Aalto University
School of Science and Technology
Faculty of Information and Natural Sciences
Master’s Degree Programme in Security and Mobile Computing

Yu Chen

A Privacy-enhanced Reputation System for Mobile Ad hoc Services

Master’s Thesis
Espoo, June 28, 2010

Supervisors: Professor Antti Ylä-Jääski, Aalto University
Professor Øivind Kure, The Norwegian University of Science and Technology
Instructor: Dr. Zheng Yan, Nokia Research Center, Helsinki
Aalto University
School of Science and Technology
Faculty of Information and Natural Sciences
Degree Programme of Security and Mobile Computing

Author: Yu Chen
Title of thesis: A Privacy-enhanced Reputation System for Mobile Ad hoc Services

Date: June 28, 2010
Pages: 7 + 62

Professorship: Data Communications Software
Code: T-110

Supervisors: Professor Antti Ylä-Jääski
Professor Øivind Kure
Instructor: Dr. Zheng Yan

Mobile ad hoc networks (MANETs) are new paradigm of networks offering unrestricted mobility without any underlying infrastructure. Nowadays, MANETs are becoming a promising platform for pervasive social networking and content services. In such services, various information flows over the network. In order to evaluate information trustworthiness, this thesis presents a reputation system to evaluate trust/reputation of entities in MANET services, thus assisting user decisions.

We propose AdRep – a reputation system with a hybrid trust framework that can support trust/reputation evaluation by both MANET nodes and a trusted server for different contexts and scenarios. Thus ephemeral node experiences and/or historical node behaviors can be aggregated to provide more accurate trust information in the context of frequent change of node pseudonyms for privacy enhancement. We further implement two contextual reputation systems – AdChatRep and AdContRep, by applying AdRep in two MANET services – pervasive social chatting and content recommendation. Our simulation results show AdContRep’s effectiveness with regard to unfair rating attacks and on-off attacks. Our two-stage user study evaluates AdChatRep’s usefulness, effectiveness and easy acceptance for mobile users.

Keywords: Mobile ad hoc networks, MANET, trust, reputation, privacy

Language: English
Acknowledgements

First and foremost I would like to thank my thesis supervisors, Professor Antti Ylä-Jääski and Professor Øivind Kure. The thesis work would not have been possible without their support, guidance and valuable comments. I owe my deepest gratitude to my thesis instructor Dr. Zheng Yan for providing me the topic, her shining ideas and patient teaching. Without her help, I could not have learnt so much during thesis work.

It is an honor for me to appreciate my colleagues in Nokia who have helped me: N. Asokan, Philip Ginzboorg, Kari Leppanen, Mikko Tirronen, Jan-Erik Ekberg, Pekka Laitinen and Miaoqing Tan. It is also a valuable experience to work with them and learn from their brilliant thinking. I am also grateful to all participants in the user study for their precious time, suggestions, comments and encouragement.

I sincerely thank my dearest friend Lan Huang who carried out the ‘Graduation Countdown Log’ Plan together with me. Her remote spiritual support has helped me remove all obstacles. My sister Yu Su has been constantly caring me during these two years. My flatmates Sanja Scepanovic and Heein Song have been accompanying me and bringing so much sunshine to my work and life. My friends as well as classmates Tran Thi Thuy Trang and Jiawei Chen have walked me through difficulties during thesis writing. Kiitos!

Last but not least, I thank my parents, my brother and my sister-in-law for their eternal love. Thanks to life for all the joy and pain, happiness and sorrow, opportunities and difficulties in these two years. They are my greatest memories and treasure.

Espoo June 28th 2010

Yu Chen
# Abbreviations and Acronyms

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MANET</td>
<td>Mobile Ad hoc Network</td>
</tr>
<tr>
<td>ID</td>
<td>Identity</td>
</tr>
<tr>
<td>PDA</td>
<td>Personal Digital Assistant</td>
</tr>
<tr>
<td>TS</td>
<td>Trusted Server</td>
</tr>
<tr>
<td>UID</td>
<td>User identity</td>
</tr>
<tr>
<td>PID</td>
<td>Permanent identity</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
<tr>
<td>AdRep</td>
<td>A reputation system for Ad hoc services</td>
</tr>
<tr>
<td>AdChatRep</td>
<td>A reputation system for Ad hoc chatting services</td>
</tr>
<tr>
<td>AdContRep</td>
<td>A reputation system for Ad hoc content services</td>
</tr>
</tbody>
</table>
# Contents

Abbreviations and Acronyms iv

1 Introduction 1
   1.1 Motivations ........................................... 1
   1.2 Terminologies ......................................... 2
   1.3 Problem Description .................................. 2
   1.4 Thesis Work ........................................... 3
   1.5 Organization .......................................... 4

2 Related Work 5
   2.1 Academic Work ........................................ 5
   2.2 Practical Work ....................................... 8
      2.2.1 Chatting Services ................................. 8
      2.2.2 Recommendation Services ......................... 9

3 System Design 10
   3.1 MANET Services ....................................... 10
      3.1.1 MANET Chatting Service .......................... 11
      3.1.2 MANET Content Recommendation Service ........ 12
   3.2 Assumptions and System Requirements ............... 12
      3.2.1 Assumptions Related to MANET Services .......... 12
      3.2.2 Assumptions Related to a Trusted Server ........ 13
      3.2.3 Requirements on System Design ................. 14
3.3 A Hybrid Reputation Management Framework . . . . . . . . . . . . . . . . . . . . . . . 15
  3.3.1 Decentralized Reputation Framework . . . . . . . . . . . . . . . . . . . . . . . 15
  3.3.2 Centralized Reputation Framework . . . . . . . . . . . . . . . . . . . . . . . . 15
  3.3.3 A Hybrid Reputation Management Framework . . . . . . . . . . . . . . . . . . . 15
3.4 Trust/reputation Evaluation Model . . . . . . . . . . . . . . . . . . . . . . . . . . 20
  3.4.1 Local Trust Evaluation on Mobile Devices . . . . . . . . . . . . . . . . . . . . 20
  3.4.2 Global Reputation Evaluation on TS . . . . . . . . . . . . . . . . . . . . . . . 21

4 Implementation and Prototypes 22
  4.1 AdChatRep: A Reputation System for MANET Chatting . . . . . . . . . . . . . . . . 22
    4.1.1 Implementation Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . 22
    4.1.2 System Functionalities . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
    4.1.3 Implementation and Prototype . . . . . . . . . . . . . . . . . . . . . . . . . . 26
  4.2 AdContRep: A Reputation System for MANET Content Recommendations . . . . . . . 34
    4.2.1 Implementation Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . 34
    4.2.2 System Functionalities . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 35
    4.2.3 Implementation and Prototype . . . . . . . . . . . . . . . . . . . . . . . . . . 37

5 Evaluation 42
  5.1 Evaluation on AdChatRep . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 42
    5.1.1 Evaluation Design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 42
    5.1.2 Results and Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 46
  5.2 Evaluation on AdContRep . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48
    5.2.1 Evaluation Design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48
    5.2.2 Evaluation Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51

6 Conclusions 55
  6.1 Contributions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55
  6.2 Discussions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 56
  6.3 Future Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 57
List of Tables

4.1 System implementation requirements in AdChatRep ........ 23
4.2 Differences among PIDs,UIDs and nick names in AdChatRep 26
4.3 System implementation requirement in AdContRep .......... 35
4.4 Differences between PID and UID in AdContRep .......... 37

5.1 Mean of rating on AdChatRep usefulness in Round 1 and Round 2 ........................................ 46
5.2 Mean of rating on AdChatRep effectiveness in Round 1 and Round 2 .......................................... 47
5.3 Mean of rating on reputation visualization style of AdChatRep in Round 1 and Round 2 .................... 48
5.4 Analysis on chatting log in Round 2 .......................... 48
List of Figures

3.1 Illustration of the trust/reputation framework of AdRep .... 18

4.1 Detected nearby users in AdChatRep .......................... 27
4.2 An example UI of user profile in AdChatRep .................. 27
4.3 An example UI of updated global reputation from the TS in AdChatRep ........................................ 28
4.4 An example UI of creating a new chatting community in AdChatRep ........................................ 28
4.5 An example UI of joining a community in AdChatRep .... 29
4.6 Chatting UI with on-chat voting in AdChatRep ............ 29
4.7 On-chat voting UI in AdChatRep ............................... 30
4.8 UI of the user list for afterwards voting in AdChatRep .... 30
4.9 Afterwards voting UI in AdChatRep ............................ 31
4.10 Peer user profile UI in AdChatRep ............................ 31
4.11 UI of detailed local trust information of peer users in AdChatRep ........................................ 32
4.12 Local trust visualization in AdChatRep ........................ 32
4.13 An example of user presence history in AdChatRep ........ 33
4.14 An example of changing UID in AdChatRep ............... 33
4.15 Local data management UI in AdContRep ................... 38
4.16 Detailed user information in AdContRep ..................... 39
4.17 An example of content recommendation query in AdContRep . 39
4.18 An example of content recommendation reply in AdContRep . 40
4.19 An example of feedback categories in AdContRep ........ 40
4.20 An example of content quality feedback in AdContRep .... 41
5.1 Content reputations at a query node: (a) each recommender node votes one content honestly; (b) each recommender node votes all contents honestly; (c) 5 malicious nodes vote one content dishonestly; (d) 5 malicious nodes vote all contents dishonestly. 52

5.2 Content reputations at TS: (a) each recommender node votes one content honestly; (b) each recommender node votes all contents honestly; (c) 5 malicious nodes vote one content dishonestly; (d) 5 malicious nodes vote all contents dishonestly. 53

5.3 F measure of nodes and TS regarding malicious node detection performance: (a) unfair rating attack ($thr = 30$); (b) on-off attack ($thr = 5$). 54
Chapter 1

Introduction

1.1 Motivations

Mobile ad hoc networks (MANETs) are new paradigm of networks offering unrestricted mobility without any underlying infrastructure. Nowadays, MANETs are becoming a promising platform for pervasive social networking and content services that could greatly benefit mobile users due to connection convenience and free cost. Users in MANETs communicate with each other in a decentralized and self-organized manner. For example, a MANET user could query people in the vicinity about which movie is worth watching in front of a cinema. She/he can also chat with people nearby using his/her mobile device about who would like to share the cost of riding a taxi to a same place together. In such situations, strangers could likely be socially connected due to physical closeness. On the other hand, socially connected people could be aware when they are physically close to each other.

Although the above MANET services are promising, we are facing an important issue. Obviously, various content information flows in such a pervasive social networking system. Examples are a mobile application installation link, a URL of a service, a web page and a textual message typed by a user. How much should a user trust different content information received over the MANET? Meanwhile, in order to preserve personal privacy and avoid malicious tracking, the user generally would like to use an anonymous identity (ID) or pseudonym in the content recommendation query/response and MANET chatting. To solve this problem, we propose AdRep - a privacy enhanced reputation system for MANET services. The system evaluates trust/reputation of entities in MANET services, thus helping mobile users to
CHAPTER 1. INTRODUCTION

make suitable decision for information selection and consumption.

1.2 Terminologies

MANETs are self-organized networks consisting of mobile nodes that rely on no fixed infrastructure [28]. In MANETs, each mobile node is both a host and a router [22]. An important characteristic of MANET node is its limitation in data transmission range. Thus, in long-distance transmission process, source and destination nodes establish connectivity by a number of MANET nodes forming a multi-hop route. MANETs are therefore becoming a promising platform for many services due to no requirement for setting up and maintaining fixed networks [28]. Mobile nodes are entities that vary from service to service. We have different names for MANET node for each type of service.

Reputation, as defined verbally by Oxford dictionary [12], is the opinion that people in general have about an entity. This implies that reputation is generated based on a certain number of people’s opinions. The opinions could be generated based on personal experiences, other people’s comments, and so on, mostly with regard to the target entity’s popularity and quality. Reputation is dynamically evolved according to newly accumulated experiences and comments. It is subjective since one entity’s reputation could be different for different people. Reputation could help users to make proper decisions in numerous services. Higher reputation indicates higher level of trustworthiness, thus motivate more social activities.

Trust, as defined by Rousseau [34], is “a psychological state comprising the intention to accept vulnerability based upon positive expectations of the intentions or behavior of another”. The definition give us the hint that trust underlies many situational variables [35]. The meaning of trust in different contexts should not always be the same. We apply trust model to evaluate level of trust (trustworthiness) in concrete contexts or scenarios.

1.3 Problem Description

In order to design a privacy enhanced reputation system for MANET services, we need to solve a number of problems as described below:
• Which reputation system architecture is appropriate for MANET services with user privacy preservation?

• Which trust model is practical for such a system in order to enhance privacy?

• Which trust/reputation evaluation algorithms are effective and robust, thus can be applied in the system?

Regarding reputation system architecture, both centralized and decentralized structures can be found in literature [24]. We aim to make use of the advantages of both in our design. Meanwhile, we also need to consider some practical reasons raised by industrial business expectations.

Furthermore, we need to design a suitable trust model based on the designed architecture. This model should be flexibly and commonly used to evaluate trust/reputation of various entities related to MANET services.

Privacy preservation is a very important issue in MANET services. Users tend to leak information of their personal data and private information (e.g., location information). The reputation system should take user privacy preservation into account.

The reputation system should also be effective and robust against major attacks. Reputation algorithms embedded into the system should be effective to evaluate trust/reputation. The system should also be robust against major attacks, such as unfair rating attacks and on-off attacks [39]. Finally, we hope the reputation system is useful and helpful for mobile users.

1.4 Thesis Work

This thesis presents AdRep, a reputation system with hybrid reputation management framework that can be applied into MANETs to evaluate trust/reputation of entities by both MANET nodes and a trusted server, thus helping mobile users to make suitable decision for content selection and consumption. The trust/reputation evaluation model in AdRep supports numerous contexts for various MANET services. Thus ephemeral node experiences and/or historical node behaviors can be aggregated together to provide more accurate trust information in the context of frequent change of node pseudonyms for privacy enhancement. Based on AdRep, we implement two concrete reputation
systems – AdContRep and AdChatRep – to support two different services in MANETs: content recommendation service and pervasive social chatting service. Our simulation results show AdContRep’s effectiveness with regard to unfair rating attack and on-off attack. Our two-stage user study evaluates AdChatRep’s usefulness, effectiveness and easy acceptance for mobile users.

The main tasks of the thesis are:

- Design a reputation system (AdRep) that commonly applies in various MANET services;
- Apply AdRep in MANET chatting and MANET content recommendation and implement two concrete reputation systems: AdChatRep and AdContRep;
- Evaluate AdChatRep’s usefulness, effectiveness and UI design acceptance by user study;
- Evaluate the effectiveness and robustness of AdContRep reputation algorithms through simulations (using Matlab).

1.5 Organization

The rest of the thesis is organized as follows. Chapter 2 briefly reviews related work. In Chapter 3, we present AdRep – a hybrid reputation system for situational contexts. Chapter 4 describes the design and implementation of AdContRep and AdChatRep, followed by simulation results and user study results in Chapter 5. We conclude the thesis by summarizing the author’s contributions and discussing additional issues and future work.
Chapter 2

Related Work

2.1 Academic Work

Jøsang et al. classified reputation system architecture into two main types: centralized and distributed [24]. The system architecture determines how ratings and reputation scores are communicated between participants in a reputation system. In the literature, distributed trust evaluations have been studied in MANET, seldom the solutions support node privacy [40, 41, 31]. This could cause such potential attacks as bad mouthing attacks or unfair rating attacks targeting at a specific node [38]. Most existing systems maintain a statistical representation of reputation by borrowing tools from the realms of game theory (Confidant [18] and Core [29], and peer-to-peer networks [44]) and Bayesian analytics [17]. These systems try to counter selfish routing misbehavior of nodes by enforcing nodes to cooperate with each other and counter any arbitrary misbehavior of nodes. However, little work has paid attention to the content reputation issue in MANET with node privacy as a main concern. On the other hand, practical reputation systems generally apply a centralized server to collect feedback for reputation generation (e.g. eBay [33], Yahoo auctions [32], and Internet-based systems such as Keynote [16]). However, many existing systems (e.g. Amazon, eBay) lack considerations on the credibility of a user’s rating. This greatly influences the quality of produced reputations. The usage of pseudonym and the ease of its change additionally complicate the picture by allowing participants to effectively erase their prior history.

AdChatRep and AdContRep adopt a hybrid reputation system architecture, where trust and reputation values are evaluated in a distributed way, but
with the support of a centralized trusted server.

In the literature, trust and reputation mechanisms have been widely studied in various fields of distributed systems, such as ad hoc networks, peer-to-peer (P2P) systems, Grid computing, pervasive computing and e-commerce [46]. Many mechanisms have been developed for supporting trusted communications and collaborations among computing nodes [40, 41, 48, 26, 25]. Examples are FuzzyTrust system [36], the eBay user feedback system [33], PeerTrust model [45], an objective trust management framework (OTMF) for MANET [25] and Credence - a robust and decentralized system for evaluating the reputation of files in a P2P system [42]. In these researches, trust can be modeled, calculated and thus expressed using a value. However, none of the above studies consider how to support privacy. Thus it is hard to directly apply them into AdChatRep and AdContRep.

Recently, a number of reputation systems have been proposed in the context of digital contents and social networking [13]. For example, Thomas Adler and Alfaro proposed a content driven reputation system for Wikipedia authors solely on the basis of content evolution; but not on user-to-user comments or ratings [13]. The concept of data centric trust in volatile environments, such as ad hoc networks, was introduced in [31] to evaluate the node trust through the data reported by it. Gupta, et al. proposed a partially distributed reputation system for P2P systems by introducing a reputation computation agent (RCA). Its system structure is similar to AdChatRep and AdContRep. But this system does not concern the challenges caused by privacy enhancement. In addition, the RCA is applied only for calculating peer’s reputation based on its contributions to the system. In AdChatRep and AdContRep, we apply the trusted server to update both node trust and content reputation on the basis of long-term historical social behaviors.

In most reputation systems in the ad hoc networks, the reputation of a node is shared globally in the network. The purpose is to make the reputation of a node known to all other nodes and decrease the detection time. Thus maintaining and disseminating indirect reputation information incur overhead at both the individual node and the network. OCEAN [15] discounts second-hand reputation exchange and only utilizes local reputation based on direct observations in order to achieve a reasonable performance. AdChatRep and AdContRep concern both local-aware and global-aware reputations by aggregating local experiences and global experiences together. By deploying the trusted server, the overhead of reputation maintenance and dissemination is
CHAPTER 2. RELATED WORK

eliminated among MANET nodes.

Inconsistent reputation problem (i.e., different nodes may have different reputation values for the same node) often occurs in the ad hoc networks due to subjective reason and/or different local experiences. This makes it hard to distinguish correct reputation ratings from reputation voting messages. LARS (Locally Aware Reputation System) [23] was proposed to deal with selfish behaviors and malicious behaviors (e.g., packet dropping and unfair rating). In LARS, the reputation of a node is derived from direct observation and exchange of second hand reputation information is disallowed. In AdChatRep and AdContRep, we apply the trusted server to uniform node recommendation trust based on local experiences reported by nodes. This trust/reputation information is issued to the node by the server. Serving as the initial value of trust, it is further evolved based on new experiences collected at the individual node. In addition, the above process is iterated. Thereby, we avoid the inconsistent reputation problem and eliminate node trust inaccuracy caused by multi-hop reputation dissemination. Trust/reputation evaluation is based on first-hand experiences and direct votes no matter at the TS or the node.

A reputation rating system based on past behavior of evaluators was proposed in [21]. Trust in the evaluator indexes its impact on the rating system. The trust value is dynamically adjusted based on past estimation performance. In AdChatRep and AdContRep, the node trust is adjusted based on its past social performance. It is evaluated by each individual node and at the TS based on ephemeral and historical experiences, respectively.

Nowadays, reputation systems may face the problem of unfair ratings by artificially inflating or deflating reputations [33, 32, 19, 49]. They are vulnerable to a number of potential attacks, such as Sybil attack, on-off attack, independent/collaborative bad mouthing attack, and conflict behavior attack [20, 27]. The usage of pseudonyms introduces new challenges since it makes hard to trace malicious behaviors. It also influences the accuracy of reputation. Sun et al. proposed a number of schemes to overcome some of the above attacks, but they did not consider the additional challenges caused by privacy preservation [38, 39]. AdContRep aims to overcome a couple of traditional attacks, such as the unfair rating attacks and on-off attacks.

A lot of work has been conducted regarding user interface design in order to improve user’s trust, mainly for web sites and in the context of e-commerce.
CHAPTER 2. RELATED WORK

Still, prior art left rooms for further studies on the effects of trust information on social networking and, in particular, on how to provide trust information for mobile users. In AdChatRep, we use a reputation indicator to indicate each user’s local reputation during chatting and provide detailed information about local reputation generation and global reputation. They are interface design elements that provide the cue of trust information in MANET chatting services [47]. Particularly, a user’s local reputation value could play as valuable credibility of the voting on the user. But few previous researches investigated visualizing reputation’s effects on mobile users in the context of pervasive social networking, which is one of our research targets in AdChatRep.

2.2 Practical Work

There are many existing projects related to our work. Herein, we give a brief review.

2.2.1 Chatting Services

AdChatRep is similar to a number of research projects regarding social networking functions, such as Micro-blog [7] and AdSocial [37]. However, trust and reputation in social networking are not considered in these projects.

Micro-blog by Duke University also supports query, chatting and recommendation in ad-hoc networks. Posts about different contents are floating on the map of the application user interface. Whenever a user travels, posts about the location are floated to the user. If there is not enough information, users can choose to query in ad-hoc networks, and replies could be added to the current location. It applies in many areas such as tourism, advertisement, emergent alerts and etc.

AdSocial is a social network based on Ad-hoc networks. Important functions include Presence Detector, Buddy Search and chatting, VoIP calls, video calls and ad-hoc games. It supports real time communication with friends. Examples of use cases could be chatting in the bar or on the train. Besides that, AdSocial supports multiple message exchange methods, e.g. Wi-fi and Bluetooth.
CHAPTER 2. RELATED WORK

More specifically, we use ride sharing as a use case of AdChatRep. [7] and [37] introduce a dynamic taxi-sharing service using intelligent transportation in Taipei city. Parties involved in the service include customers who want to take a taxi, dispatcher and the taxi driver. In order to find the suitable people to share a ride, customers need response from the dispatcher system, with less customer-tailored options. AdChatRep assists users to make sharing decision locally with mobile phones.

2.2.2 Recommendation Services

Advogato [1] is an online service to provide a platform for free software developers to advocate softwares and promote research. One significant impact of Advogato is its trust metrics behind the service, which is the basis of many research projects [2]. Advogato trust metric stimulates users to contribute quality ensured softwares and protects against attacks. In AdContRep, user trust is evaluated based on the user’s recommendation performance. However, Advogato only applies a centralized architecture for reputation generation, whilst AdChatRep and AdContRep adopt hybrid reputation system architecture.

Netflix prize [9] is a competition to encourage the design of best movie recommendation algorithms. However, privacy is becoming a concern for Netflix algorithms since 2007 [11]. Both AdContRep and AdChatRep enhance privacy through frequent change of pseudonyms in MANET based social networking, but at the same time keep track users’ reputation information by applying a centralized trusted server.

MovieLens [8] is a movie sharing and recommendation website developed by GroupLens Research [5] at the University of Minnesota. While MovieLens focuses on movie (content) reputation, AdContRep can provide both content reputation and users’ recommendation trust.
Chapter 3

System Design

In this chapter, we introduce AdRep, a reputation system for various types of MANET services.

3.1 MANET Services

MANET services are applications that apply mobile ad hoc networks as a platform to provide functionalities to mobile users. We refer to MANET users as people who hold a mobile device (e.g., mobile phone or PDA) installed with MANET service applications. Each mobile device running MANET service application is a MANET node.

Typical MANET services are MANET games, MANET learning tools, traffic information sharing, and Internet connection sharing. For example, Gedda-Headz [4], which is available on both Nokia Ovi Store [10] and Apple Store [3], supports multiple participants playing games via Bluetooth. The winner of a game could immediately get an avatar called Headz from his/her rivals as a trophy. Sharing information via MANET with classmates in the same class brings great convenience to learning and education. Traffic information shared by drivers in the same area or street is more accurate than that from a central server. Sharing Internet resources via MANET when Internet connection or capability is limited is another useful service. In this thesis, we mainly consider two types of MANET services: pervasive social chatting and content recommendation.
3.1.1 MANET Chatting Service

Mobile ad hoc chatting enables users to chat locally with other users nearby. Such services are especially useful for assisting immediate decisions, including carpooling, purchase sharing, coupon sharing, etc. Assisting immediate decisions is particularly beneficial when users have difficulty to request information from a centralized service provider. Unlinkability is a common problem when smart phones or PDAs have no connections to the Internet, or the access is expensive or inconvenient.

In MANET chatting, users physically connected are not necessarily socially connected. In other words, users nearby might not know each other. Therefore, it is essential to provide a mechanism to help them decide who should be trusted during chatting. In MANET chatting, the most important indicator to assist the user decision is reputation. In other words, users not only aim to search for neighboring users who may hold similar interest, but more significantly, the most trustworthy ones. The extent to which reputation counts in decision assistance largely depends on the crucial level of chatting activities.

The chatting scenarios considered in MANET chatting are described as below.

- **Product sharing.** Claire is at a large shopping center alone. A product she likes is on sale under condition that 'Buy 3 Pay for 2'. However, she only needs one. She wants to ask her neighbors, whom she doesn’t know, via MANET who want to share the purchase or product.

- **Ticket sharing.** Claire decides to watch a movie in a cinema. The price for one ticket is 13.8€. However, if she can buy a packet of 5 tickets, it will be 8.6€ for each. She wants to share the ticket packet with her neighbors whom she doesn’t know. She asks whether they would like to share the discount via MANET.

- **Taxi sharing.** After the movie, a lot of people are leaving the cinema. Claire wants to watch a figure skating competition, which is quite far away. She would like to take a taxi and wants to share this ride with others. She discusses this with her neighbors via MANET and decides the people who can share riding with her.
3.1.2 MANET Content Recommendation Service

The MANET content recommendation service helps users to find the best contents by sending recommendation requests in MANET. Although there are many recommendation websites to collect and provide recommendation information, they hardly function when Internet is temporarily unavailable. Particularly, users in a local area sometimes can provide more convincing feedback directly to request users.

Similar to MANET chatting, content reputation is crucial in content recommendation. This service could be meaningless if users are allowed to provide irresponsible feedback. Trust and reputation mechanism is needed in such a service under two conditions. First, users are encouraged and rewarded if they give honest and good recommendations. Second, malicious feedback or bad recommendations are punished to ensure the service’s health. Among different types of recommendations, movie recommendation is one example. Below is an example scenario.

- **Movie recommendation** Alice wants to watch a movie at a cinema, but she has no idea which movie to choose. She tries to ask people around her for recommendation via MANET.

3.2 Assumptions and System Requirements

We aim to design a reputation system that supports various MANET services. Our design is based on a number of assumptions.

3.2.1 Assumptions Related to MANET Services

We list the assumptions related to MANET services below.

- **Assumptions about MANET.** We assume that MANET is always available as long as there are two mobile devices (e.g. mobile phones). The mobile devices can communicate with each other in an ad-hoc or peer-to-peer fashion via WLAN, Bluetooth or other cognitive radio media. The communication cost is low or free regardless of any Internet access provided by mobile operators or WLAN.

- **Assumptions about users.** Herein, users are human beings using mobile devices that can self-organize an ad hoc network. They would
like to hide their privacy, thus use pseudonyms to communicate with other people in MANET services. Users may not be familiar with each other, thus could be malicious users who attack the designed reputation system.

- **Assumptions about User Trust.** Our reputation system provides trust information to the MANET users in order to help them make a proper decision in MANET services. This information could be about reputation of a MANET user (e.g., in MANET chatting) or reputation of a content (e.g., in MANET recommendation services). We assume that MANET users generally believe trust information provided by our reputation system. Thus our system could assist users' decision.

### 3.2.2 Assumptions Related to a Trusted Server

We further assume a trusted server is a trusted third party which accumulates useful information to generate reputation of various entities in MANET services. These entities include various contents flowing over MANET, MANET nodes and MANET users. We further list a number of additional assumptions related to the trusted server.

- **Connection assumption.** There is no guarantee that the server can be accessed at any time. The connection to the trusted server is not always available. The trusted server is not in MANET. It is operated by a trusted service provider and can be connected via a fixed network, e.g., the Internet.

- **Trust assumption.** We assume that trust and reputation values generated by the Trusted Server (TS) are trustworthy enough for users. We assumed that trust/reputation values issued by the server are not tampered by any other parties.

- **Privacy assumption.** We hope private data are not revealed to any other parties. However, it is difficult to maintain trust and reputation information without disclosing private data to the trusted server. Therefore, we assume that user fully trust the trusted server to maintain their personal data. The trusted server does not disclose any private data of users to other parties.
3.2.3 Requirements on System Design

The reputation system should meet several practical and industrial requirements listed below.

1. **Context awareness.** MANET services cover broad areas and we aim to design a reputation system that could suit numerous types of services. However, the concrete meanings for trust and reputation are diversified in different kinds of services. Even in different scenarios within one type of service, it is not easy to provide a universal way to evaluate trust and reputation. On the other hand, designing one reputation systems for each specific type of service complicates the work. Therefore, we aim to design a trust/evaluation model that could support many types of MANET services.

2. **Privacy enhancement.** One concern in MANET services is privacy preservation. In many MANET services, nodes within one MANET are also physically around. Nodes within the network distinguish each other by their identities. Thus, the presence of node identity reveals node location. Meanwhile, chances are high to guess node identities by obtaining their location information. For example, if we detect one user with identity $Uid_1$ appears in around 10 meters, the user’s location information is leaked: within 10 m. On the other hand, if the service has detected one user within 10 meters and we really do find a person in around 10-meter distance using MANET services, it is easy to map between a user and his/her identity in real life. The problem described requires the reputation system to provide privacy enhancement mechanisms.

3. **Business compatibility.** The designed reputation system should also be practical to meet industrial needs. The requirements for business compatibility mainly comprise convenient deployment, low transfer cost, high extendability, easy maintenance and etc. Consequently, the reputation system should be not only functional but also lightweight.

4. **Power consumption.** As MANET services run on mobile devices, the reputation system should be power efficient and cause as little additional overhead as possible.
3.3 A Hybrid Reputation Management Framework

3.3.1 Decentralized Reputation Framework

The disadvantages of a decentralized reputation framework are obvious. The first problem is inconsistent reputation. In a pure decentralized reputation system, the reputation value of a node is stored in each node device. Reputation value is evaluated locally based on different experiences. In addition, different users may have different taste on the same node. Therefore, the reputation generated at different nodes could be quite different. Exchanging reputation information among nodes could cause extra communication traffic and consume more device power.

Besides, a fully decentralized reputation system increases chances of attacks. It is difficult to inform the whole network about malicious nodes due to mobility. For example, Node \( K \) might attack the network in Place \( A \) and leave bad behavior history there. But when Node \( K \) flees to Place \( B \) where every node in the vicinity is new, Node \( K \) could be treated as a good one since the nodes in Place \( B \) are not informed about previous malicious behaviors of Node \( K \).

3.3.2 Centralized Reputation Framework

In centralized reputation model, inconsistent reputation problem is overcome. Instead of maintained at each device separately, node reputation can only be evaluated by a trusted server. The trusted server collectes all feedback from other users on Node \( K \) and integrates it to provide a unified reputation value.

However, it is not convenient in practice for users always connecting to the server and therefore reputation value update is slow or even unavailable. Furthermore, users are not motivated to report and give feedback to a server since it is costly on both time and price.

3.3.3 A Hybrid Reputation Management Framework

We apply a hybrid framework for AdRep, based on the above analysis on both decentralized and centralized reputation management framework.
3.3.3.1 An Example of Hybrid Reputation Framework in Nature

Bees have been widely studied a successful example of communicating in ad hoc networks in the nature [43]. We can learn much from bee colonies on how to locate best places for hives [30]. Instead of communicating in a fully distributed fashion, the bee colony sends out a small number of scouts to search for potential hive positions. After the survey, they report their results to the centralized colony.

The potential best location list is updated based on the feedback from the scout bees. Meanwhile, the scout bees that are sent out communicate in a decentralized fashion to cooperate together for finding a better location. The bees that have discovered the best location will be awarded according to bee society rules. This is similar to award the users that behave well in a society with high reputation.

As summerised from Harvard Business Review [30], bees’ decision includes both information discovery and information integration.

“A centralized structure works well for discovery, because the individual’s role is to find information and report it back. In contrast, a richly connected network works best for integration and decision making, because it allows the individual to hear everyone else’s opinion about the expected return from each of the alternatives.”

3.3.3.2 Practical Reasons

The bee example explains how a hybrid model can support better information discovery and information integration and thus more elaborate reputation scheme in MANET services. Except for this, we have some practical reasons to design a hybrid trust model to support MANET services.

1. **Enhancing privacy.** Generally, privacy includes personal data privacy and location privacy. For pervasive social networking services, location privacy is easily disclosed. For example, a user can be recognized immediately by its User ID (UID). More specifically, if User A appeared in the local network with Uid A, then next time a user appears with Uid A again, it must be User A.
CHAPTER 3. SYSTEM DESIGN

Introducing the trusted server supports location privacy by allowing users to change their UIDs upon request. Even though a user may have many pseudonyms (UIDs), only one UID is active at a time, which is signed together with a timestamp and user’s current reputation value by the server. Other users have no information about whether two UIDs correspond to the same user. However, the server can map various pseudonyms to the same user since it knows the real ID and the user need to get confirmation from the TS if he/she wants to update the current pseudonym.

2. **Solving reputation inconsistency problem.** In AdRep, opinions of a node are reported to the server when the server is linkable. Although user preserves local reputation values, the level of inconsistency is greatly reduced because a global reputation value is issued by the server by aggregating all information reported by MANET nodes.

3. **Protecting the system against many attacks.** A centralized trusted server also prevents a reputation system from potential attacks. After discovering a malicious node, the server can notify the whole network more efficiently than spreading the notice via MANET.

4. **Supporting other promising services.** A centralized trusted server also supports various promising services as listed below.

   - Location based services. The centralized server supports pushing various types of information to users based on their locations. The type of information that the server can push is personalized by the user.

   - Friend notification. The server can also inform friends' approaching even though the friends change UIDs.

3.3.3.3 System Framework

Before illustrating reputation framework of AdRep, we define two types of trust/reputation.

**Definition 1. Local trust** on an entity is the trust value evaluated locally at a MANET node. Local trust is evaluated independently by each node. Therefore, the values of local trust on the same entity might not be necessarily be identical on different node.
Definition 2. *Global reputation* of a MANET node is the reputation evaluated by the TS. It is an attribute of a MANET node. Therefore, a MANET node has only one reputation at a time. However, users can update their reputation values by requesting to the TS.

We illustrate the trust/reputation framework of AdRep in Figure 3.1. Local trust values are evaluated locally at the MANET node side, whereas global reputation is evaluated at the TS.

The framework on MANET node side consists of 5 modules.

Figure 3.1: Illustration of the trust/reputation framework of AdRep
• **MANET Interface** supports MANET communication.

• **C/S Interface** supports connection to the trusted server. It supports three types of requests to the TS: reputation update, feedback on other users and user pseudonym’s (e.g., UID) update.

• **Personal Info DB** maintains node information, including node identity (that could be a pseudonym), its profile and its trust/reputation information.

• **Trust DB** stores local trust information on contacted nodes. History of other nodes assists trust evaluation on contacted node.

• **Local trust evaluator** integrates all information and data about trust based on MANET communications and calculates the trust value, which could assist user decision.

The framework at the TS side consists of 7 modules listed below.

• **Pseudonym update processor** receives users’ request on pseudonym updates and return a new pseudonym. The support for updating pseudonym enhances MANET user privacy.

• **Global reputation update processor** receives users’ request on updating global reputation and then looks up and distributes the current global reputation value.

• **Feedback processor** collects users’ opinions and feedbacks on other users or entities and inserts feedback information into Trust/Rep DB.

• **Trust/Rep DB** stores trust and reputation information of users and other entities. Such information includes trust/reputation values, the time when trust/reputation values are generated and etc. It also maintains feedback information for reputation evaluation (e.g., who gives feedback on whom, time of feedback).

• **Identity DB** maintains a mapping table between users and their pseudonyms. Each user has one permanent ID that is shared only between the user and the TS. Meanwhile, the permanent ID maps with several pseudonyms. Only the latest pseudonym is active at a time. Identity DB also keeps information of valid time of each pseudonym.
• **Global reputation evaluator** generates global reputation of a trustee entity based on globally collected information, e.g., users’ voting on other users in ad hoc chatting, users’ voting on some contents in content recommendation services, and the reported past social communication data by ad hoc nodes.

• **Identity token generator** signs an identity certificate for users by encapsulating the current pseudonym, reputation value, its valid time and etc. Users can validate other users’ identity by verifying trust token.

### 3.4 Trust/reputation Evaluation Model

We aim to apply a context aware trust/reputation evaluation model in AdRep for diversified MANET services.

#### 3.4.1 Local Trust Evaluation on Mobile Devices

Carsten [35] has described trust as a context and situation dependent attribute. Local trust value is determined by context \(c\), history \(h\) and time \(t\). We differentiate contexts by the types of MANET services (e.g., chatting service and recommendation service). Local trust value also varies depending on trust history. Furthermore, time is also an important factor for local trust evaluation. Thus, we define local trust of the current moment \(V_{t+1}\) as Formula 3.1, where \(v\) is a general local trust evaluation function.

\[ V_{t+1} = v(V_t) = v(c, h, t) \tag{3.1} \]

Furthermore, local trust relates a trustor, a trustee, a trust object and the environment [35]. Local trust on an object largely depends on the trust value of the trustee \((T)\), the quality of the object \((O)\) as well as the scenario \((S)\). Trust objects vary in different mobile ad hoc contexts. For example, in MANET content recommendation services, trust objects are recommended content, whereas in MANET chatting services, trust objects are user behavior. However, a same trust object might have different local trust levels due to different trustees. In content recommendation services, a same content recommended by two users might be evaluated with different trust value. A recommender with higher reputation is usually considered more trustworthy. Even given the same service (context), trust environment (scenario) can
be different. In recommendation context, the crucial level of recommending a movie is much lower than that of recommending an emergency exit. Similarly, in chatting context, sharing a piece of entertaining news is not as important as discussing about taxi sharing. Therefore, we further summarize the evaluation of local trust in Formula 3.2.

\[ V_{t+1} = v(c, h, t) = v(T_{c,h,t}, O_{c,h,t}, S_{c,h,t}) \]  \hspace{1cm} (3.2)

### 3.4.2 Global Reputation Evaluation on TS

Local trust evaluation is decided by many subjective and contextual factors. Therefore, local trust value for the same entity varies on different MANET nodes. By contrast, global reputation of an entity is an accumulative opinion collected from massive users and reports. Similar with local trust value, reputation is also context \((c)\) aware. For example, a user’s reputation for online transaction is hardly relevant with his/her reputation for writing news review. User’s social past experience or history \((h)\) also affects his/her reputation. Furthermore, reputation is a time-dependent \((t)\) attribute. People tend to trust reputation that is issued recently more than that issued long time ago. Thus, global reputation \((R)\) at the TS can be modeled in Formula 3.3, in which \(r\) is a general reputation evaluation function.

\[ R = \sum r(c, h, t) \]  \hspace{1cm} (3.3)

As reputation value is evaluated based on other users’ feedback, the trustworthiness and quality of feedback also affect global reputation evaluation. Users express their trust opinions on an entity to the TS by providing feedbacks. Thus, we herein refer to these users as trustors \((T)\). Users’ feedback could be directly based on an evaluated user or on the trust object \((O)\) relevant with the evaluated user. Evaluation on global reputation also varies in different scenarios. Therefore, we further summarize global reputation evaluation model in Formula 3.4.

\[ R = \sum r(T_{c,h,t}, O_{c,h,t}, S_{c,h,t}) \]  \hspace{1cm} (3.4)
Chapter 4

Implementation and Prototypes

MANET chatting and MANET recommendation are examples of MANET services. Based on the system design of AdRep in Chapter 3, we developed two prototype reputation systems AdChatRep and AdContRep for MANET chatting and MANET recommendation, respectively. AdChatRep is implemented based on the AwareNet/TWIN pervasive social networking platform developed by the Nokia Research Center [14], while AdContRep is a stand-alone system developed independently by the author.

4.1 AdChatRep: A Reputation System for MANET Chatting

4.1.1 Implementation Requirements

We analyze system requirements for MANET chatting by studying the ride sharing scenario. Claire wants to share a taxi with people around her to some place. We list the practical questions raised by Claire as below, categorizing them into ‘Communication’, ‘Reputation’, ‘Privacy’ and ‘UI’.

- [Communication] Who are nearby?
- [Reputation] Are they trustworthy and to which extent are they trustworthy?
- [Reputation] How to show their trustworthiness on the mobile device?
- [Reputation] How to decide trust value?
CHAPTER 4. IMPLEMENTATION AND PROTOTYPES

- **[Reputation]** Due to high user mobility, how to log user behavior and report to the server?
- **[Privacy]** If too many users are around, how to avoid sending crucial information in the public?
- **[Privacy]** How to send out queries or chat, while avoiding reveal or leak information about identity?
- **[UI]** If many users are chatting together, how to arrange the chatting UI structure so that chatting content looks easier to navigate?
- **[UI]** How to display users’ trust/reputation information in an easy-accepted style?

We summarize the requirements of system implementation in Table 4.1.

<table>
<thead>
<tr>
<th>Categories</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication</td>
<td>Support chatting in MANET</td>
</tr>
<tr>
<td></td>
<td>Support communication with the TS</td>
</tr>
<tr>
<td>Reputation</td>
<td>Evaluate local trust value on chatting users</td>
</tr>
<tr>
<td></td>
<td>Check global reputation of other users</td>
</tr>
<tr>
<td></td>
<td>Updates global reputation of user him/herself</td>
</tr>
<tr>
<td>UI</td>
<td>User-friendly chatting UI</td>
</tr>
<tr>
<td></td>
<td>User-friendly UI on reputation visualization</td>
</tr>
<tr>
<td>Privacy</td>
<td>Allow users to change nick names and user IDs.</td>
</tr>
</tbody>
</table>

Table 4.1: System implementation requirements in AdChatRep

4.1.2 System Functionalities

The AdChatRep prototype system has three modules: MANET chatting, trust/reputation management and privacy/identity management.

4.1.2.1 MANET Chatting

- **Nearby user detection** The application displays all users nearby and notifies the user when a new user appears.
Community management In order not to request and chat in public, users can create their own communities so that other users interested in the same chatting topic can join the community. We set three different crucial levels for community based chatting: important, normal and unimportant. Chatting behaviors in communities with different crucial levels have different influence on node user’s reputation generation.

Community chatting The application supports chatting in a community.

Node-Node chatting A user can select a specific node and chat individually with that user.

On-chat voting Users can vote on any chatting message during chatting. On-chat voting plays an important role in local trust evaluation.

Chatting style design Traditional chatting UI displays chatting messages in a time order. However, there are usually many users chatting at the same time in a community. Additionally, each user can change his/her nick name frequently. So we design a tree-style chatting UI in a way that users can reply to certain messages and the reply will show under the previous message in a time order.

4.1.2.2 Trust/Reputation Management

We apply the trust/reputation model introduced in Chapter 3 on MANET chatting services. In MANET services, local trust is the degree to which User A trusts User B during conversation. The significance of evaluating user trust locally is to help users make decisions by chatting with users who are physically nearby. For example, in the taxi sharing example, User A must quickly decide whether User B who intends to share a taxi with her is a nice user. Trust in MANET chatting is different from online chatting in the way that chatting users might highly likely to meet each other face to face. For example, we hope that a User B, who agrees to share the ride with User A during chatting, is not a guy who actually aims to sell his/her product to User A.

Based on Formula 3.1, we apply the local trust model of AdRep with consideration on context (c), history (h) and time (t). In AdChatRep, the context (c) is MANET chatting. Local trust evaluation also considers users’ past experience (h). AdChatRep stores the information of users who have been
CHAPTER 4. IMPLEMENTATION AND PROTOTYPES

contacted before. Time decay is another factor in local trust evaluation. AdChatRep assumes that users tend to trust information in current situation more than that in the past.

We further apply Formula 3.2 to describe local trust in AdChatRep in more details. In MANET chatting we define trustee ($T$) as a chatting application user. If User $A$ trusts User $B$, then User $B$ is the trustee. A trustee who holds higher global reputation is usually regarded more trustworthy in local trust evaluation. We further consider the influence of trust object ($O$). Trust object in AdChatRep could be a user, the content of conversation or user behavior during chatting. In chatting services, users’ opinions on each other is very subjective. Countless scenarios ($S$) can be applied in MANET chatting (i.e. product sharing, ticket sharing and taxi sharing). The way AdChatRep evaluates local trust varies in different scenarios.

Particularly, we consider the following factors in a user’s (e.g., User $B$) local trust evaluation on User $A$: global reputation of User $B$, locally stored trust values on User $B$, on-chat voting on User $B$, crucial level of the community in which the conversation content is voted, User $B$’s interest similarities with User $A$ and etc.

On the other hand, the TS accumulates massive opinions on a user (e.g., User $A$) to evaluate his/her global reputation. AdChatRep evaluates global reputation based on Formula 3.3 and Formula 3.4. The opinion is collected by user feedback (e.g. User $B$’s opinion on User $A$). If User $B$ provides feedback to the TS about his opinion on User $A$, we call User $B$ the trustor ($T$) and User $A$ the trust object ($O$). The TS does not weigh feedbacks from different users equally. The feedback from a trustor with a higher global reputation is considered more valuable. Besides that, AdChatRep also takes the reputation of the trustee into account. If an opinion of User $B$ on User $A$ deviates too much from User $A$’s previous records, then we lower the weight of the feedback and vice versa. We consider time ($t$) decay in the accumulated information. In other words, users’ current feedback weighs more than previous feedback.

4.1.2.3 Identity/Privacy Management

There are three IDs related to a user in the system: PIDs, UIDs and nick names. PID is the permanent user ID (i.e., real ID) that is never changed and is constantly kept at the TS. UID is temporary user ID, (i.e., pseudonym). UID is the ID for a MANET node to recognize a user. If UID is changed,
the node assumes that a different user is interacting with it. The advantage is that once a node changes its UID, other nodes can not recognize that they are the same node any more. UID is signed by the server and therefore the users are not allowed to change UIDs freely by themselves. The server can issue new UIDs upon user request.

The users are able to change their nick names as they want. Even though they change their nick names, as long as their UID is not changed, the node device can still identify it as the same node. Table 4.2 shows the differences among UID, PID and nick names.

<table>
<thead>
<tr>
<th></th>
<th>PID</th>
<th>UID</th>
<th>Nick names</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changable</td>
<td>No</td>
<td>Yes, by the server</td>
<td>Yes, by the user</td>
</tr>
<tr>
<td>Format</td>
<td>e.g., MANET node user’s Email address or MANET node device’s unique ID</td>
<td>16-byte</td>
<td>String, Literal text</td>
</tr>
</tbody>
</table>

Table 4.2: Differences among PIDs, UIDs and nick names in AdChatRep

4.1.3 Impelmenttion and Prototype

4.1.3.1 Hardware and Software Platforms

The AdChatRep system development contains two parts: mobile device part and trusted server part. We develop mobile device part on Nokia N900 using Python with GTK bindings and implement server part on Linux (Ubuntu 9.04) together with Apache, PHP and MySql.

We implement AdChatRep system based on and extend functions of Twin application developed by Tempere University of Technology. The Twin application is an ad-hoc chatting application based on AwareNet [14] – a MANET pervasive social networking platform developed by Nokia Research Center, Helsinki. It provides an energy-efficient and fully distributed social networking environment.
4.1.3.2 AdChatRep Features and User Interfaces

We take taxi sharing as an example to present features in AdChatRep. Claire plans to share a taxi with one more person to the airport, with the aid of AdChatRep.

‘Twin’ is the default community in AdChatRep and every user nearby (e.g., Alice and Bob) can be found in ‘Twin’ community, which is shown in Figure 4.1.

![Figure 4.1: Detected nearby users in AdChatRep](image)

Claire can update profile information (displayed in Figure 4.2) including profile picture, nick name, real name, age, gender, description, city, state,
CHAPTER 4. IMPLEMENTATION AND PROTOTYPES

Figure 4.3: An example UI of updated global reputation from the TS in AdChatRep

Figure 4.4: An example UI of creating a new chatting community in AdChatRep

country and etc. Claire can check her global reputation (i.e., 0.513) issued by the server, which is visualized by the number of diamonds.

Claire can not change global reputation values by herself, but she can update it (i.e., to 0.513621 herein) from the TS, as is shown in Figure 4.3.

Before starting conversation on taxi sharing, Claire first creates a community, because conversation of various topics may appear in ‘twin’ community. To create a community, Claire needs to input community name, crucial level and descriptions, as is shown in Figure 4.4. Users who feel interested in the topic can join the community (Figure 4.5).
Claire starts the conversation in ‘Taxi-sharing’ community with Alice and Bob (Figure 4.6).

Besides chatting, AdChatRep also supports on-chat voting, which allows users to vote ‘UP’ or ‘DOWN’ with additional comments (Figure 4.7).

Users’ local trust values are adjusted every time they are voted during chatting. For example, after being voted up, Bob is issued higher local trust value by AdChatRep. The increase in local trust value is visualized by increasing the amount of battery power (i.e., from yellow to green), as is shown in Figure 4.6. The following four types of information are displayed in chatting UI:

- Profile picture of the chatting user (i.e., community member)
CHAPTER 4. IMPLEMENTATION AND PROTOTYPES

![On-chat voting UI in AdChatRep](image)

Figure 4.7: On-chat voting UI in AdChatRep

![UI of the user list for afterwards voting in AdChatRep](image)

Figure 4.8: UI of the user list for afterwards voting in AdChatRep

- Reputation indicator that indicates each chatter’s local reputation value
- User’s nick name
- The input time of conversation message
- Message content

AdChatRep supports afterwards voting on users. The voting is reported to a central server. Users who have contacted with Claire are automatically logged on the device and Claire can vote a same user many times. The data is processed by TS with consideration on time decay.

The voting list (shown in Figure 4.8) displays information of all contacted users to help Claire distinguish among them. Displayed information of a user
includes profile picture, nick name, local trust value (with its visualization) and UID.

*Claire* gives feedback to the TS by expressing her attitude towards a user within 5 scales, namely ‘Very dislike’, ‘Dislike’, ‘Neutral’, ‘Like’, and ‘Very dislike’ (Figure 4.9).

If *Claire* is curious how the local trust value of *Bob* is generated, she can check *Bob’s* profile (shown in Figure 4.10), which presents *Bob’s* global reputation value (i.e., 0.600 herein) and its visualization (i.e., four diamonds).

Furthermore, by clicking the ‘Details’ button, AdChatRep displays some cues about how user local trust is evaluated. (Figure 4.11).
AdChatRep provides several ways to visualize reputation. For example, three styles are available to visualize local trust values, namely the amount of remaining battery power, growing process of flowers and different roles and characters in cartoons (in Figure 4.12). Users can also choose either the number of crowns or the number of diamonds to visualize global reputation.

However, some users might not want to display reputation on AdChatRep. Thus AdChatRep contains two modes: reputation visible mode and reputation invisible mode. In reputation visible mode, reputation is displayed normally while in reputation invisible mode, all reputation related information is hidden. This feature is also used in user study in the following chapter.

Even though users are able to change nick names freely, their privacy is not
preserved because they can be recognized by users who are contacted before, as is shown in Figure 4.13.

![Figure 4.13: An example of user presence history in AdChatRep](image)

AdChatRep detects user presence not by their nick names but by their UIDs. AdChatRep enhances privacy by allowing users to change UIDs while preserving global reputation values. In Figure 4.14, Claire is issued a new UID by the TS.

![Figure 4.14: An example of changing UID in AdChatRep](image)
CHAPTER 4. IMPLEMENTATION AND PROTOTYPES

4.2 AdContRep: A Reputation System for MANET Content Recommendations

4.2.1 Implementation Requirements

We analyze reputation system requirements for MANET content recommendation services based on a movie recommendation scenario. Alice decided to watch a movie in a cinema, but she has no idea which movie to watch. She does not check movie ranking list from a website or central server for either two reasons. First, she does not have Internet access to such a website. Second, movie recommendation from the website represents the general information about each movie, but she would like to know more information about a movie showed in a specific cinema.

Based on the above scenario, we need to answer the questions listed below for system design.

- [Communication] How to send out a recommendation request?
- [Communication] Whether to reply for a recommendation request? If not, how to support filtering requests?
- [Communication] How to reply for a recommendation request? In which format the reply should be sent out?
- [Communication] Normally, users only want to send request, and do not have motivation to reply, even if they have an answer for the questions. How to motivate users to reply?
- [Reputation] To what extent can we believe the reply?
- [Reputation] How to calculate the trust value?
- [Reputation] In order to enrich information from the central server, we want to encourage users to send feedback to users about their experience, e.g. opinions about content or a participant who provided a reply. How to motivate users to send feedback to a server?
- [Reputation] How to support users to download certain types of content that they are interested whenever the Internet is available?
- [Reputation] How to prevent from irresponsible recommendations such as advertisement, spam, inflation, deflation and etc?
• [Privacy] How to distinguish from different users when they are all strangers?

• [Privacy] How to avoid revealing identity when raising a request and giving a reply?

• [UI] How to display recommendation results?

Based on the above questions, we categorize AdContRep implementation requirements into ‘Communication’, ‘Reputation’, ‘Privacy’ and ‘UI’ and list them in Table 4.3.

<table>
<thead>
<tr>
<th>Categories</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication</td>
<td>Support recommendation query/reply in MANET</td>
</tr>
<tr>
<td></td>
<td>Support connection to the TS</td>
</tr>
<tr>
<td>Reputation</td>
<td>Evaluate local trust values on recommended content</td>
</tr>
<tr>
<td></td>
<td>Evaluate local trust values on recommenders</td>
</tr>
<tr>
<td></td>
<td>Validate global reputation tokens of recommenders</td>
</tr>
<tr>
<td></td>
<td>Collect feedbacks at a TS</td>
</tr>
<tr>
<td></td>
<td>Evaluate global user reputation at a TS</td>
</tr>
<tr>
<td></td>
<td>Allow users to update their global reputation by requesting the server</td>
</tr>
<tr>
<td></td>
<td>Allow users to download global content reputation of specific categories by requesting the server</td>
</tr>
<tr>
<td>UI</td>
<td>User-friendly recommendation query/reply UI</td>
</tr>
<tr>
<td></td>
<td>User-friendly UI on recommended item list</td>
</tr>
<tr>
<td>Privacy</td>
<td>Allow users to change pseudonyms</td>
</tr>
</tbody>
</table>

Table 4.3: System implementation requirement in AdContRep

4.2.2 System Functionalities

AdContRep has three main functionalities: MANET content recommendation, trust/reputation management and identity/privacy management.

4.2.2.1 MANET Content Recommendation

• Recommendation query allows users to send out a recommendation query. Users are able to choose a request from a list of categories such as movie, restaurant, food and etc.
• **Recommendation reply** provides a UI for users to respond to content recommendation queries if they have an answer and are willing to respond.

### 4.2.2.2 Trust/Reputation Management

In content recommendation service, users are trying to collect recommended items and evaluate their trust values. We evaluate content local trust values based on Formula 3.1 and Formula 3.2.

In MANET content recommendation context (c), we define the user who raises the recommendation queries as the trustor (T), users who reply as trustee and the recommended objects as trust object (O). Users store historic information (h) of content and other nodes locally to assist future evaluation. In the context of content recommendation, trust evaluation varies in different scenarios (e.g. recommending a restaurant, a bar or an electronic device). Particularly, we consider the following factors when locally evaluating content trust values:

• **Recommendation times of a content**. If a movie is recommended by the majority of reply users, then it is likely that the movie is worth watching.

• **Content rating value**. A reply highly recommending a movie can rate it with high value. On the other hand, a reply user might not know which movie is good and worth watching, but he might know some movies that are not recommended. For those movies, they are rated with low values. Users tend to choose movies that have higher ratings.

• **Content rating deviation**. It is also common that people give recommendation to the same movie with different comments. Some users give high recommendations while other users are not in favor. This cause deviation in movie recommendation values. Rating deviation cause by either different user taste or unfair rating or malicious rating.

• **Interest similarities**. We also need to consider interest similarities in the recommendations. If reply users have totally different taste from the request user, content trust value could be lower.

• **Local content database**. As mentioned before, when the Internet is available, users can download content reputation information from
the TS. Even though information from local area is real-time based, content information from a server is usually more trustworthy. Therefore, we should integrate content reputation information stored in local database whenever possible.

Formula 3.3 and Formula 3.4 indicate that reputation is calculated on a cumulative bases. The TS collects query users' opinions on received recommendation items (e.g. movies). The TS calculates content reputation based on the feedbacks. Basically, reputation values are evaluated in a similar way as content local trust evaluation, but with a larger dataset. However, different from AdChatRep, AdContRep maintains two types of global reputation: content reputation and user reputation. TS evaluates user reputation based on their recommendation activities. When Query User $A$ reports feedback on Content $M$, information about users who has recommended Content $M$ (e.g. Reply User $B$) is also sent to the TS. In other words, good recommendation helps users to promote reputation.

4.2.2.3 Identity/Privacy Management

There are two kinds of IDs in the AdContRep system: PID and UID. PID is the permanent ID issued by the server and it remains the same and is transparent to other users except the server and identity holder. UID are pseudonyms of users to distinguish with each other in MANET. Users can change pseudonyms by requesting for privacy preservation. Once pseudonym is changed, the user is treated as a new user in MANET. Table 4.4 shows the differences among UID, PID and nick names in AdContRep.

<table>
<thead>
<tr>
<th>PID</th>
<th>UID (Pseudonym)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changable</td>
<td>No</td>
</tr>
<tr>
<td>Format</td>
<td>16 bytes digits</td>
</tr>
<tr>
<td></td>
<td>16-byte</td>
</tr>
</tbody>
</table>

Table 4.4: Differences between PID and UID in AdContRep

4.2.3 Implementation and Prototype

4.2.3.1 Hardware and Software Platforms

The AdContRep system development contains two parts: mobile node part and server part. We develop mobile node part on Nokia N810 using Python
with GTK binding and server part on Linux (Ubuntu 9.04) together with
Apache, MySQL and PHP. MANET communication is based on Wireless LAN.
There is no guarantee for MANET nodes to connect to the trusted server.

4.2.3.2 AdContRep Features and User Interfaces

We present functions of AdContRep with the movie recommendation sce-
nario. Alice, who is physically in a cinema called Finnkino\footnote{Finnkino is a
Finnish film distributor and the biggest movie theatre chain in Finland},
tries to collect suggestion from other AdContRep users who are nearby. Figure 4.15
displays local data management UI in AdContRep, which enables Alice to update her
reputation and download content recommendation data collected at the TS
side.

![Figure 4.15: Local data management UI in AdContRep](image)

From Figure 4.16, we can find detailed information of Alice, including her
User ID (i.e., 22826847), user name (i.e., etugvqag), valid period of the user
name (i.e., from 2009.12.10 to 2010.1.10), query reputation (i.e., 1) and re-
ply reputation (i.e., 12). Even though each AdContRep user has only one
permanent User ID, they can update their pseudonyms from the server. In
AdContRep, users identify each other by their pseudonyms. Pseudonyms are
not sent in plain texts but signed by the TS as trust tokens. AdContRep
users validate a user’s pseudonym by checking his/her trust token. The TS
maintains data that maps between different pseudonyms and the permanet
User IDs.
Alice then sends out a query to nearby AdContRep users. The query UI shown in Figure 4.17 includes query category, query content and Alice’s personal information for peer users to authenticate. The information is displayed on Alice’s query UI for her to check which types of personal information is sent and their correctness.

After timeout, which is set by Alice, she will receive some recommendation replies from nearby users, shown in Figure 4.18. The results include movie names, average rating values and number of recommendations. The average
rating value is visualized by the number of stars.

![Recommendations on Movie & TV](image)

Figure 4.18: An example of content recommendation reply in AdContRep

Let us assume that Alice chose and watched the movie ‘Avatar’ based on recommendations. She is encouraged to give feedback on the content (movie) quality to the TS. Alice could only give feedback on the categories she has queried before. In Figure 4.19, Alice has requested content recommendations in two categories: ‘Movie&TV’ and ‘Gourmet Food’.

![Feedback](image)

Figure 4.19: An example of feedback categories in AdContRep

Within ‘Movie&TV’ category, Alice could further rate one item from the list
of recommendations that she had received but not yet reported before, as shown in Figure 4.20. Even though Alice only votes for content quality, the information of users who have recommended the content is sent together in feedback data to the TS. This becomes the major evidence of user reputation evaluation at the TS.

![Feedback Interface](image)

Figure 4.20: An example of content quality feedback in AdContRep
Chapter 5

Evaluation

This Chapter presents our evaluation results of AdChatRep and AdContRep. Herein, we adopt two different evaluation methodologies to prove the effectiveness of these two systems. For AdChatRep, we apply user study to prove its usefulness, effectiveness and UI design acceptance. For AdContRep, we design a number of simulations to show its advantages in assisting users’ decision and its robustness against unfair rating and on-off attacks.

5.1 Evaluation on AdChatRep

5.1.1 Evaluation Design

We carried out two rounds of user study. The first round of study was a pre-prototyping survey that investigated participants’ opinions on the following issues before AdChatRep implementation: 1) AdChatRep’s usefulness, 2) factors influencing trust/reputation in chatting, and 3) reputation visualization UI design. The second round of user study was conducted after prototyping AdChatRep, we invited different participants to use AdChatRep based on our designed scenarios. In this round of user study, we not only collected participants’ opinions on the issues surveyed in the first round study, but also further analyzed user behaviors based on chatting logs, voting values and interviews.
5.1.1.1 Round 1 – Pre-prototyping Survey

We invited 20 participants in the survey. 75% of them are university students. Among the participants, 65% are male and 35% are female. 90% of them are between 22 and 27 years old, while 10% are above 36. According to the background information, all of them have Internet chatting experiences, 75% have mobile Internet chatting experiences, but none of them have experienced MANET chatting.

The survey questions include three parts:

- **Part 1** Significance of developing a reputation system for MANET chatting
- **Part 2** Factors concerned to build up trust in MANET chatting
- **Part 3** Reputation visualization UI design

**Part 1** investigates participants’ opinions on how important reputation system is in the three scenarios mentioned in Chapter 3, namely product sharing (Scenario A), ticket sharing (Scenario B) and taxi sharing (Scenario C).

**Part 2** aims to investigate participants’ opinion on the following factors during chatting.

- Factor A: Crucial level of chatting topic, which represents how important the topic is.
- Factor B: Depth of chatting, mainly concerns with rounds of chatting among users
- Factor C: Interest similarity. So far we define interest similarity of two users as the number of communities commonly joined by users.
- Factor D: On-chat voting, which is the voting towards chatting content during chatting. On-chat voting is displayed at the same time with chatting content.
- Factor E: Afterwards voting, the voting for a specific user. Afterwards voting is also the feedback to the TS and is the main cue for the TS to evaluate the reputation of a user.
CHAPTER 5. EVALUATION

The reputation or trust value of a user can be visualized in various formats. In Part 3, participants further rated the degree to which they with regard to a number of reputation visualization styles. We provide four styles in the survey:

- Style A: Size, reputation visualized through a chatter’s input text font size, the bigger the higher reputation;
- Style B: Number, reputation visualized through number of objects, e.g., stars, the more the higher reputation;
- Style C: Growing process, reputation visualized through growing process of a creature, e.g., a tree, the older the higher reputation;
- Style D: Roles, reputation visualized according to different roles in a community, e.g., roles of family members in cartoon ‘The Simpsons’.

5.1.1.2 Round 2 – Prototype-based User Study

In Round 2, we use between subjects design to test the effects of AdChatRep on mobile users in three ad hoc chatting scenarios. We test 2 groups of participants; each group consists of 7 participants. Group 1 uses AdChatRep with reputation information hidden to them while participants in Group 2 chatted with AdChatRep with trust and reputation information visualized. We aim to test and investigate user behavior differences between Group 1 and Group 2. To fulfill this, we logged chatting contents and participants’ opinions with each other after scenario-based chatting. Thus, we can find the effects of AdChatRep on mobile users during MANET chatting.

We carried out Round 2 user study in two student villages and randomly selected 7 participants in each student village. All participants are university students aging between 23 and 28, except one from Group 1 is a teacher above 30. Among 7 participants in each group, 3 of them are female and 4 are male.

We designed the user test in a board game style in order to better organize the study and make the results of two groups’ tests comparable. Similar to a board game, we defined scenarios, assigned roles and issued action rights. We distributed a card to each participant, which indicated clearly scenarios and the participant’s role and actions in each scenario. In this way, each participant had a good understanding of his/her task. We also provided detailed instructions for each scenario.
CHAPTER 5. EVALUATION

The participants simulated the three chatting scenarios mentioned above. For each scenario, they chatted in a pre-created community. The participants were required to use different nick names in each scenario. We provided the nick names for each participant on the card to simplify the test.

**Scenario A.** Nick names of participants were A, B, C, D, E, F and G. G could buy 3 toothpastes and pay for 2. However, G needed only 2. G tried to share the purchase with someone else. However, G could buy 3 toothpastes by himself/herself, because they were not expensive after all. B could consider the deal if the price was reasonable, but not necessarily. A, C, D, E and F did not need to buy toothpastes at all. They joined the chatting community just out of curiosity. After chatting, each participant gave their feedback on contacted participants, but not necessarily to all, to the TS. The participants were encouraged to update their global reputation after chatting.

**Scenario B.** Participants changed their nick names to Yi, Er, San, Si, Wu, Liu and Qi. They were assigned with different roles. Er bought a packet of 5 movie tickets but he needed only 4. He/she planned to share the 5th with another person who might need to purchase one. Yi, Liu and Qi all chatted and voted seriously in the community in order to buy the ticket from Er, because it was cheaper. Meanwhile, San, Si and Wu did not need any ticket at all, and they joined the community just out of curiosity. After chatting, they gave feedback and update reputation.

**Scenario C.** We assigned new nick names to participants: Dow, Ray, Mee, Far, Sew, Lao and Tao. They chatted with new roles in a new scenario. Dow needed to take a taxi from the student village to the airport due to too much luggage to carry. He/she tried to find no more than 3 people to share the taxi as well as the cost. The more people share the price, the lower the cost for each. Ray and Mee were trying to join the sharing, even if it was a little bit more expensive than taking a bus. Far and Sew were not so motivated to take a taxi. They did not have much luggage to take. However, they would consider joining if the price was low. Lao and Tao were not going to the airport at all. They joined the community just out of curiosity. Similarly, we invited participants to give feedback to the TS and update reputation afterwards.

During the above tests, chatting information, including chatting time, users, contents and on-chat votings, and afterwards voting information were logged.
for analysis. Additionally, participants were invited to complete a survey. The survey has the same items as the pre-prototyping survey except some additional ones, e.g., comments on AdChatRep. Finally, we randomly interviewed some participants for their opinions on AdChatRep.

5.1.2 Results and Analysis

We collected the following results for analysis: survey in Round 1 and Round 2, chatting logs and voting and reputation logs in Round 2 and interviews after Round 2. For the survey, participants rated each item from 1 to 5 and we calculate the average value of ratings towards each item. We then compare the results between Round 1 and Round 2 and between Group 1 and Group 2.

In Part 1 of the survey, participants rated the level of usefulness of a reputation system in three scenarios. Users ranked the level of usefulness from 1 to 5. Table 5.1 presents the mean of usefulness rating in Round 1 and Round 2. The results showed a slight increase of mean of usefulness from Round 1 to Round 2. This indicated that users have stronger feeling that reputation information is important after experiencing MANET chatting. Another observation is the extent to which users needed reputation assistance varies in different scenarios.

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>Round 1</th>
<th>Round 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Group 1</td>
<td>Group 2</td>
</tr>
<tr>
<td>Sc1. Product Sharing</td>
<td>3.70</td>
<td>3.71</td>
</tr>
<tr>
<td>Sc2. Ticket Sharing</td>
<td>3.70</td>
<td>4.00</td>
</tr>
<tr>
<td>Sc3. Taxi Sharing</td>
<td>4.00</td>
<td>4.29</td>
</tr>
</tbody>
</table>

Table 5.1: Mean of rating on AdChatRep usefulness in Round 1 and Round 2

In Part 2 of the survey, participants rated to which extent they agreed with factors that are related with deciding reputation values. The ratings range from 1 (totally disagree) to 5 (totally agree). The factors and mean rating value in two rounds are shown in Table 5.2. The importance of crucial level is lowered in both groups in Round 2. We discover that in Round B, even though users were chatting on different topics in different communities, participants were not always chatting on the topic relevant with the community. For example, participants were also talking about news, weather and sport
while doing the test. This gives us the hint that user behavior in chatting is subjective and not confined with communities. Another observation is that participants in Round 2 were more interested in on-chat voting. In Round 1, participants had no experience on on-chat voting. Meanwhile, in Round 2 Group 1, as all reputation information was hidden from participants, on-chat voting provided little help to them on trust and reputation. In Round 2 Group 2 however, users felt more interested in chatting and voting because trust values of chatting users were updated and visualized dynamically.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Round 1</th>
<th></th>
<th>Round 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Factor 1. Crucial level of topic</strong></td>
<td>4.1</td>
<td>3.9</td>
<td>3.7</td>
<td></td>
</tr>
<tr>
<td><strong>Factor 2. Depth of chatting</strong></td>
<td>3.7</td>
<td>3.5</td>
<td>3.7</td>
<td></td>
</tr>
<tr>
<td><strong>Factor 3. Interst similarity</strong></td>
<td>3.2</td>
<td>3.4</td>
<td>3.6</td>
<td></td>
</tr>
<tr>
<td><strong>Factor 4. On-chat voting</strong></td>
<td>3.5</td>
<td>3.6</td>
<td>4.1</td>
<td></td>
</tr>
<tr>
<td><strong>Factor 5. Afterwards voting</strong></td>
<td>3.9</td>
<td>3.7</td>
<td>3.7</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.2: Mean of rating on AdChatRep effectiveness in Round 1 and Round 2

We collected participants’ opinions on different reputation visualization styles in Part 3. Participants provided rating ranging from 1 (totally dislike) to 5 (totally like) and we calculated their average values in Table 5.3. Generally, participants would prefer visualizing reputation by the number of objects such as stars or diamonds, because they are simple. The results also show that participants in Round 2 like the Size Style much more than participants in Round 1. The reason could be that participants in Round 1 found it boring after long chatting with same content style. The change in text size could emphasize texts from users who had higher reputation. Additionally, Participants in Round 2 had more interest in displaying reputation through growing process after using AdChatRep.

We further analyzed chatting logs in both groups in Round 2. Table 5.4 presents the time spent on chatting time and chatting length in each scenario. Generally, participants chatted more with content but more efficiently in Group 2. In Group 1, participants started the application in Scenario 1 and became interested in Scenario 2. However, they were gradually getting bored with chatting in Scenario 3. On the other hand, results in Group 2 show the opposite. Participants felt more interested and involved in chatting with reputation information and visualization. The results together with user behavior in the two groups further indicate that reputation display promotes
CHAPTER 5. EVALUATION

<table>
<thead>
<tr>
<th>Factors</th>
<th>Round 1</th>
<th>Round 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Group 1</td>
<td>Group 2</td>
</tr>
<tr>
<td><strong>Style 1. Size</strong></td>
<td>1.70</td>
<td>3.00</td>
</tr>
<tr>
<td><strong>Style 2. Number</strong></td>
<td>4.20</td>
<td>4.14</td>
</tr>
<tr>
<td><strong>Style 3. Growing process</strong></td>
<td>3.86</td>
<td>3.40</td>
</tr>
<tr>
<td><strong>Style 4. Roles</strong></td>
<td>3.20</td>
<td>3.57</td>
</tr>
</tbody>
</table>

Table 5.3: Mean of rating on reputation visualization style of AdChatRep in Round 1 and Round 2

user involvement in chatting.

<table>
<thead>
<tr>
<th></th>
<th>Round 1</th>
<th>Round 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sc.1</td>
<td>Sc.2</td>
</tr>
<tr>
<td><strong>Chatting duration (minutes)</strong></td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td><strong>Number of chatting characters</strong></td>
<td>391</td>
<td>724</td>
</tr>
</tbody>
</table>

Table 5.4: Analysis on chatting log in Round 2

After each user test in each group, we randomly interviewed several participants. In Group 1, interviewed participants were interested in other potential use cases of AdChatRep. One participant provided more use cases such as dating and selling last-minute ticket at concert hall. However, participants felt difficult to browse chatting history in a long chatting. Thus, chatting log navigation requires improving. In Group 2, interviewed participants regarded the user test with AdChatRep as a board game. They suggested expanding AdChatRep to a new game, in a way that even strangers nearby could play the game together. In addition, they provided further suggestions for better chatting UI.

5.2 Evaluation on AdContRep

5.2.1 Evaluation Design

AdContRep aims to assist user decision in choosing the most appropriate content in MANET. We design a number of simulations to first show the advantages of AdContRep in assisting users’ decision, and then demonstrate the effects of unfair rating and on-off attacks on AdContRep. In AdContRep,
malicious nodes can provide dishonest recommendations on contents in order to frame good ones and/or boost bad ones. This attack, referred to as the bad mouthing or unfair rating attack is the most straightforward attack [38, 39]. Malicious nodes could also behave well and badly alternatively, hoping that they can remain undetected while causing damage [38, 39]. This attack is called as the on-off attack, which is also one type of conflict behaviour attack. AdContRep aims at overcoming at least the above two common attacks. Two algorithms are designed to implement the hybrid trust model of AdContRep. Algorithm 1 is used by the MANET query node to evaluate content reputation. Algorithm 2 is applied by the TS to evaluate content global reputation and MANET node’s recommendation trust. Refer to [50] for details.

In our simulations with Matlab, the total number of users is \( L = 50 \), we randomly select a query node (from good nodes) and a random number of nodes to respond each query. We consider three contents \( (M = 3) \) with quality low \( (T_{c_1} = 0.1) \), medium \( (T_{c_2} = 0.5) \) and high \( (T_{c_3} = 0.9) \), respectively. For simulation simplification, we assume that ad hoc routing trust can be ensured. Our simulation is based on a number of assumptions:

- **Content quality.** There are three levels of contents: high quality \( (C_1) \), middle quality \( (C_2) \) and low quality \( (C_3) \). Users vote for each content according to their understanding towards different content.

- **Honest voting.** Users vote for each content according to their opinions towards content. Honest users vote for a content according to its actual quality level. For example, an honest user votes \( C_1 \) as quality low, \( C_2 \) as middle and \( C_3 \) as high. We do not consider the case that honest users vote differently based on their subjective tastes. In other words, even if \( C_1 \) is good, a user might honestly vote it low because he does not like it.

- **Randomness.** Based on the characteristics of mobile ad hoc networks, we assume the following randomness in our simulations: 1) a random user sends recommendation request (i.e., query); 2) a random number of users reply; 3) the query responding users are randomly selected; 4) a user randomly selects some contents to vote.

- **Content reputation.** The reputation of each content increases or decreases based on user voting. The range of content reputation value is between 0 and 1.
• **User’s recommendation trust.** User’s recommendation trust is evolved based on their voting quality. User’s recommendation trust is evaluated at both mobile node side and the trusted server side. The range of its value is between 0 and 1.

We adopt commonly used metrics in information retrieval, Recall ($R$), Precision ($P$) and F measure ($F$) to describe the malicious node detection performance [6]. For each good node $k$, the number of nodes that belong to Malicious Node (MN) and are indeed detected as MN, denoted as $x(k)$; the number of nodes that don’t belong to MN but are added to MN, denoted as $y(k)$; the number of nodes that belong to MN but are not detected as MN, denoted as $z(k)$. With these data we do a precision-recall evaluation. We define:

$$R(k) = \frac{x(k)}{x(k) + z(k)}$$  \hspace{1cm} (5.1)

$$P(k) = \frac{x(k)}{x(k) + y(k)}$$  \hspace{1cm} (5.2)

$$F(k) = \frac{2P(k)R(k)}{P(k) + R(k)}$$  \hspace{1cm} (5.3)

The node average F measure of the whole network is:

$$F = \sum_{k=1}^{\vert G \vert} \frac{F(k)}{\vert G \vert}$$  \hspace{1cm} (5.4)

where $\vert G \vert$ is the number of good nodes in the network.

We design 4 simulations as described below to evaluate AdContRep system.

**Simulation 1.** We test the performance of Algorithm 1 when the server connection is not available with four scenarios: 1) each query responding node randomly selects one content to vote honestly; 2) each responding node votes all contents honestly; 3) 5 fixed attackers in the network randomly vote one content dishonestly (e.g. deflate $C_3$ and inflate $C_1$ and $C_2$), whilst other responding nodes randomly vote one content honestly; 4) 5 fixed attackers vote all contents dishonestly, whilst other responding nodes vote all contents
honestly. In all of above scenarios, we fix the query node and suppose that all node pseudonyms are not changed.

**Simulation 2.** We further test the performance of both Algorithms 1 and 2 when the server connection is available periodically with the same four scenarios. We randomly select a query node in each query. The query node reports the query results each time when it connects the server after a query. The node’s pseudonym is changed if it has responded 3 queries. During the change, the server issues a new trust token attached to the new pseudonym.

**Simulation 3.** We test two scenarios to evaluate the performance of AdContRep against unfair rating attacks: 1) 5 fixed attackers in the network randomly vote one content dishonestly, whilst other responding nodes randomly vote one content honestly; 2) increase the number of attackers to 10 nodes.

**Simulation 4.** Furthermore, we test another two scenarios to evaluate the performance of AdContRep against the on-off attack: 1) 5 fixed attackers randomly vote one content with honest and dishonest recommendations alternatively, whilst other responding nodes randomly vote one content honestly; 2) increase the number of attackers to 10 nodes.

### 5.2.2 Evaluation Results

Figure 5.1 Content reputations at a query node: (a) each recommender node votes one content honestly; (b) each recommender node votes all contents honestly; (c) 5 malicious nodes vote one content dishonestly; (d) 5 malicious nodes vote all contents dishonestly.

Figure 5.1 shows Simulation 1 result of content reputations at a query node. We observe that 1) Algorithm 1 performs well when there is no any attack in the system, as shown in Figure 5.1 (a) and (b); 2) it can adjust the node recommendation trust to improve the content reputation evaluation, as shown in Figure 5.1 (c) and (d); 3) Algorithm 1 performs better if more recommendation information is collected locally (comparing Figure 5.1 (a) to Figure 5.1 (b) and Figure 5.1 (c) to Figure 5.1 (d)). Note that in Figure 5.1 (a) and (c), some reputation values of $C_3$ are very low, this is caused by a small number of recommendations since a random number of nodes are selected to respond the queries in this simulation.
CHAPTER 5. EVALUATION

Figure 5.1: Content reputations at a query node: (a) each recommender node votes one content honestly; (b) each recommender node votes all contents honestly; (c) 5 malicious nodes vote one content dishonestly; (d) 5 malicious nodes vote all contents dishonestly

The Simulation 2 result of content reputation at TS is shown in Figure 5.2. Three observations are made. First, AdContRep performs very well at the TS when there is no attack in the system (Figure 5.2 (a) and (b)). Second, AdContRep can adjust the node recommendation trust to identify the malicious node and improve the content reputation evaluation efficiently even though the node pseudonyms could be changed frequently (Figure 5.2 (c.2, c.3) and (d.2, d.3)). Third, AdContRep performs better if more recommendation information is accumulated by the TS (comparing Figure 5.2 (a) to Figure 5.2 (b) and Figure 5.2 (c) to Figure 5.2 (d)). Finally, the TS can evaluate node trust and content reputation more accurately and efficiently than the nodes (comparing Figure 5.2 to Figure 5.1), thus it is significant to introduce the TS into AdContRep.

The influence of the unfair rating attack investigated by Simulation 3 is demonstrated in Figure 5.3 (a). In the evaluation for unfair rating and on-off attacks, we observe that the malicious node detection at the node is much
Figure 5.2: Content reputations at TS: (a) each recommender node votes one content honestly; (b) each recommender node votes all contents honestly; (c) 5 malicious nodes vote one content dishonestly; (d) 5 malicious nodes vote all contents dishonestly.

slower than the TS due to the frequent change of node pseudonyms. Thus, introducing the TS can effectively improve the AdContRep’s performance and at the same time enhance the node privacy. In addition, it takes a bit longer time for AdContRep to detect all unfair rating nodes if their percentage is higher.

The influence of the on-off attack investigated by Simulation 4 is demonstrated in Figure 5.3 (b). We test two scenarios: 1) 5 fixed attackers randomly vote one content with honest and dishonest recommendations alternatively, whilst other responding nodes randomly vote one content honestly; 2) increase the number of attackers to 10 nodes. We get similar results to Figure 5.2 (a). We also observe that the system can still achieve good performance with the help of the TS when the on-off attack happens. This is because we decay past recommendations at the TS. The influence of past good or bad behaviors is trivial on current content reputation generation. Additionally, bad voting punishment helps AdContRep easily find potential
Figure 5.3: F measure of nodes and TS regarding malicious node detection performance: (a) unfair rating attack \((thr = 30)\); (b) on-off attack \((thr = 5)\)

malicious nodes. For the purpose of node-server performance comparison, the TS does not issue the blacklist of malicious nodes to each node in the simulations.
Chapter 6

Conclusions

6.1 Contributions

MANET has been widely used for distributed communications. Various services can be developed based on MANET. However, lack of trust retards the success of many MANET based services (i.e., MANET services). This thesis work proposes AdRep – a reputation system that could apply on various MANET services with privacy concern. We summarize the contributions of the thesis as below.

First, we design AdRep based on a hybrid trust/reputation management framework that supports trust evaluation and reputation generation on any entities involved in the MANET based services. We adopted the trusted server to support more accurate and efficient trust/reputation evaluation and user privacy preservation. We designed a hybrid trust model that can support trust/reputation management in both MANET node devices and the trusted server. Due to the diversity of different MANET services, we designed a context aware trust model that can be adaptively configured based on different service contexts. Our trust model can be applied to implement various reputation systems for different MANET services, by providing concrete instances to the primitive trust model.

Second, we implemented two concrete reputation systems AdChatRep and AdContRep to illustrate our framework and model’s applicability. In AdChatRep, the context module is instantiated by MANET chatting, while the context module of AdContRep is instantiated by MANET content recommendation. We implemented AdChatRep based on Nokia N900 and AdCon-
CHAPTER 6. CONCLUSIONS

tRep on Nokia N810. Both prototype systems have been demonstrated inside Nokia with sound feedback.

Third, we further evaluated both AdChatRep and AdContRep using different methodologies. We launched a survey exploring usefulness, chatting trust influencing factors and reputation visualization UI design of AdChatRep before implementation. After prototyping it, we further conducted a user study based on AdChatRep usage and survey users’ opinions on the usefulness, effectiveness and UI acceptance of AdChatRep. In addition, we also studied user behaviors based on real chatting logs and system records. We achieved sound feedback on AdChatRep from user study participants. For AdContRep, we evaluated its trust and reputation algorithms through Matlab simulations. We proved the effectiveness of our algorithms [50] and their robustness against unfair rating attacks and on-off attacks.

6.2 Discussions

MANET services cover a broad area and this thesis is limited within only a few use cases and scenarios. The main idea is not to provide one concrete fit-in-all reputation system but a general reputation system and apply it with practical solutions, which should be usable and deployable. This thesis work focuses on solving some practical issue instead of covering all aspects of MANET, e.g., partitioning and poor reliability.

The methodology used for evaluating a reputation system is another issue worth discussing. Evaluation on the reputation system itself and its supported MANET service is interrelated. In the user study, we found that participants seldom separately considered reputation system and chatting service, but tended to express their opinions on both by treating them as an integrated system. For example, the UI design of chatting also influenced their opinions on AdChatRep. The user study aims to evaluate the effectiveness of the whole system and its supported system, thus complicated because the results are influenced by many subjective factors of participants and context variables. Nevertheless, evaluating algorithms of AdContRep by simulation is straightforward targeting only on the effectiveness of applied algorithms. In a nutshell, the two methodologies applied represent different philosophies: to evaluate in a complicated real life to get the results closer to the reality and in a simple simulated environment to predict the system's performance in a limited number of situations. Probably, this explains why
CHAPTER 6. CONCLUSIONS

Netflix Prize [9] provides a large sample collected from real life and encourages people from all over the world to design the best algorithms for movie recommendation.

Regarding the users' motivation to give feedback to the trusted server, people generally behave for their greatest benefits under most circumstances. Without a strong motivation, users generally would not like report their feedback to the server. The problem of lacking feedback motivation also exists in many other reputation systems such as eBay. One consideration is to award users who are active in giving honest feedback. However, this issue is still open for discussion.

One limitation of our systems lies in the conflict between security and privacy. We have not covered much about security issues in the thesis. Privacy is meaningless without security assurance. Therefore, we assume that security solutions are already available in both systems. Actually, many security assurance solutions should be applied in order to deploy our reputation systems, for example, authentication for node-server communication, authorization and access control on user private data at the trusted server, and key management in AdChatRep and AdContRep for securely enabling our designed system, etc.

6.3 Future Work

For future work, we can continue along several directions. First, both AdChatRep and AdContRep require more functions and implementation. For a real trial or product, more work is needed. Second, many research issues regarding reputation framework and trust model for pervasive social computing are still open and worth our further study. In addition, some participants in user study suggested us expanding the reputation system for more MANET services such as new board games and dating. We plan to develop more interesting services based on the current work.
Bibliography


[50] ZHENG, Y., AND YU, C. AdContRep: A privacy enhanced reputation system for manet content services. Submitted for publication.