OMiN - Opportunistic Microblogging Network

Neil Wells, supervised by Tristan Henderson

Abstract

OMiN is a pocket switched network running on smartphones. It allows users to send and receive microblog messages without using any global infrastructure such as the internet. Smartphones in close proximity to each other pass on messages according to a variation of the PROPHET routing protocol. Steps have been taken to verify the security of the network by protecting it against known attack vectors.

Declaration

I declare that the material submitted for assessment is my own work except where credit is explicitly given to others by citation or acknowledgement. This work was performed during the current academic year except where otherwise stated.

The main text of this project report is **NN,NNN*** words long, including project specification and plan.

In submitting this project report to the University of St Andrews, I give permission for it to be made available for use in accordance with the regulations of the University Library. I also give permission for the title and abstract to be published and for copies of the report to be made and supplied at cost to any bona fide library or research worker, and to be made available on the World Wide Web. I retain the copyright in this work.

Contents

Introduction

TODO

Objectives

Primary Objectives

- design and implement a protocol for discovering nodes in close proximity and passing messages and necessary metadata between them.
- Create a core library to manage message storage and routing.
- Implement a simple epidemic routing algorithm to send messages to all available nodes.
- Design a routing algorithm using user metadata to route messages while disguising message content and metadata.

Secondary Objectives

- Create a smartphone UI.
- Implement a more advanced routing algorithm.
- Design and implement a mechanism to decide whether a node is trustworthy or not.
- Evaluate the performance of the implemented routing algorithms.

Tertiary Objectives

- Compare the real world vs simulated performance of the routing algorithms.

Context Survey

The following provides brief summary of the current state-of-the-art in opportunistic network technology. Only the most relevant subjects will be addressed in order to give the reader sufficient background information to fully understand the project.

Opportunistic Networks

An opportunistic network is a network where connections between nodes are sparse and a direct path from source to destination cannot be guaranteed. For this reason, opportunistic networks must be able to store messages and forward them when connections become available. Problems that have been solved in conventional connected networks (security, routing etc.) are much harder to solve in opportunistic networks.

A Pocket Switched Network (PSN) is a form of opportunistic network where people are represented by network nodes (commonly a smartphone). Much research has been done in developing routing algorithms and security mechanisms for PSNs based on the predictable nature of human behaviour.

Similar Projects

Haggle

One of the largest opportunistic platforms is Haggle (http://www.haggleproject.org)[1], a pocket switched network designed to run on smartphones. There are implementations for a number of clients including Android (play.google.com/store/apps/details?id=org.haggle.kernel) and Windows Mobile.

By monitoring use of the platform, the authors discovered trends in inter-contact times and contact durations, showing that existing algorithms are poorly suited to real world models[2].

FireChat

FireChat (opengarden.com/firechat) is a new smartphone application used for off-the-grid messaging between nearby users. It has been used to circumvent government restrictions in Iraq (http://www.theguardian.com/technology/2014/jun/24/firechat-updates-as-40000-iraqis-download-mesh-chat-app-to-get-online-in-censored-baghdad) and during the Hong Kong protests (http://www.theguardian.com/world/2014/sep/29/firechat-messaging-app-powering-hong-kong-protests).

However, the app mostly relies on an internet connection, and its simple protocol is insecure (http://breizh-entropy.org/~nameless/random/posts/firechat_and_nearby_communication) and unable to implement the store-and-forward functionality of a proper opportunistic network.

SWIM

The Shared Wireless Infostation Model (SWIM) proposes using an opportunistic network to monitor whales[3]. Small nodes are attached to the whales, which record data such as location and interaction with other whales. Connected nodes transfer this data between each other. Whenever data is transferred to a base station (the paper proposes using seabirds), it can be collected and stored.

Because data is shared between nodes, it is no longer necessary to find a whale with a sensor in order to acquire data from that sensor. This is a perfect example of the power of opportunistic networks in an environment with very limited connectivity.

Routing Algorithms

Routing in opportunistic networks can be reduced to the problem of finding a destination node in a constantly changing graph. Most routing algorithms therefore build on existing graph search techniques. However, because the graph is constantly changing and is not necessarily random, such techniques are not necessarily the most effective (as shown by the Haggle project).

Context Based Routing

Context based routing is a form of greedy searching, where a single message is continually passed to the node most likely to reach the destination. While it is not guaranteed to find the optimum path to the destination, it uses very few resources as there is only one message being stored.

Epidemic Routing

The opposite of context based routing is epidemic routing - a form of uniform cost search[4]. Copies of the message are passed at every opportunity until the network is saturated. This is often likened to the spread of a virus. While this approach will always find the optimal path (because it takes all possible paths), it is very resource intensive - all nodes are expected to store every possible message. For this reason, routing protocols that use similar techniques (dissemination based routing) concentrate on avoiding unnecessary use of resources.

PROPHET

The Probabilistic Routing in Intermittently Connected Networks (PROPHET) algorithm[5] uses a form of the A* algorithm. A utility function (derived from recent encounters with nodes) is used to predict whether the message should be passed on

to a node. This use of heuristics greatly improves performance when compared to epidemic routing.

Bubble RAP

The Haggle project discovered that algorithms that treat routing as a generic graph search problem are often unsuited to PSNs. Bubble RAP[6] works on the idea that a social connections graph has tree like structure, where close nodes form a community. In order to send messages to a different community, the message is moved up the tree towards highly connected nodes near the root, and then back down the tree towards the destination community and, eventually, the destination node.

This has been shown using the data collected from Haggle to be more effective than standard routing algorithms.

Security

Opportunistic networks (especially pocket switched networks) must consider the possibility of an unsecure node which does not conform to the protocol specification (either through accident or design) or attempts to discover private information. Common attack types include:

- Sybil attacks: impersonating another node in order to send messages that appear to be from that node or to receive messages intended for the node.
- Majority attack: by controlling a large number of nodes, an attacker can control a network which assumes that the majority of nodes can be trusted.
- Snooping: gathering information such as message metadata to discover private information such as message contents and user location.
- Selfishness: failing to follow the protocol in order to reduce resource usage (e.g. failing to pass on messages).

Trust Based Security

Trust based security mechanisms depend on generating a list of trusted or untrusted nodes. This is commonly based on trusting 'friends' in a social network[8] or distrusting nodes exhibiting strange behaviour[9]. While trust based security mechanisms can lower the chances of an attack, they are not infallible.

Certificate Based Security

In order to increase security, a security mechanism must reduce the number of trusted parties to an absolute minimum. This can be done using an infrastructure of

public key certificates to verify the identity of a node. However, the problem of distributing this infrastructure without trusting an arbitrary number of nodes is unsolved. On the internet, it is done by trusted certificate authorities, who manage and release certificates. However, having a central trusted authority is infeasible in an opportunistic network, as all nodes would have to connect to it directly at some point. Some mechanisms, such as [11] do use a central server, but only require it to be available for nodes joining the network. Mechanisms for distributed certificate distribution require some level of trust in network nodes. For example, [10] does this by building a graph of certificates determining who trusts who. Any abnormalities in the trust graph may indicate foul play. However, this scheme is still vulnerable to a majority attack.

Requirements Specification

User Requirements

Non-Functional Requirements
- High: The user shall be able to create a unique identity.
- High: The user shall be able to send text messages to all others who follow the user or a hashtag in the message.
- High: The user shall be able to 'follow' any user and receive messages sent by that user.
- Low: The user shall be able to 'follow' any hashtag and receive messages containing that hashtag.
- Low: The user shall be able to send encrypted direct messages to a single user.
- Low: The user shall be able to send multimedia messages to all others who follow the user or a hashtag in the message.

Functional Requirements
None

System Requirements
Non-Functional Requirements
- High: The system shall work on smartphones or tablets capable of connecting to a wifi network.
- High: The system shall allow creation of user identities with a unique cryptographic identity.
- High: The system shall automatically connect to nearby nodes and pass on relevant information such as messages.

- Medium: The system shall provide a mechanism for securely distributing the cryptographic identity of a user.
- Medium: The system shall protect user metadata such as location and friends list from all other nodes.
- Medium: The system shall ensure that messages cannot be modified in transit or that such modifications can be detected.
- Medium: The system shall ensure that nodes cannot send a message that appears to be from another user.
- Medium: The system shall be robust and able to continue functioning when it encounters an unexpected state such as a malfunctioning or untrustworthy node.
- Medium: The system shall ensure that encrypted direct messages cannot be read by third parties.
- Low: The system shall be able to support multiple user identities on a single node.

Functional Requirements
- High: The system shall collect anonymous logging data for debugging and profiling purposes.
- Medium: The system shall ensure that a reasonable number of messages reach their intended recipients.
- Medium: The system shall restrict the size of the message buffer by evicting messages.
- Medium: The system shall use a minimal amount of the available power.

Software Engineering Process

TODO

Ethics

In order to test the real world performance of the network, we may ask people to use the application. In this case, some metadata will be collected on users, with their consent. This may include:

- An anonymous user ID.
- Anonymised 'Friends list' (or equivalent) of users.
- Times and locations of encounters between anonymous users.

- Metadata of messages passed during encounters, including message ID and origin ID, but NOT message contents.

Design

TODO

Implementation

TODO

Evaluation and Critical Appraisal

TODO

Conclusions

TODO

References

[1] Scott J, Crowcroft J, Hui P, Diot C, Others. Haggle: A networking architecture designed around mobile users. In: WONS 2006: Third Annual Conference on Wireless On-demand Network Systems and Services; 2006. p. 78-86.

[2] Chaintreau A, Hui P, Crowcroft J, Diot C, Gass R, Scott J. Impact of human mobility on opportunistic forwarding algorithms. Mobile Computing, IEEE Transactions on. 2007;6(6):606-620.

[3] Small T, Haas ZJ. The Shared Wireless Infostation Model: A New Ad Hoc Networking Paradigm (or Where There is a Whale, There is a Way). In: Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking &Amp; Computing. MobiHoc '03. New York, NY, USA: ACM; 2003. p. 233-244. Available from: http://doi.acm.org/10.1145/778415.778443.

[4] Vahdat A, Becker D, Others. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University; 2000.

[5] Lindgren A, Doria A, Schelén O. Probabilistic Routing in Intermittently Connected Networks. SIGMOBILE Mob Comput Commun Rev. 2003 Jul;7(3):19-20. Available from: http://doi.acm.org/10.1145/961268.961272.

[6] Hui P, Crowcroft J, Yoneki E. Bubble Rap: Social-based Forwarding in Delay Tolerant Networks. In: Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing. MobiHoc '08. New York, NY, USA: ACM; 2008. p. 241-250. Available from:http://doi.acm.org/10.1145/1374618.1374652.

[7] Pelusi L, Passarella A, Conti M. Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. Communications Magazine, IEEE. 2006 Nov;44(11):134-141. Available from: http://dx.doi.org/10.1109/MCOM.2006.248176.

[8] Trifunovic S, Legendre F, Anastasiades C. Social Trust in Opportunistic Networks. In: INFOCOM IEEE Conference on Computer Communications Workshops, 2010; 2010. p. 1-6. Available from: http://dx.doi.org/10.1109/INFCOMW.2010.5466696.

[9] Bigwood G, Henderson T. IRONMAN: Using Social Networks to Add Incentives and Reputation to Opportunistic Networks. In: Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on; 2011. p. 65-72. Available from: http://dx.doi.org/10.1109/PASSAT/SocialCom.2011.60.

[10] Capkun S, Buttyan L, Hubaux JP. Self-organized public-key management for mobile ad hoc networks. Mobile Computing, IEEE Transactions on. 2003 Jan;2(1):52-64. Available from: http://dx.doi.org/10.1109/TMC.2003.1195151.

[11] Shikfa A, Onen M, Molva R. Bootstrapping security associations in opportunistic networks. In: Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on; 2010. p. 147-152. Available from:http://dx.doi.org/10.1109/PERCOMW.2010.5470676.

Appendices

Appendix A - Testing Summary

TODO

Appendix B - Status Report

TODO

Appendix C - User Manual

TODO

Appendix D - Maintenance Document

TODO