In this experiment, we consider a peer to be socially reliable if it has in-degree higher than 5. This threshold is selected based on average in-degree

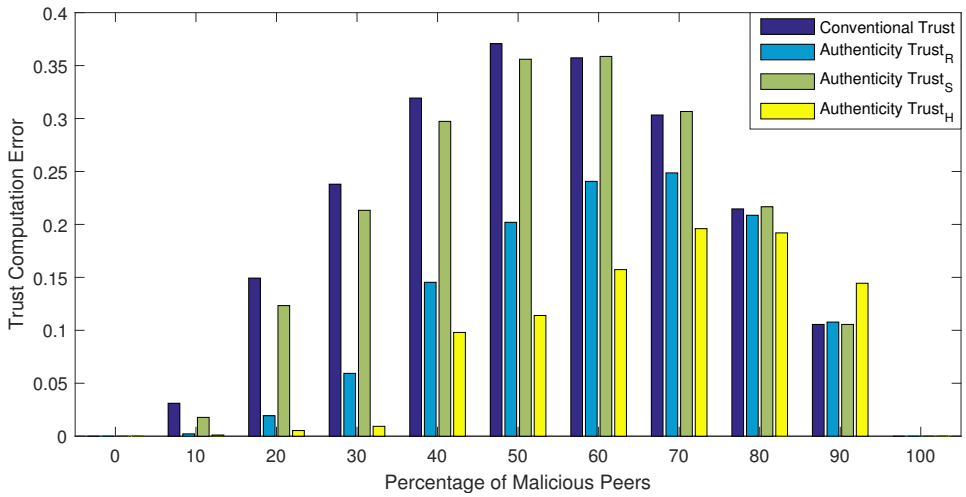

Figure 4.4: Trust computation error in the presence of collusive grouping

value. Ratings from all other peers are rejected as they are likely to be given by fake peers introduced to conduct a Sybil attack. However, this also leads to rejection of rating coming from legitimate peers who have low interaction rate in the network. This experiment does not consider the presence of traitors and collusive groups in the network. Therefore, no error is observed when all peers in the network are malicious in nature.

Conclusion: Social Reliability parameter provides an effective defense mechanism against Sybil attacks by selecting socially reliable peers.

Experiment 3: This experiment compares the three metrics used to compute the credibility of peers. We recall that the credibility parameter determines the ability of a peer to give true recommendations. Three metrics are used to determine user credibility: reliability, similarity and honesty expressed by Equation 4.8, 4.9 and 4.10 respectively. In this experiment, cooperation between malicious peers is not considered. Thus the network considered is non-collusive in nature. A malicious peer present in the network rate other malicious peers correctly whereas rate legitimate peers falsely.

Analysis: Figure 4.4 compares the performance of *Conventional Trust* with that of *Authenticity Trust* in terms of trust computation error. We observe that the conventional approach is very sensitive to peers who provide false recommendations. This is due to the fact that all recommendations received are considered equally. On the other hand, in *Au-*

thenticity Trust each recommendation is weighted with the credibility of the user. We can observe that using the reliability metric in *Authenticity Trust*$_R$ the false recommendations can be considerably filtered out. However, the similarity metric in *Authenticity Trust*$_S$ is not very effective in estimating a peer's credibility. This was attributed to our setup with a highly clustered network in which a large number of calls were placed between malicious and legitimate peers. The Honesty metric in *Authenticity Trust*$_H$ shows the best results in the presence of false ratings.

Conclusion: Honesty metric is able to detect liars present in the network by comparing each rating with the computed trust. Thus each peer's recommendation is weighted with the ability of that peer to lie in the network. In case of very large number of liars present in the network this metric may not perform adequately as computed trust would largely be based on false recommendations.

Experiment 4: Lastly, the feasibility of *Authenticity Trust* is tested under attack by collusive groups formed by malicious peers in the network. In a collusive group, malicious peers cooperate with each other by rating each other falsely, thereby enhancing their reputation in the network. In this experiment, we divide malicious peers into two sets of collusive groups. Malicious peers cooperate with each other by giving false ratings to each other inside the group. However, outside their group they rate correctly. We choose to examine *Authenticity Trust*$_H$ in the presence of collusive groups as honesty metric performs best in determining a peer's credibility.

Analysis: Figure 4.5 shows that the *Authenticity Trust* performs very well in the presence of collusive groups. The performance worsens when the percentage of malicious peers is very high. This is because in large collusive groups a high number of malicious peers cooperate with each other. The evaluated trust is largely based on false ratings and it is difficult for *TrustCall* to detect liars in the network.

Conclusion: Authenticity Trust performs very well in the presence of collusive groups. A very large collusive group is unlikely to occur in a communication network. A high number of smaller disjointed collusive groups may be present.

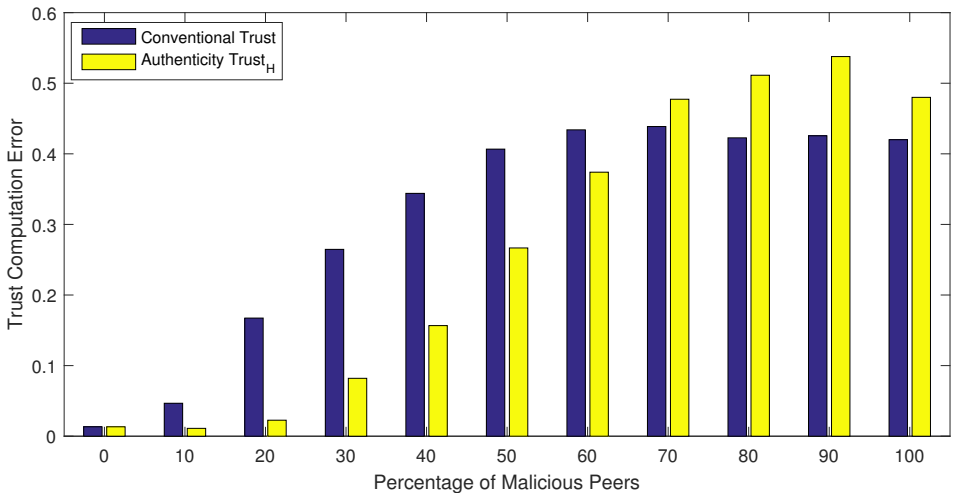

Figure 4.5: Performance of TrustCall in the presence of malicious peers

4.5.3 Performance Evaluation of Behavioral Trust

Behavioral Trust is used to describe the popularity and acceptance of a user in a communication network. The *RankCall* algorithm described by eq 4.13 is used to rank and classify peers based on their in-degree, out-degree and talk time values. This classification is further used to assign them trust values. Therefore, in this experiment we show the performance of our algorithm that assigns *Behavioral Trust* values in *TrustCall* model.

Experiment 5: We consider a CSP having 300 subscribers, where 25% of its subscribers are malicious in nature. The CSP uses *RankCall* to rank callers and categorizes them into different popularity sets as described by Equation 4.14. In this experiment we illustrate the average behavior of all peers in the network compared with the popular and unpopular ranked peers in the network. The 10% of the highest-ranked peers are declared as popular whereas 10% of the lowest-ranked peers are declared as unpopular.

Analysis: Figure 4.6 illustrates the mean degree of the peers communicating in the network. We can observe that the in-degree of popular peers is much more than the average in-degree of the network. This shows their importance and acceptance in the network. On the other hand, the in-degree of the lowest-ranked peers is very low compared to the average in-degree value of the network. Furthermore, their out-degree value is

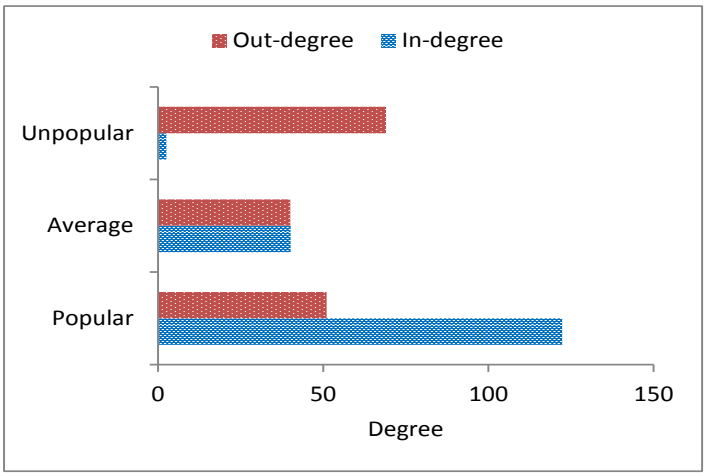

Figure 4.6: Mean degree of the network

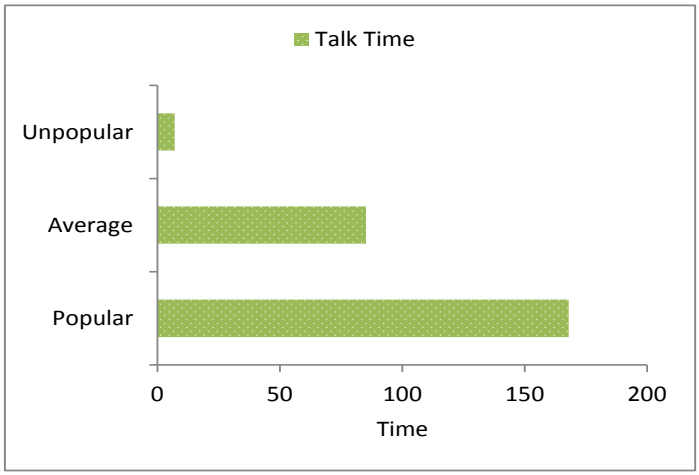

Figure 4.7: Mean talk time of the network

almost 10 times that of their in-degree value. Thus, they are likely to be involved in malicious activities such as Spam, Phishing and nuisance marketing. As they are highly unpopular due to their involvement in malicious activities, they do not receive a lot of incoming calls from other members of the network. We can also observe from Figure 4.7 that the call duration for the lowest ranked peers is much lower than the average talk time of the network. Call duration is an important metric to define trust relationships between peers. Therefore, higher ranked peers should be assigned with high trust values, while low-ranked peers should be assigned with low trust values as they are the least trusted in the network.

Conclusion: This experiment proves that peers with low in-degree, low call duration and high out-degree values are ranked the lowest by the *RankCall* algorithm. Such peers are much more likely to be malicious in nature. Therefore, *Behavioral Trust* assigns them low trust values as described in Equation 5.4. Callers who have high in-degree and high call duration values are ranked the highest and thus will be more likely to act legitimately in the network. The behavior patterns of each caller vary based on different attributes of their profile such as geographical location, profession and interests. As our future work, we plan to study the behavior of different types of users present in the communications network to enhance mechanisms in order to evaluate *Behavioral Trust*. Furthermore, we intend to use behavior patterns to differentiate between different types of malicious behaviors such as fake profiles, Spam, Phishing etc.

4.5.4 Effectiveness of TrustCall

In this subsection, we compare our approach with one of the most simplistic yet effective trust mechanisms, known as 'PeerTrust'. Peertrust is a reputation based trust model used over P2P file sharing networks. It considers various factors to quantify the trustworthiness of users over P2P networks [XL04]. PeerTrust provides a reasonably good performance against oscillating behaviors, collusive groups, false ratings, and man-in-the-middle attacks in reputation systems. The performance of PeerTrust is equivalent to that of other popular P2P trust models such as EigenTrust and PowerTrust [MP09b]. However, it is considered to be

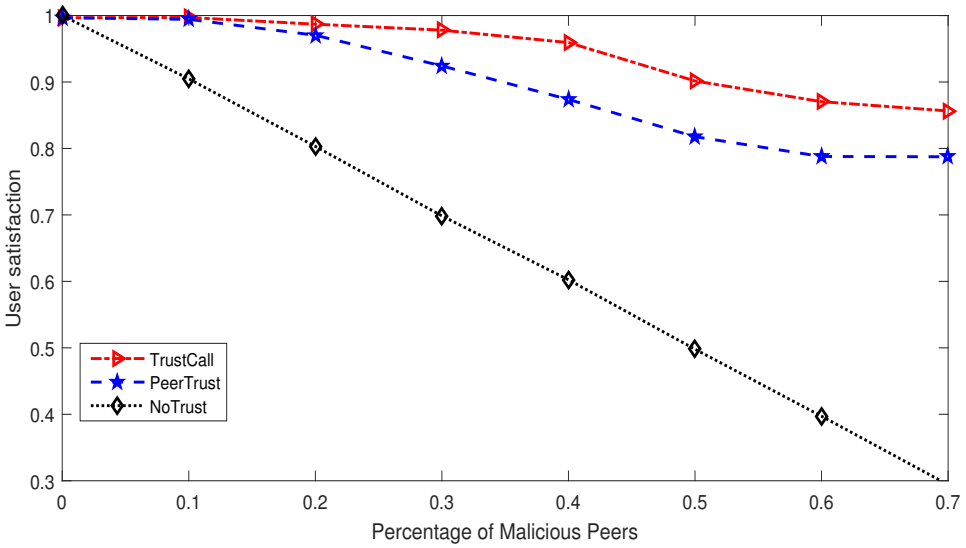

Figure 4.8: Performance Comparison: User Satisfaction in Non-Collusive Network

the most simple and easy to implement trust model. Therefore, we chose to compare the performance of *TrustCall* with *PeerTrust* when applied over web conversational services. However *PeerTrust* only considers recommendations to evaluate user reputation. Therefore, in order to have a fair comparison we chose $\alpha = 1$ for *TrustCall* model in eq 4.2.

Experiment 6: We compare *TrustCall* with *PeerTrust* under three scenarios i) collusive ii) non-collusive and iii) Sybil attacks as shown in Figure 4.8, Figure 4.9 and Figure 4.10 respectively. The network settings for collusive, non-collusive and Sybil attacks are same as those specified in Section 4.5.2. We compare user satisfaction when no trust is computed and when trust is computed using *TrustCall* and *PeerTrust*.

Analysis: Figure 4.8, Figure 4.9 and Figure 4.10 provides the performance comparison in terms of user satisfaction with respect to the number of malicious peers present in the network. If trust is not computed all calls are accepted whether they are malicious or legitimate in nature. On the other hand, trust computation allows users to decide whether to accept or reject calls based on a caller's trustworthiness. Trust is computed using *PeerTrust* and *TrustCall* models. We can observe that there is a linear decrease in the performance when no trust is computed. However, user satisfaction improves immensely when the caller's trustworthiness is computed.

Conclusion: TrustCall outperforms *PeerTrust* in all three scenarios.

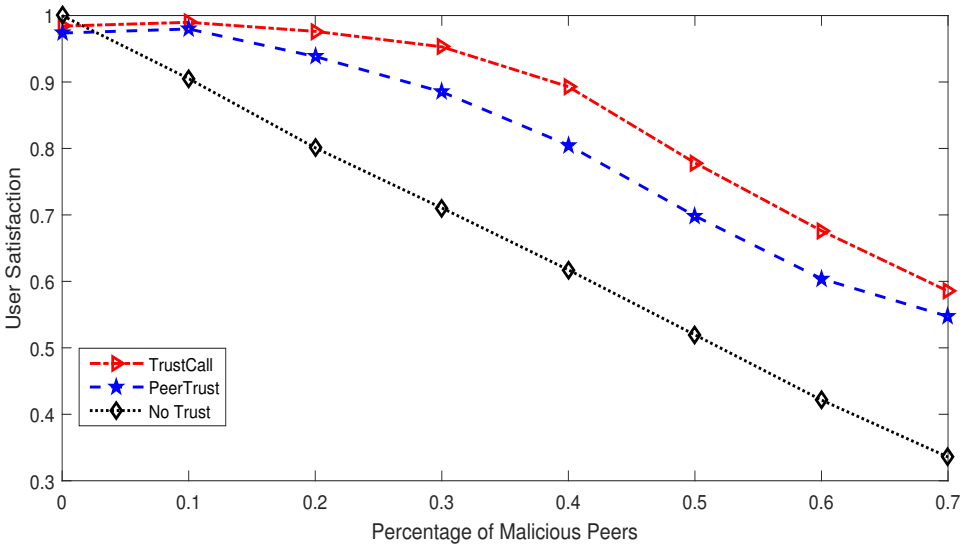

Figure 4.9: User Satisfaction in Collusive Network

Thus, proving its effectiveness over communication networks. The trust computed by *TrustCall* is better suited to distinguish between malicious and legitimate users present in communication networks. However, peers may be able to discard their bad reputation by re-entering the network with a new identity. To completely prevent users from re-entering the network, new methods of identity verification should to be introduced that would restrict users ability to make duplicate identities over communication networks.

4.6 Conclusion

In this chapter, we detail potential social security threats present over web communication networks. A simplistic heuristic-based trust model is proposed that computes reputation of callers in web communication networks. *Authenticity Trust* is computed that describes the legitimacy and genuineness of user identity based on the recommendations received from its communicating participants. Experiments are conducted to demonstrate the performance of *Authenticity Trust* in the presence of traitors, false ratings, Sybil attacks and collusive groups. *Behavioral Trust* is used to rank the peers based on their popularity and acceptance of a user in a communication network. Experiments prove that callers having high

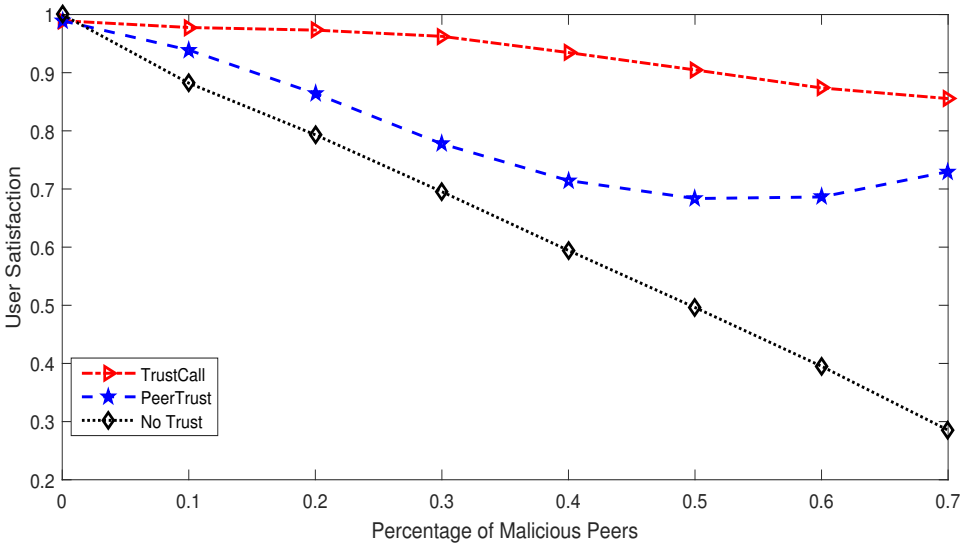

Figure 4.10: User Satisfaction under Sybil attacks

out degree, low in degree and short call duration are ranked the lowest in the network.

Chapter 5

Combating Nuisance Calls

" Every major communication tool on the Internet has spam and abuse problems. "
Evan Williams

Contents

5.1 Introduction

The free of cost nature of web conversational services has attracted telemarketers, prank callers, and spammers to send nuisance calls. Nuisance calls can be described as unsolicited bulk spam phone calls generated over communication networks for sales, marketing and deceptive purposes. Nuisance calls can be manually or automatically generated. Nuisance calls are not only irritating and annoying but also pose several security threats. It remains essential for web operators to mitigate nuisance calls in order to protect their subscribers. Detecting nuisance calls over phone networks remain a difficult challenge as call content cannot be judged before it takes place. The most feasible approach is to inspect call behavior of users in the network.

In this chapter we provide present nuisance call model that describes different types and attributes of nuisance calls. Secondly a nuisance call detection framework "*N-Combat*" is designed that computes user reputation based on the call duration, out-degree, the callee's reliability and on feedback received from callee. A dual time window is used to address the dynamic behavior of spammers and a watchdog mechanism to deter whitewashing attacks in communication networks. We compute the performance of our framework in the presence of four types of callers: ordinary, specific, telemarketers and auto-dialers. The experiments prove that the framework is effective in detecting a high amount of spam generated by telemarketers and auto-dialers. Experiments also show that *N-Combat* has a very low false negative rate when compared to existing threshold based SPIT detection methods.

5.2 Related Work

SPIT is defined as unsolicited, automatically dialed pre-recorded telephone calls over VoIP networks [QvdMP07]. One of the most feasible approach to detect SPIT is to inspect each caller's CDR [Ker12]. Several calling features such as call frequency and average call duration are computed to detect spammers in the network.

Authors in [BSG+11] use real phone call datasets to identify two features, namely Strong Tie property and Weak Tie property. Strong Tie

property shows that legitimate caller spend 80 % of their talk time with only 4-5 people. Whereas, Weak Tie property declares users making large number of outgoing calls having very short call duration as suspicious. The authors further propose a ranking algorithm namely SymRank. SymRank uses both in-degree and out-degree of callers to rank them in the network. The low ranked callers are likely to be spammers as they have low number of in-coming calls and a high amount of out-going calls.

Progressive Multi Gray-leveling (PMG) [SAS06] uses call frequency to distinguish the spammers from legitimate ones. As spammers make a large number of calls, their call frequency is much higher than legitimate callers. In order to detect spammers efficiently PMG monitors the behaviors of caller in short-term and long-term. Discrete Event System Specification (DEVS) [KKKJ09] proposes SPIT level classification based on six different call features. Each feature is weighted based on the possibility of manipulation and strength of evidence for a SPIT level decision. The largest weights are given to the number of callees and call duration. A decision threshold is defined to decide whether a caller is spammer or not.

CallRank [BAP07] computes direct trust between callers using the average call duration. The global reputation of callers is computed using Eigen trust in order to combat SPIT. CallerRep [AM13] improves CallRank by computing global reputation using three call features call duration, interaction rate, and caller out-degree. Authors in [SWN12] propose spam combating system based on the entropy of average call duration. They applied mahalanobis distance to call duration and time of call to distinguish SPIT from non-SPIT callers. Voice Spam Detector (VSD) [KD07] is a multi-stage SPIT filter based on call pattern and volume analysis, and feedback among the various filter stages.

The existing SPIT detection mechanisms have a number of limitations. Firstly, they fail to focus on eliminating the likelihood of blocking legitimate calls. Legitimate callers such as call center representatives, emergency services and job seekers tend to have high number of short duration out-going calls in a short duration of time. This results in a false detection of legitimate callers as their call features are similar to spammers. Secondly none of the existing behavioral based reputation

mechanisms have dealt with whitewashing attacks. Web communication services allow user identities to be created without any identification proof. This facilitates spammers to shed their bad reputation and re-enter the network in order to continue spamming.

Lastly, most of the existing method focus on detecting automatically dialed pre-recorded spam calls. These methods have not considered combating manually generated spam calls. A study conducted in [UKs17] shows that the percentage of users receiving live telemarketing calls are approximately two times higher than that of recorded calls. Furthermore, according to the findings of the YouGov survey [WS14], 39.6 out of 57.6 million people who have been contacted by telemarketers feel that they need more protection against them.

5.3 Nuisance Call Model

Nuisance calls are described as unsolicited bulk phone calls generated over communication networks. The purpose of nuisance calls is to advertise for sales and marketing purposes. They may also include deceptive information used for Phishing and scam purposes. In this section, we present a nuisance call model which includes the classification, attributes, criteria and threat model related to nuisance calls.

Types of Nuisance Calls: Nuisance calls can be manually generated or automated. We classify nuisance calls into four major types:

1. *Live:* A live nuisance call is a call made by a real person for sales or deceptive motives. Telemarketers are salespersons whose objective is to persuade customers to buy their products or services during calls. Other than that scam calls are used to commit frauds over the phone such as Phishing where a caller tricks a user into providing their confidential information such as their credit card information.

2. *Recorded:* Recorded nuisance calls are computer-generated automatically dialed pre-recorded phone calls broad-casted for marketing and promotional campaigns.

3. *Silent*: Silent calls are generated by automated calling systems used to generate DoS attacks by bombarding the network with silent calls.

Attributes of spammers: The main objective of spammers is to generate calls for marketing, sales and Phishing purposes. This makes their specific call behavior distinguishable in the network:

- *Diversity:* Spammers attempt to cover a large number of unique users with a non repetitive call behavior.
- *Non-reciprocal behavior:* Spammers rarely receive calls from users with whom they communicate.
- *Unbalance in-out degree:* Spammers have a high amount of outgoing calls with a very low amount of incoming calls.
- *Short call duration:* The average call duration of a spammers is short, as the callee tries to end the conversation after detecting the nature of the call.
- *High Call Rate:* Spammers usually try to send as many calls as possible in a certain period of time.

Criteria to Mitigate Nuisance Calls: For nuisance mitigating mechanisms to be effective over Internet telephony they should meet a number of criteria:

1. *Least Delay:* The mechanism should not cause any observable delay to the call connection process.
2. *Efficient:* The nuisance detection mechanism should maximize the detection of nuisance calls while eliminating the likelihood of blocking legitimate calls.
3. *Robust:* The nuisance combating mechanism should be robust against security threats generated to avoid spammers' detection.

Threat model: To design a robust nuisance combating solution, it should take into account different type of threats and attacks summarized as follows:

- *Whitewashing:* Upon detection a spammers can easily shed its bad reputation and call records by purposely leaving and re-entering the network with a new identity in order to continue spamming.

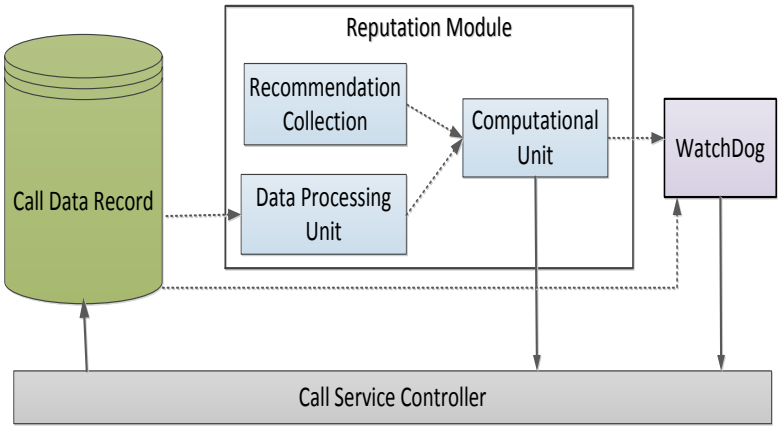

Figure 5.1: Functional Architecture of N-Combat

- *Dynamic behavior:* Spammers may behave legitimately in the network before starting to spam in order to avoid their detection for a longer time.
- *False recommendation:* Spammers may provide false recommendations to legitimate callers.
- *Collusive group:* A collusive group is a group of users who communicate with each other in order to compensate for the short average call duration and other behavioral features that are used to detect spammers

5.4 Nuisance Call Combating Framework

In this section, we propose *N-Combat*: a nuisance call combating framework for web communication services. The general architecture of *N-Combat* is presented in Figure 5.1. It consists four components *Call Data Record*, *Call Service Controller*, *Reputation Module* and *Watchdog* that will be detailed in the following subsections.

5.4.1 Call Data Record

The service provider records each call transaction in a *Call Data Record* (CDR) server. It contains the following information about a call: caller identity, callee identity, time-stamp and completion status. The CDRs

are potentially infinite and it is not possible to store them completely. Furthermore, older data might become irrelevant over time. Therefore, we apply the concept of a sliding window to our data collection. A window is represented by T_k where $k > 1$ consists of n time units t. Each new window is formed by inserting a new unit into its previous window and deleting the oldest one. The size and number of time units in each window depends upon the policy of the service provider.

5.4.2 Reputation Module

The *Reputation Module* consists of data processing, feedback collection and computational units.

The data processing unit uses the CDR to create the neighborhood network of each caller. The neighborhood is collection of a caller and the entire set of callees to whom that caller has called. An example of a neighborhood is shown in Figure 5.2 where the caller has four callees. This caller has repetitive behavior with callees u_2 and u_3 whereas it has a reciprocal calling behavior with callees u_2, u_3 and u_4. The Total Call Duration (TCD) between caller and callee is the sum of the durations of all calls placed between them in both directions. This incorporates the reciprocal and the repetitive nature of calls between caller and callee.

The feedback collection unit is responsible for collecting reports about a caller. Callee u_j can report caller u_i as follows:

$$Report_{u_j \rightarrow u_i} = \begin{cases} 0 & \textit{if reported as spammer} \\ 1 & \textit{if not reported} \end{cases} \tag{5.1}$$

In order to combat false recommendation the feedback collection unit imposes *Report Collection Strategy* (*RCS*): i) a report cannot be independent of a call process session; ii) each callee can report a caller whereas a caller cannot report a callee; and iii) a callee can only report a caller once even if there are multiple calls placed. Using RCS the impact of false feedback in the network can be reduced. The feedback is restricted to callees only, as spammers generate a high amount of calls and are highly likely to provide false reports.

The computational unit computes the reputation of each caller using: i) the call duration between a caller and each callee; ii) the reliability

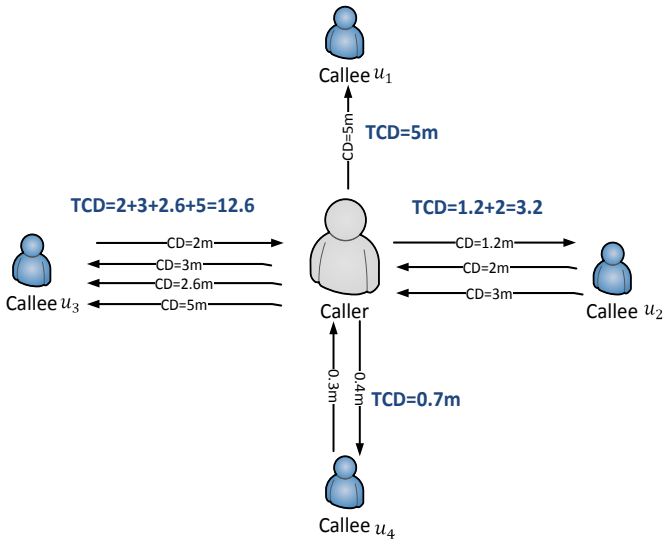

Figure 5.2: Caller's neighborhood in communication network

of each callee; iii) the callee's feedback; iv) the out-degree of the caller. The reputation of a caller has a numeric value between 0 and 10. The reputation $Rep^{T_i}(u_i)$ of the caller u_i for time window T_i is presented as follows:

$$Rep^{T_i}(u_i) = \frac{\sum_{j=1}^{n_{u_i}} TCD_{u_i,u_j} \times Report_{u_j \to u_i} \times Rel^{T_{i-1}}(u_j)}{OD_{u_i}} \tag{5.2}$$

where n_{u_i} is the total number of users called by u_i and OD_{u_i} is the out-degree of caller u_i. TCD_{u_i,u_j} is the Total Call Duration between caller u_i and callee u_j. TCD is capped to a value of 10 min, thus has a range of $0 < CD \leqslant 10$ min. A higher TCD indicates a strong trust relationship whereas a low TCD represents a weak trust relationship between two users. The TCD_{u_i,u_j} is weighted with the amount of reliability of each callee, represented by $Rel(u_j)$. Legitimate users in the network are more likely to be reliable whereas the malicious users in the network are highly unreliable [XL04]. Therefore, reliability is directly linked to the reputation of a callee computed at previous time window T_{i-1} as follows:

$$Rel^{T_i}(u_j) = Rep^{T_{i-1}}(u_j)/10 \tag{5.3}$$

Dual Time Window: To address the dynamic behavior of spammers in the network, we propose the concept of a dual time window. A caller's

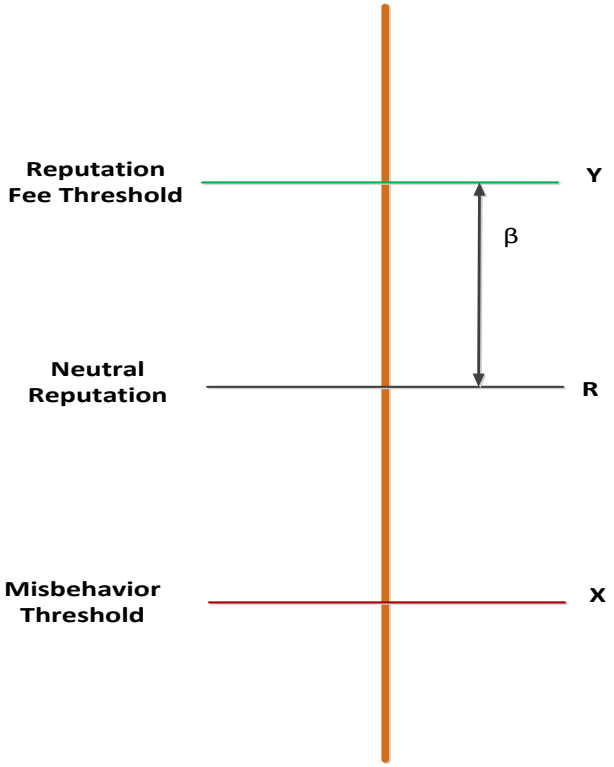

Figure 5.3: Reputation Levels

reputation is evaluated in the recent time window T_k producing reputation value Rep^{T_k}. Meanwhile, another reputation value $Rep^{\hat{T}_k}$ is computed using a time window $\hat{T}_k$ that is much smaller than the original time window T_k. The smaller time window reflects the caller's most recent behavior. The second value $Rep^{\hat{T}_k}$ will be returned as the final trust value of caller u_i if it is smaller than the first value by a certain threshold. This indicates that the peer has started spamming recently. Otherwise, the first value Rep^{T_k} will be returned as follows:

$$Reputation(u_i) = \begin{cases} Rep^{\hat{T}_k} & if\ Rep^{T_k} - Rep^{\hat{T}_k} > threshold \\ Rep^{T_k} & otherwise \end{cases} \tag{5.4}$$

The dual time window makes the reputation of a caller difficult to build, as its reputation cannot be quickly increased by a small number of good call transactions, but it will quickly drop if the caller starts spamming in the network.

5.4.3 Watchdog

The *Watchdog* component is used in order to deter whitewashing attacks. The lowest reputation a caller can have without being detected as a spammer is the threshold X as shown in Figure 5.3. The chance of whitewashing attack increases when the reputation of a caller is close or below to the threshold X. In order to prevent this our system categorizes callers as newcomer and mature.

Newcomer: A newcomer caller is a new user that enters to our system. No historic or social relationships are provided for this user. For this reason, our system will give him the opportunity to communicate without being checked for nuisance calls. However, the newcomer will have certain restrictions depending upon the service provider policy. For instance, it could communicate with certain number unique callee in the network or it will be able to send certain number of call requests in a fixed time period. These restrictions will not allow newcomer to spam in the network.

Mature: A mature user will be allowed to communicate freely in the network. An newcomer has to achieve the reputation of threshold Y and pay a fee β in the form of good reputation a fixed time period T, in order to become a mature caller. This is the social cost incurred to newcomers in order to communicate freely in the network. Upon becoming a mature caller it will gets a neutral reputation R which then allows the caller to communicate freely without any restrictions.

In our system, spamming in the network cannot be realized before becoming a mature caller. To achieve a status of mature caller, the user needs time and a good reputation level. Each time, an attacker changes its identity, it will be required to earn a respectable reputation before starting spamming in the network. Therefore, our approach removes the advantages that whitewashing attackers can provide for spammers while it gives newcomers a fair chance of making social relationship.

5.4.4 Call Service Controller

The *Call Service Controller* is responsible for processing all call requests. It extracts information from different components of the framework when a user initiates a call request. The steps performed by the *Call Service*

Controller are listed below.

Step1: The caller and callee's identity are extracted from the call request. The status of caller is determined using the *Watchdog* component.

Step 2: If caller is *newcomer*, the conditions of the restrictions applied to newcomers by the service provider are checked. The call request is sent if the caller satisfies the conditions, otherwise it is rejected.

Step 3: If caller is mature, the *reputation module* is used to check the reputation of the caller to determine whether the incoming call is a nuisance or is legitimate using a predefined threshold as follows:

$$Call\ Type = \begin{cases} Nuisance & if\ Reputation(u_i) < Threshold \\ Legitimate & otherwise \end{cases} \tag{5.5}$$

Step 4: If the reputation of the caller is higher than a certain threshold the call is sent. Otherwise, based on the callee preference *Call Service Controller* will either i) reject the call, ii) sent the call request with warning, or iii) notify the callee about the call request. Based on caller reputation level the service provider may also decide to punish the caller by blocking future calls for a certain time period.

5.5 Experimentation and Results

In this section, we analyze the performance of *N-Combat* by conducting simulations.

5.5.1 Network generation

Call transaction records are very hard to obtain due to privacy issues. Therefore, we use the structural properties [NGD+06] of telecom call graphs and call statistics [MACL10] [KKKJ09] [BSG+11] to generate a synthetic call data record. In our network setting (Table 5.1), we simulate ordinary and specific callers as legitimate whereas telemarketers and auto-dialers as spammers.

Ordinary callers usually have a high call rate within their social group, and moderate call rate outside their social group. We use Poisson distribution $P(x) = \frac{e^{-\lambda}\lambda^x}{x!}$ for call rate with a mean $\mu = 2$ calls per time

Table 5.1: Configuration

Caller Type	Nature	Call Duration	Call Rate
Ordinary	Legitimate	Exponential $\mu = 6$	Poisson $\mu = 2$
Specific	Legitimate	Exponential $\mu = 5$	Poisson $\mu = 7$
Telemarketer	Spammer	Exponential $\mu = 5$	Poisson $\mu = 7$
Autodialer	Spammer	Exponential $\mu = 2$	Constant=10

unit. 80% of the calls generated by ordinary caller are distributed within their social group [BSG+11]. Ordinary callers have long-duration repetitive and reciprocal call behavior with its social group and short-duration moderate calling rate outside the social group. Their call duration exhibits an exponential distribution $P(x) = \lambda e^{-\lambda x}$ with average holding time $\mu = 6$ minutes.

Specific callers are legitimate callers such as emergency services, banks, job seekers which have high out-degree and moderate call duration. Therefore, we use poison distribution with $\mu = 7$ and exponential distribution $\mu = 5$ for call rate and call duration respectively. We use similar call parameters for telemarketers as they have non-repetitive and non-reciprocal call patterns with a high out-degree. Autodialer are computer generated pre-recorded spam have short duration calls that usually do not exceeds 2 min. Auto-dialers are programmed to call specific amount of callee in time period. Therefore, we use a constant value of 10 callee per time unit for the call rate. Spammers try to cover a large number of callee while rarely receive calls from legitimate callers.

5.5.2 N-Combat Performance

We compute the effectiveness of *N-Combat* in detecting different types of callers in the network. We simulated a network of 300 callers consisting of 60% ordinary, 10% specific, 10% telemarketers and 10% auto-dialers. A time window consist of $n = 5$ time units. Each new window is formed by inserting a new time unit into its previous window and deleting the oldest one. We use the detection accuracy metric to compute the performance. Accuracy is the number of correct identification of caller's nature over total number of similar nature callers. A correct identification oc-

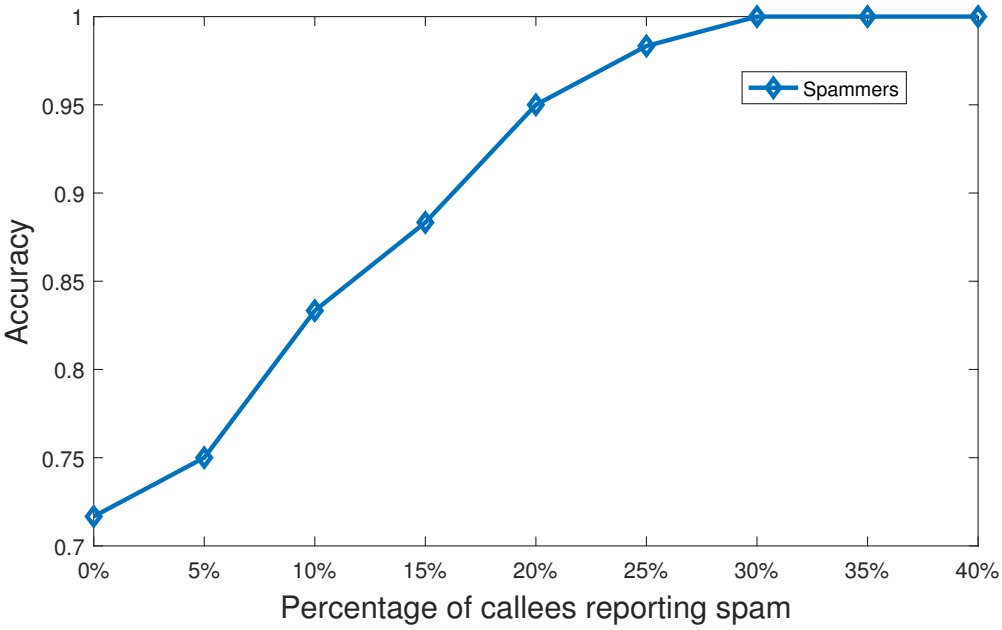

Figure 5.4: Impact of Spam reports

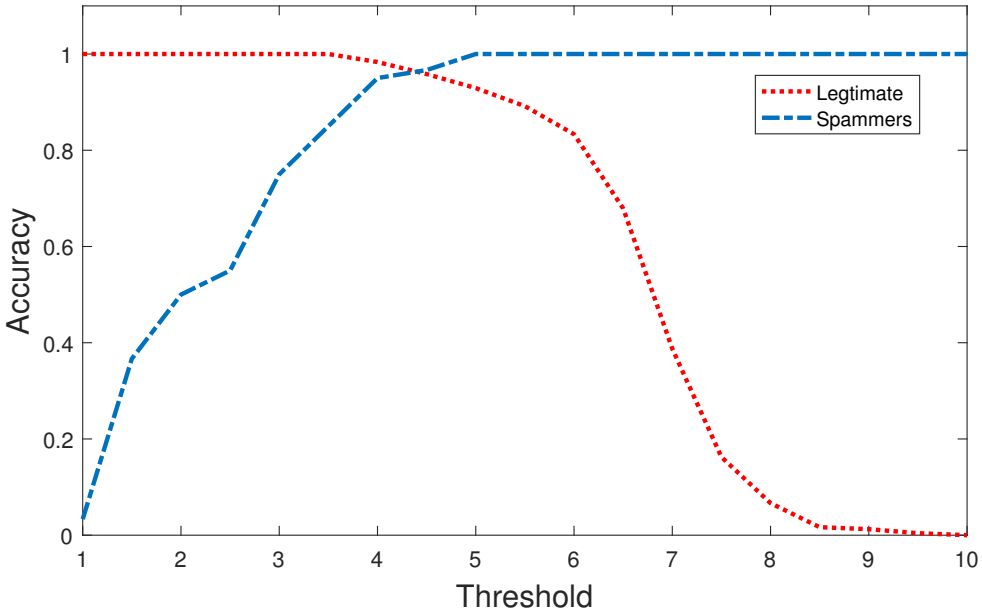

Figure 5.5: Affect of reputation threshold

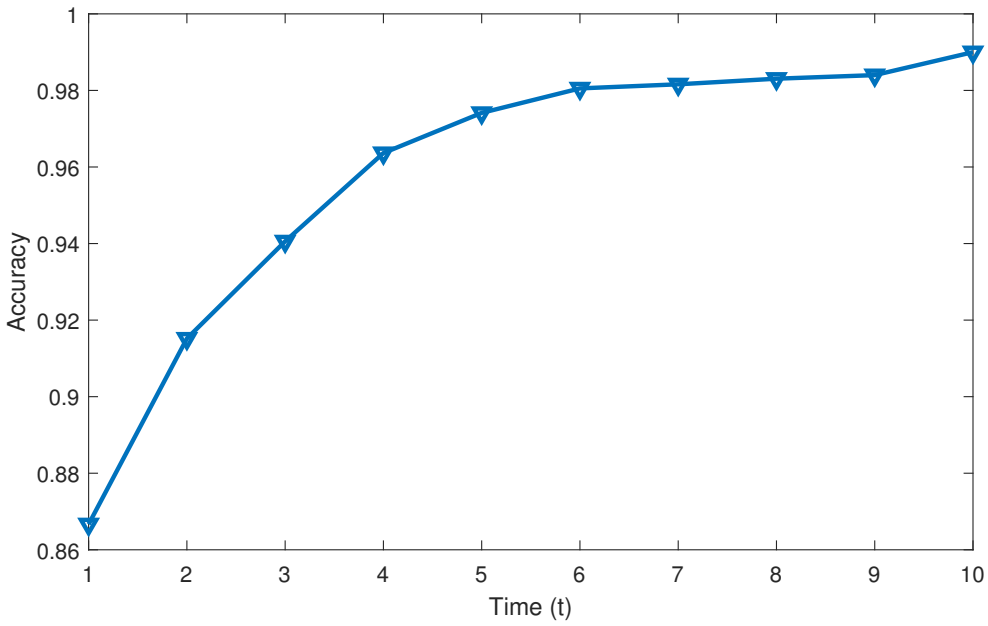

Figure 5.6: Detection accuracy with increasing time

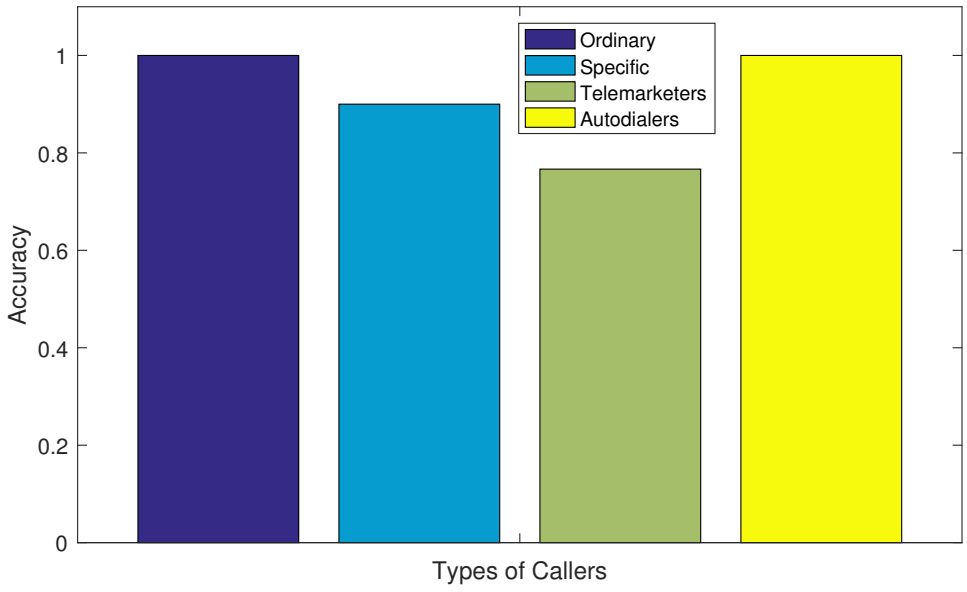

Figure 5.7: Detection accuracy of different types of callers

curs when ordinary and specific callers are detected as legitimate while telemarketers and autodialer are detected as spammers. In the following we details and discuss some of the experimentation results.

Figure 5.4 represents detection accuracy of spammers with increase percentage of spam calls being reported by callee. The detection accuracy increases with the expansion of callee reports. Reporting spam decreases their overall reputation which allows them to be detected more easily. We can notice that if 30 percent of the callee in the network start reporting the accuracy goes to 1. In order to encourage more callers to report Spam in the network incentives may be given to provide feedback about caller.

Figure 5.5 gives a reference to select an appropriate reputation threshold. In this figure, we can observe that a large number of legitimate callers will be detected as spammers if the threshold is set higher than 6. On the other hand, a large number of spammers will be wrongly identified in case the threshold is set lower than 3. The objective is to maximize the detection of nuisance calls while minimize the likelihood of rejecting legitimate calls. Calls generated from legitimate callers are very important and must not be wrongly identified. Therefore, a recommendable threshold should be around 4 where almost none of the legitimate callers are wrongly identified while maximum number of spammers are correctly identified.

Figure 5.6 represents the detection accuracy increase with simulation time. The threshold is set to 4 whereas 30% callees are considered to report spam in the network. At initial stage, the callers have neutral reputation. With increasing of time the call data records and feedback from callers allow callers to build their reputation. The legitimate callers gain reputation while the spammers lose their reputation. Figure 5.7 further details the detection of caller's type in the first time window. Ordinary and autodialer are are detected easily with an accuracy of 1. Whereas specific callers and telemarketers are difficult to detect because of their similar calling behavior.

These experiments show the effectiveness of *N-Combat* mechanism against different types of callers present in the network. *N-Combat* is able to maximize the detection of spam calls by detecting telemarketers and auto-dialers in the network. With a high amount of spam being

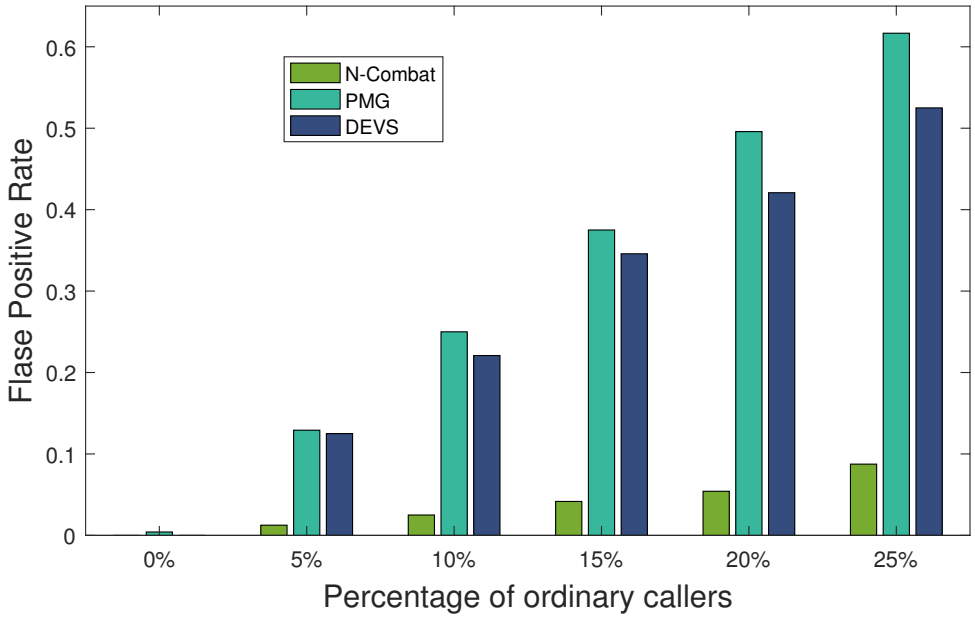

Figure 5.8: Performance Comparison using False Positive Rate

reported, *N-Combat* will be able to have a high detection accuracy for spammers. An appropriate selection of threshold will allow *N-Combat* to effectively differentiate between specific callers and telemarketers. This will result in a very low false detection rate where calls generated by specific callers such as job seekers and call center representatives will not be detected as spam.

5.5.3 Comparison

We use the previously defined network to compare *N-Combat* with closely related threshold based SPIT combating models namely PMG [SAS06] and DEVS [KKKJ09]. PMG uses call density to determine a grey level for spammer. If the grey level of a caller reaches a certain threshold the calls made by caller are blocked. DEVS [KKKJ09] is based on call duration and number of call recipients. A decision threshold is defined to decide whether a caller is a spam caller or not. In order to compare between the three solutions PMG, DEVS and our framework (*N-Combat*), we use *False Positive Rate* for comparison. *False positive rate* is the amount of legitimate callers wrongly identified as spammers over the total number of legitimate callers in the network.Figure 5.8 compares the performance

of *N-Combat* using *False Positive Rate* against different percentage of ordinary callers present in the network. Specific callers are legitimate callers that have similar call patterns than spammers. Existing methods such as PMG and DEVS do not consider the presence of such callers in the network. The figure shows that *N-Combat* outperforms PMG and DEVS in terms of *False Positive Rate*. Thus, it can be concluded that *N-Combat* is an effective mechanism having extremely low *False Positive Rate*.

5.6 Conclusion

In this chapter, we addressed the problem of combating nuisance calls generated over web communication services. Our approach is based on the computation of caller's reputation using call history and feedback collected from callees. The framework also provides a solution to mitigate whitewashing attacks. We compute the performance of our framework in the presence of four types of callers: ordinary, specific, telemarketers and auto-dialers. The experiments prove that the framework is effective in detecting a high amount of spam generated by telemarketers and auto-dialers. *N-Combat* has a very low false negative rate when compared to existing threshold based SPIT detection methods.

Part IV

Conclusion

" In every end, there is also a beginning. "
Libba Bray

In this thesis, we analyze the challenge of establishing trust between participants in real-time web communication environments. We propose a novel and comprehensive framework for computing trust in real-time web communication services. The framework provides information about the trustworthiness of callers present in the communication networks. This chapter summarizes the major achievements and open issues that requires further investigation.

Summary of Contributions
In this thesis, we have accomplished the following:

- We have firstly done an interesting literature review. In this review we compared the telco federated model with the OTT web model. We further presented the threat taxonomy for VoIP detailing different types of threats. In order to protect users from social threats, trust between communicating participants needs to be computed. To understand the concept of trust we therefore studied different trust computation methods used to enhance security in online applications.

- We have formally defined the concept of trust in real-time web communication. We have identified the major trust relationship and parameters used for computing trust in WebRTC.

- We proposed the use of OpenID Connect protocol to authenticate communicating peers in web communication services. We showed how OIDC can be effectively used for identity provisioning while preserving user privacy.

- We presented a trust computational model to evaluate the reputation of each caller in communication network. Our solution is based

on Authenticity trust and Behavioral trust. Authenticity trust describes the legitimacy of a caller's identity whereas behavioral trust shows the caller's popularity and acceptance in the network. The computed trust is used to advise and assist whether and how much a particular user can be trusted over the communication network.

- We proposed a nuisance call combating mechanism that will allow web service providers to differentiate between spammers and legitimate callers. We use the behavior of the caller to evaluate its reputation in the network. The caller's reputation is then used to detect nuisance calls generated over the network. Upon detection the nuisance call is either blocked or sent with a warning to alert the callee.

Future Contributions

The research on computing trust in real-time web communication can be continued in several directions. The contributions in this thesis has led to some open issues which requires further investigation. We identify three major areas for future work.

- *Trust Visualization*: In literature there are only a few visualizations that target the communication of trust. However, these basic trust visualizations (such as star interface used e-commerce) are able to only communicate a single aggregated trust value. Many factors such as the reliability of trust value are hidden from the user. As a future work, we have to take steps towards understanding how to communicate trustworthiness of callers in web communication services. Novel trust visualizations should be designed based on findings from human-computer interactions by conducting user surveys. The visualization should potentially improve decision making of users in the network.

- *Reputation Interoperability*: Future communication platforms will enable cross-domain interoperability allowing subscribers from different domains to communicate with each other. Therefore, trustworthiness of a caller computed by one domain should be understandable in other domains. However, the trust computational methods are developed for a closed domains. Every domain has

its own information sources, computational methods and representations. As a result, interpreting and transferring reputation of a caller from one domain to another is not possible. As a future work, data models should be developed for exchanging reputation information. Ontologies and other semantic web technologies should be used to achieve trust interoperability and portability.

- *Caller Behavior Analysis*: Call data records can be used to retrieve information about caller's behavior in the network. The study of graph theoretic information from call graphs will allow to better understand the underlying behavior of malicious callers in the network. As a future work, call data records from operators should be studied to identify statistical properties of callers. This study will help to design effective strategies to combat spam over web communication services. Mechanisms should also be introduced to combat instant messaging spam where unsolicited instant messages are sent over the network.

Bibliography

[3GP] 3GPP. Web real-time communications (webrtc) access to the ip multimedia (im) core network (cn) subsystem (ims) 3gpp ts 24.371.

[AB02] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74:47–97, Jan 2002.

[AEG+10] S. Adali, R. Escriva, M. K. Goldberg, M. Hayvanovych, M. Magdon-Ismail, B. K. Szymanski, W. A. Wallace, and G. Williams. Measuring behavioral trust in social networks. In *2010 IEEE International Conference on Intelligence and Security Informatics*, pages 150–152, May 2010.

[AM13] M.A. Azad and R. Morla. Caller-rep: Detecting unwanted calls with caller social strength. *Comput. Secur.*, 39:219–236, nov 2013.

[BA13] A. Nennker A. Narayanan B. Adida, T. Kuijsten. Browserid [online], available https://github.com/mozilla/id-specs/blob/prod/browserid/index.md. 2013.

[BAP07] V. A. Balasubramaniyan, Mustaque Ahamad, and Haesun Park. Callrank: Combating spit using call duration, social networks and global reputation. In *Fourth Conference on eamil and anti-spam (CEAS 2007)*, 2007.

[BB15] Victoria Beltran and Emmanuel Bertin. Unified communications as a service and webrtc: An identity-centric perspective. *Computer Communications*, 68:73 – 82, 2015.

[BBC14] V. Beltran, E. Bertin, and N. Crespi. User identity for webrtc services: A matter of trust. *IEEE Internet Computing*, 18(6):18–25, Nov 2014.

[BBC+15a] 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)*, pages 208–215, Feb 2015.

[BBC15b] V. Beltran, E. Bertin, and S. Cazeaux. Additional Usecases and Requirements for WebRTC Identity Architecture. Technical report, March 2015.

[BBJ+16] Adam Bergkvist, Daniel C. Burnett, Cullen Jennings, Anant Narayanan, and Bernard Aboba. WebRTC 1.0: Realtime Communication Between Browsers. Technical report, May 2016.

[BCT+13] 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*, pages 46–52, Oct 2013.

[BdMM] J. Bradley, B. de Medeir, and C. Mortimore. Openid connect core 1.0. *The OpenID Foundation*.

[BDP04] William E Burr, Donna F Dodson, and William T Polk. *Electronic authentication guideline*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology, 2004.

[BSG+11] 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*, pages 41–45, April 2011.

[CC02] Keith S. Coulter and Robin A. Coulter. Determinants of trust in a service provider: the moderating role of length of

relationship. *Journal of Services Marketing*, 16(1):35–50, 2002.

[CCFJ16] R. Copeland, K. Corre, I. Friese, and S. El Jaouhari. Requirements for Trust and Privacy in WebRTC Peer-to-peer Authentication. Technical report, September 2016.

[CHD05] Elizabeth Chang, Farookh Hussain, and Tharam Dillon. *Trust and Reputation for Service-Oriented Environments: Technologies For Building Business Intelligence And Consumer Confidence*. John Wiley & Sons, Inc., USA, 2005.

[Cla13] R. Clarke. Introduction to dataveillance and information privacy, and definitions of terms. *Roger Clarke's Dataveillance and Information Privacy Pages*, 2013.

[COBY11] 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*, pages 451–456, July 2011.

[Deu73] Morton Deutsch. The resolution of conflict: Constructive and destructive processes. *American Behavioral Scientist*, 17(2):248–248, 1973.

[DGSJ+16] Willem De Groef, Deepak Subramanian, Martin Johns, Frank Piessens, and Lieven Desmet. Ensuring endpoint authenticity in webrtc peer-to-peer communication. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, SAC '16, pages 2103–2110, New York, NY, USA, 2016. ACM.

[EBA95] Ebay e-commerce company, [online] https://www.ebay.fr/. Founded: September 3, 1995.

[FAC04] Facebook - online social networking website, [online] https://www.facebook.com/. Founded: February 2004.

[FKS14] Daniel Fett, Ralf Küsters, and Guido Schmitz. An expressive model for the web infrastructure: Definition and application to the browser id sso system. In *Proceedings of the 2014 IEEE Symposium on Security and Privacy*, SP '14, pages 673–688, Washington, DC, USA, 2014. IEEE Computer Society.

[Gol05a] Jennifer Golbeck. Personalizing Applications through Integration of Inferred Trust Values in Semantic Web-based Social Networks. In *Semantic Network Analysis Workshop at the 4th International Semantic Web Conference*, November 2005.

[Gol05b] Jennifer Ann Golbeck. *Computing and Applying Trust in Web-based Social Networks*. PhD thesis, College Park, MD, USA, 2005. AAI3178583.

[GPH03] Jennifer Golbeck, Bijan Parsia, and James Hendler. Trust networks on the semantic web. In Matthias Klusch, Andrea Omicini, Sascha Ossowski, and Heimo Laamanen, editors, *Cooperative Information Agents VII*, pages 238–249, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[GS00] T. Grandison and M. Sloman. A survey of trust in internet applications. *IEEE Communications Surveys Tutorials*, 3(4):2–16, Fourth 2000.

[Har] D. Hardt. The OAuth 2.0 Authorization Framework. Technical report.

[HWS09] Chung-Wei Hang, Yonghong Wang, and Munindar P. Singh. Operators for propagating trust and their evaluation in social networks. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '09, pages 1025–1032, Richland, SC, 2009. International Foundation for Autonomous Agents and Multiagent Systems.

[IRCaK+17] I.Friese, S.Gondor R. Copeland, F. Beierle andA. Kupper, R. L. Pereira, and J. M. Crom. Cross-domain discovery of

communication peers identity mapping and discovery services. In *European Conference on Networks and Communications (EUCNC) 2017*, pages 451–456, June 2017.

[JB12] A. Johnston and D. Burnett. *WebRTC: APIs and RTCWEB protocols of the HTML5 real-time web*. Digital Codex LLC, 2012.

[JCC+16] Ibrahim Tariq Javed, Rebecca Copeland, Noël Crespi, Felix Beierle, Sebastian Göndör, Axel Küpper, Ahmed Bouabdallah, Marc Emmelmann, Andreea Ancuta Corici, Kevin Corre, Jean-Michel Crom, Frank Oberle, Ingo Friese, Ana Caldeira, Gil Dias, Ricardo Chaves, and Nuno Santos. Global Identity and Reachability Framework for Interoperable P2P Communication Services. In *ICIN 2016 : conference on Innovations in Clouds, Internet and Networks*, pages 59 – 66, Paris, France, March 2016. IFIP Open Digital Library.

[JFH+05] Audun Jøsang, John Fabre, Brian Hay, James Dalziel, and Simon Pope. Trust requirements in identity management. In *Proceedings of the 2005 Australasian Workshop on Grid Computing and e-Research - Volume 44*, ACSW Frontiers '05, pages 99–108, Darlinghurst, Australia, Australia, 2005. Australian Computer Society, Inc.

[JG05] Devon Johnson and Kent Grayson. Cognitive and affective trust in service relationships. *Journal of Business Research*, 58(4):500 – 507, 2005. Special Section: Attitude and Affect.

[JHW13] C. Jennings, T. Hardie, and M. Westerlund. Real-time communications for the web. *IEEE Communications Magazine*, 51(4):20–26, April 2013.

[JØs98] Audun JØsang. A subjective metric of authentication. In Jean-Jacques Quisquater, Yves Deswarte, Catherine Meadows, and Dieter Gollmann, editors, *Computer Security — ESORICS 98*, pages 329–344, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

[KD07] Prakash Kolan and Ram Dantu. Socio-technical defense against voice spamming. *ACM Trans. Auton. Adapt. Syst.*, 2(1), March 2007.

[Ker12] A. Keromytis. A comprehensive survey of voice over ip security research. *IEEE Communications Surveys Tutorials*, 14(2):514–537, Second 2012.

[KG07] Ugur Kuter and Jennifer Golbeck. Sunny: A new algorithm for trust inference in social networks using probabilistic confidence models. In *Proceedings of the 22Nd National Conference on Artificial Intelligence - Volume 2*, AAAI'07, pages 1377–1382. AAAI Press, 2007.

[KKKJ09] Hyung-Jong Kim, Myuhng Joo Kim, Yoonjeong Kim, and Hyun Cheol Jeong. Devs-based modeling of voip spam callers? behavior for spit level calculation. *Simulation Modelling Practice and Theory*, 17(4):569 – 584, 2009.

[KP09] Stylianos Karapantazis and Fotini-Niovi Pavlidou. Voip: A comprehensive survey on a promising technology. *Computer Networks*, 53(12):2050 – 2090, 2009.

[KSGM03] 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*, WWW '03, pages 640–651, New York, NY, USA, 2003. ACM.

[LCQC14] L. Li, W. Chou, Z. Qiu, and T. Cai. Who is calling which page on the web? *IEEE Internet Computing*, 18(6):26–33, Nov 2014.

[Lei12] B. Leiba. Oauth web authorization protocol. *IEEE Internet Computing*, 16(1):74–77, Jan 2012.

[LFGG+14] L. López-Fernández, M. Gallego, B. García, D. Fernández-López, and F. J. López. Authentication, authorization, and accounting in webrtc paas infrastructures: The case

of kurento. *IEEE Internet Computing*, 18(6):34–40, Nov 2014.

[LR12] S. Loreto and S. P. Romano. Real-time communications in the web: Issues, achievements, and ongoing standardization efforts. *IEEE Internet Computing*, 16(5):68–73, Sept 2012.

[Lyn11] L. Lynch. Inside the identity management game. *IEEE Internet Computing*, 15(5):78–82, Sept 2011.

[MACL10] P. Melo, L. Akoglu, C.Faloutsos, and A. Loureiro. *Surprising Patterns for the Call Duration Distribution of Mobile Phone Users*, pages 354–369. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[MDS95] Roger C. Mayer, James H. Davis, and F. David Schoorman. An integrative model of organizational trust. *The Academy of Management Review*, 20(3):709–734, 1995.

[MGM06] Sergio Marti and Hector Garcia-Molina. Taxonomy of trust: Categorizing {P2P} reputation systems. *Computer Networks*, 50(4):472 – 484, 2006. Management in Peer-to-Peer Systems.

[MP09a] A. J. McLeod and S. E. Pippin. Security and privacy trust in e-government: Understanding system and relationship trust antecedents. In *2009 42nd Hawaii International Conference on System Sciences*, pages 1–10, Jan 2009.

[MP09b] Félix Gómez Mármol and Gregorio Martínez Pérez. Security threats scenarios in trust and reputation models for distributed systems. *Computers & Security*, 28(7):545 – 556, 2009.

[MR] D. McGrew and E. Rescorla. Datagram Transport Layer Security (DTLS) Extension to Establish Keys for the Secure Real-time Transport Protocol (SRTP). Technical report.

[NGD+06] Amit A. Nanavati, Siva Gurumurthy, Gautam Das, Dipanjan Chakraborty, Koustuv Dasgupta, Sougata Mukherjea,

and Anupam 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*, pages 435–444, 2006.

[NSP11] 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)*, pages 841–846, Nov 2011.

[OXF84] Oxford dictionary publisher: Oxford university press, [online] http://www.askoxford.com/. Originally published: February 1, 1884.

[PBMW99] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[Per] Persistence market research. (2016, may). voip services market.

[PM03] A. Pashalidis and C. J. Mitchell. A taxonomy of single sign-on systems. In *Information security and privacy*, pages 249–264. Springer, 2003.

[PT10] A. Pfitzmann and H. Tschofenig. Terminology for Talking about Privacy by Data Minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management. Technical report, July 2010.

[QNTS08] J. Quittek, S. Niccolini, S. Tartarelli, and R. Schlegel. On spam over internet telephony (spit) prevention. *IEEE Communications Magazine*, 46(8):80–86, August 2008.

[QvdMP07] Vincent M. Quinten, Remco van de Meent, and Aiko Pras. *Analysis of Techniques for Protection Against Spam over Internet Telephony*, pages 70–77. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[RC04] I. Ray and S. Chakraborty. A vector model of trust for developing trustworthy systems. In *Computer Security–ESORICS 2004*, pages 260–275. Springer, 2004.

[Res15] E. Rescorla. Security Considerations for WebRTC. Technical report, February 2015.

[Res16] E. Rescorla. WebRTC Security Architecture. Standards Track, June 2016.

[RETa] Deliverable 1.1: Use cases and sustainable business models for rethink, [online] https://https://bscw.rethink-project.eu/pub/bscw.cgi/d30179/d1.1 Horizon 2020 RETHINK Project.

[RETb] Deliverable 2.1: Framework architecture definition, [online] https://bscw.rethink-project.eu/pub/bscw.cgi/d30184/d2.1organization=.

[retc] rethink project trustful hyper-linked entities in dynamic networks, [online] https://rethink-project.eu/. Horizon 2020 Research Project grant agreement n° 645342.

[RTC] Real-time communication in web-browsers (active wg), [online] http://tools.ietf.org/wg/rtcweb. IETF.

[SAS06] D. Shin, J. Ahn, and C. Shim. Progressive multi gray-leveling: a voice spam protection algorithm. *IEEE Network*, 20(5):18–24, Sept 2006.

[SDS10] F. Sabena, A. Dehghantanha, and A. P. Seddon. A review of vulnerabilities in identity management using biometrics. In *2010 Second International Conference on Future Networks*, pages 42–49, Jan 2010.

[SG09] Amardeo C. Sarma and João Girão. Identities in the future internet of things. *Wireless Personal Communications*, 49(3):353–363, May 2009.

[SH02] Bomil Suh and Ingoo Han. Effect of trust on customer acceptance of internet banking. *Electronic Commerce Research and Applications*, 1(3):247 – 263, 2002.

[SHZK05] S. Song, K. Hwang, R. Zhou, and Y. K. Kwok. Trusted p2p transactions with fuzzy reputation aggregation. *IEEE Internet Computing*, 9(6):24–34, Nov 2005.

[SKY03] Skype, communication tool for free calls and chat, [online] https://skype.com/. Initial release date: August 2003.

[SWN12] H. Sengar, X. Wang, and A. Nichols. *Call Behavioral Analysis to Thwart SPIT Attacks on VoIP Networks*, pages 501–510. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[TACM12] K. Toumi, C. Andrés, A. Cavalli, and M. EL Maarabani. A vector based model approach for defining trust in multi-organization environments. In *7th Int. Conf. on Risks and Security of Internet and Systems, CRISIS'12*, page in press. IEEE Computer Society Press, 2012.

[UKs17] GfK UKs. Nuisance calls research 2017, [online] https://www.ofcom.org.uk/research-and-data/telecoms-research/nuisance-calls/nuisance-calls-research-2017. Ofcom Online Report, Jan/Feb 2017.

[VCCdS09] Patricia Victor, Chris Cornelis, Martine De Cock, and Paulo Pinheiro da Silva. Gradual trust and distrust in recommender systems. *Fuzzy Sets and Systems*, 160(10):1367 – 1382, 2009. Special Issue: Fuzzy Sets in Interdisciplinary Perception and Intelligence.

[VIB10] Viber - instant messaging and voice over ip application, [online] https://www.viber.com/. Initial release: December 2, 2010;.

[Weba] A study of webrtc security, [online] http://webrtc-security.github.io/. NTT Communications project.

[WEBb] Web real-time communications working group, [online] http://www.w3.org/2011/04/webrtc. W3C.

[WEC11] Wechat - multi-purpose messaging, social media and mobile payment app developed by tencent, [online]

https://web.wechat.com/. Initial release date: January 21, 2011.

[WHA09] Whatsapp messenger, freeware and cross-platform messaging and voice over ip service owned by facebook, [online] https://www.whatsapp.com/. Founded in February 24, 2009.

[WL08a] Y. Wang and E. P. Lim. The evaluation of situational transaction trust in e-service environments. In *2008 IEEE International Conference on e-Business Engineering*, pages 265–272, Oct 2008.

[WL08b] Y. Wang and K. J. Lin. Reputation-oriented trustworthy computing in e-commerce environments. *IEEE Internet Computing*, 12(4):55–59, July 2008.

[WL10] T. Wang and Y. Lu. Determinants of trust in e-government. In *2010 International Conference on Computational Intelligence and Software Engineering*, pages 1–4, Dec 2010.

[WS14] E. Ware and J. Surtees. Got their number ending the harm caused by nuisance calls and text, [online] https://www.stepchange.org/portals/0/documents/media/report. StepChange Online Report, 2014.

[XL04] Li Xiong and Ling Liu. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Trans. on Knowl. and Data Eng.*, 16(7):843–857, July 2004.

[ZH07] 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*, 18(4):460–473, 2007.

[ZM00] Giorgos Zacharia and Pattie Maes. Trust management through reputation mechanisms. *Applied Artificial Intelligence*, 14(9):881–907, 2000.