Ubiquitous Interpersonal Communication over Ad-Hoc Networks and the Internet

Edoardo Biagioni
University of Hawaii at Mānoa
esb@hawaii.edu

***Abstract*—*The hardware and low-level software in many mobile devices are capable of mobile-to-mobile communication, including ad-hoc 802.11, Bluetooth, and cognitive radios.*

We have started to leverage this capability to provide interpersonal communication both over infrastructure networks (the Internet), and over ad-hoc and delay-tolerant networks composed of the mobile devices themselves.

This network is decentralized in the sense that it can function without any infrastructure, but does take advantage of infrastructure connections when available. All interpersonal communication is encrypted and authenticated so packets may be carried by devices belonging to untrusted others. The decentralized model of security builds a flexible trust network on top of the social network of communicating individuals.

This social network can be used to prioritize packets to or from individuals closely related by the social network. Other packets are prioritized to favor packets likely to consume fewer network resources.

Each device also has a policy that determines how many packets may be forwarded, with the goal of providing useful interpersonal communications using at most 1% of any given resource on mobile devices.

One challenge in a fully decentralized network is routing. Our design uses Rendezvous Points (RPs) and Distributed Hash Tables (DHTs) for delivery over infrastructure networks, and hop-limited broadcast and Delay Tolerant Networking (DTN) within the wireless ad-hoc network.

I. Introduction

In the days of Plain Old Telephone Systems (POTS), a company would buy or rent an expensive PBX to connect the telephones in its offices to the worldwide telephone network.

Today many individuals own an inexpensive packet switching system to connect multiple computers to the worldwide Internet.

Some people have already started to use smartphones or other mobile devices as personal hotspots, obtaining Internet access without having to purchase a separate piece of equipment. At present, the performance of these hotspots is very limited, and the hotspot itself must have access to the Internet.

This paper describes ad-hoc technology to extend the reach of personal hotspots both further from the Internet and to a wider group of people than just the owners of hotspots, to the point where people are able to communicate even without the Internet. The goal is to to provide low-bandwidth interpersonal communication and peer-to-peer social networking without regard to availability of infrastructure or ability or willingness to pay for a commercial service.

Ad-hoc technology is inefficient and unreliable compared to today's Internet. The advantage of ad-hoc and peer-to-peer technologies is that to work they only need two or more suitable general-purpose devices. The project described here, AllNet, is designed to take advantage of the Internet and other infrastructure when available, and use ad-hoc and delay-tolerant networking to continue delivering packets when Internet access is not available.

Any project that accomplishes these goals is likely to share these features:

- Support for mobile devices. These devices are widespread, wireless and self-powered, and can be used (for a limited time) even without infrastructure.
- Distributed network access. It must be possible to join and use the network without permission or approval from a central authority. In case of emergencies, centralized operations ("registering") may be difficult or cause unnecessary delay.
- Usefulness at low bit rates and high latencies. When high-speed networks are available, they can be used, but at other times, the network should still be useful.
- Security. Ad-hoc technology uses untrusted intermediaries to deliver packets, so all *personal* communication must be encrypted end-to-end. *Public* communications can be sent in the clear.
- Authentication. Once my device knows my contacts' public keys, it can easily tell me whether a signed packet is from one of them or not.
- Social Network. Pervasive authentication of known contacts makes it possible for my devices to keep track of my social network.

The desire to support mobile devices with distributed network access suggest that, unlike the current Internet, addressing should not be based on the point of attachment to the network. In AllNet, addresses are self-selected bitstrings, often the hash of a phrase meaningful to the owner of the device. Packet delivery is by a combination of limited wireless broadcasts, sending to designated Internet hosts, and

network nodes self-organizing into Distributed Hash Tables (DHTs).

While multiple devices might by chance (or maliciously) select the same random address, in AllNet such collisions just mean that the packet might be physically delivered to multiple destination devices. Since personal packets are encrypted, delivery to multiple destinations does not allow eavesdroppers to read the contents of a packet. For accidental collisions, the receiving device automatically treats any packet it cannot decrypt as any other packet not intended for this device.

To be useful even when network performance is low, one of the applications of AllNet is a persistent chat system similar to SMS. SMS already provides delay tolerant low-bandwidth communications, but requires cellular infrastructure and is sometimes unavailable to individuals because of the way it is priced. With AllNet, such communication would take place over the Internet when available, and by ad-hoc and delay-tolerant networking when these are available.

For example, an AllNet chat message might be delivered in a fraction of a second if both devices are connected to the Internet. The recipient either has a public (routable) IP address, or requests packet forwarding from another computer that does have such a public IP address. In either case, this public IP address must be known to the sender. If the sender does not know an IP address for this receiver, the sender forwards the packet to any node forming an Internet-wide DHT. If the recipient has requested packet delivery from DHT nodes corresponding to its address(es), the packet is delivered promptly.

If ad-hoc networking is available, the same packet will also be sent to the devices within reach of the sender. Each such device forwards the packet if that can be done within strict limitations on battery and bandwidth usage. If the destination can be reached through ad-hoc networking, the packet might be delivered in less than a second, or after a delay of many minutes if one or more of the intermediate and final devices turn off their radios part of the time.

Finally, the device will store the packet for a limited time, and make it available on request. The buffer size is limited, with priority given to packets for destinations known from the social network and, all else being equal, to newer packets. If a packet is still present in the device when the packet's destination is in range, the packet is delivered at that time, whether that be a few seconds or a few days later.

To be useful with such delay variability, each chat message is tagged with a unique sequence number and the time of transmission. The destination chat program uses the sequence number and timestamp to discard duplicate packets. Reusing a sequence number with a later timestamp allows the sender to request that a message be amended or deleted. The recipient is free to either honor or disregard such requests, but normally the chat program shows the latest version with an indication that older versions are available.

This chat protocol is the first applications of AllNet. Another attractive applications would allow devices within range of each other to be used as Wi-Fi walkie-talkies. Here there are no bandwidth limitations, but the maximum distance between devices is limited.

AllNet provides a way for two people who are within wireless range to securely exchange public keys. The basic mechanism is to transmit in the clear the public key, together with an HMAC of the key and of a short secret string that the two parties have exchanged. Communicating the secret string is easy if the two can talk directly to each other. Otherwise, the secret string can be sent through a trusted third party or by other, relatively secure mechanisms such as telephone calls.

We have designed and developed AllNet beginning in the first half of 2012. Although the design is still evolving, two preliminary Linux versions (version 0 and version 1) have been completed, and the implementation of version 2 is underway. We expect that version 2 will be sufficiently useful to see initial use among the public at large, and we plan to port this implementation to a number of mobile platforms.

The next section is the main section of this paper, and gives details of the design of AllNet. Section III summarises the current performance and other interesting features of AllNet. Section IV surveys related work, including a previous paper on AllNet, other projects that overlap with AllNet, and technologies that the AllNet design builds on. Section V reviews present status, future work, and gives concluding remarks.

II. Design

The design of AllNet includes a number of components. We begin by describing (Section II-A) the packet forwarding algorithm, a novel and simple mechanism that combines existing approaches to make AllNet effective at delivering data under a variety of circumstances.

AllNet is specifically designed to keep resource usage below a specific, very low level for traffic that does not directly benefit the owner of the device or the owner's friends. Essential components of this design include a low power wireless forwarding algorithm (Section II-B) and a packet prioritization scheme (Section II-C). To distinguish friends and assign them a greater share of resources, AllNet provides an algorithm for keeping track of the social network of the owner of the device and anonymously sharing it with others (Section II-D).

Finally, AllNet packets are designed to be public-key encrypted. Rather than using certificate authorities or a web of trust, the key exchange mechanism of AllNet (Section II-E) allows the exchange of keys among individuals who know each other or a common friend.

A. Addresses and Message Forwarding

AllNet transmission combines wireless ad-hoc broadcasting with Internet transmission.

Every packet carries a destination ID. These destination IDs are bitstrings up to 8 bytes long. AllNet assigns no meaning to these destination IDs, which are self-selected by each device.

Should two people select the same ID, resulting in a collision, the only consequence is that they might attempt to decrypt each other's packets. The decryption will not succeed, but the attempted decryption will waste some energy.

Every ID is sent with a one-byte stating the number of valid bits in the ID. More bits in the ID allows a packet to be delivered more precisely to its destination, using fewer network resources and therefore giving the packet higher priority. Fewer bits lets the sender frustrate any efforts at traffic analysis. In AllNet the sender of each packet makes this tradeoff.

A destination ID is used as an index into the Distributed Hash Table (DHT), and is used to identify and prioritize traffic for this device or for other devices known through the social network.

Each device generating or forwarding a packet:

1) broadcasts it on locally connected networks
2) broadcasts it to listeners
3) sends it to the DHT node(s) corresponding to the destination ID
4) sends it to any rendezvous points (RPs) it knows for this destination ID

In each case, the forwarding is subject to AllNet resource limitations. As long as these limitations are not reached, each packet is forwarded to all local nets, to all listeners, to all DHT nodes and to all RPs corresponding to the destination ID. Otherwise, only higher-priority packets are forwarded, as described in Section II-C.

A listener establishes a TCP connection another AllNet node. A node wanting to receive packets, on the Internet but not itself part of the DHT, and perhaps behind a firewall, might listen to several of the DHT nodes responsible for the parts of the address space corresponding to the node's own destination addresses.

Well-behaving nodes that are part of the DHT both receive the packets themselves (handing them to local applications), and forward the packets to each listener in accordance with step 2. In this way, a node that is not in the DHT may receive all its packets by being a listener to a DHT node corresponding to one of its IDs.

Rendezvous points (RPs) are only used when a device can send Internet packets directly. The sender must have been given the IP and port number of a machine with a stable, publicly routable IP address that the receiver will connect to to retrieve its packets. The machine with that IP address is the RP. A functioning RP will forward packets to the receiver, either by prior agreement, or by the receiver connecting to the RP as a listener.

In general, the usage of stable RPs is preferred to using the DHT, and more so if the RP is under the control of personal friends. RPs somewhat resemble mail servers, allowing communication between a sender and a receiver that may not have stable IP addresses or even be consistently connected to the Internet.

If an ack is sent in response to a received packet, the ack may carry, encrypted, the IP and port of the RP from which the packet was first received. This may be used by the sender when sending further packets to the same destination, prioritizing RPs deliver quickly.

Listening to nodes in the DHT corresponding to my address is useful as a backup when no RPs can be identified, and is essential when first connecting to AllNet. It is the way any node can pick an arbitrary string and automatically have a public routable address.

The design of the DHT is modeled after the DHT in Kademlia[1].

B. Sleep and Wake Cycles for Wireless

There are different kinds of traffic in AllNet. One of the important distinctions are between traffic that I (the owner of the device) have originated, traffic sent by my friends, and other traffic.

In the first case, there are few if any resource limitations. The radio is on whenever I wish to send packets, and also whenever I can predict that I will receive a message I am interested in.

For example, a cellphone walkie-talkie application could keep the radio on at all times, possibly discharging the battery relatively quickly to support low-latency and high-bandwidth communication among devices that are within range. But when my device is only forwarding packets on behalf of others, I am likely to want to limit the amount of battery energy used for forwarding.

One goal in the design of AllNet is that the network be useful even if resource usage is limited to about 1% of the total available on each device. Recognizing that wireless transmission and reception may consume significant power, this means keeping the radio off (or available for other uses) 99% of the time. More in general, we consider duty cycles (fraction of the time that the radio is on) of p or less, so the radio is off for fraction $1 - p$ of the time.

AllNet follows the general scheme of the Block Transfer Protocol [2] by synchronizing senders and receivers, then sending multiple packets one after another once the sender and receiver have synchronized.[1]

[1]A simpler unsynchronized scheme has receivers listen fraction p of the time and senders send each packet $1/p$ times. This is inefficient for small p.

To synchronize, any device with packets to send (a sender S) listens for announcements from other devices that wish to receive packets (receivers R). Once S has heard from R, a brief exchange similar to RTS/CTS provides some protection against collisions. S can then send a number of packets to R. Other receivers within range may also receive the same transmission.

If R takes time t to turn on its radio, transmit its announcement, wait for a reply, and if nothing heard, turn the radio off again, a duty cycle at most p requires R to turn off its radio at least time t/p between announcements. S listens for at least time t/p to reliably hear announcements.

A sender can tell whether packets in its queue have high priority. For such packets a sender may use a duty cycle $p' \geq p$, where $p' = 1$ for the sender's own packets, and $1 \geq p' \geq p$ for packets sent by the sender's friends.

If the sender listens for t and turns off the radio for time t_o to give an overall duty cycle of $p' = t/(t_o + t)$, this gives $t_o = t(1 - p')/p' \approx t/p'$. Then the worst-case one-hop latency for a packet sent to a neighbor within range is $lat_{worst} \approx t/(pp')$.

C. Message Prioritization

When traffic is light, AllNet eventually forwards all packets it receives whose hop count has not expired.

When traffic is heavy, senders must decide whether and when to send packets. In AllNet every device prioritizes the packets it sends, with highest priority given to the packets the device's owner wishes like to send, and descending priority to packets for the owner's friends and then packets that benefit the network as a whole. This last is hard to determine, but AllNet suggests heuristics to favor some kinds of packets over others. None of these heuristics are required for participation in AllNet, but supporting them may improve the performance of the network as a whole.

1) Priority Computation: AllNet automatically prioritizes packets based on local information available to the forwarding node.

The priority is a real number between 0 and 1, internally represented in fixed point notation as an integer between 0 and 2^{30}.

To favor packets generated by the local system, local applications are allowed to specify their own priority for outgoing traffic. If not specified by the application, the priority of local packets defaults to $0.875(7/8)$.

Similarly, if a packet carries a sender ID and a matching certificate identifying one of the people to whom I have agreed to give resources (a friend), these packets are given a priority of $0.75(3/4)$.

For all other packets, a variety of independent priorities P_i are computed based on different information, then combined by multiplying them together. Since each $P_i \leq 1$, any factor that produces a low individual priority gives a low overall priority for the packet.

$$Priority = \prod_i P_i \tag{1}$$

2) Priority Factor from Social Distance: The social distance d is defined to be $d = 1$ for my friends, $d = 2$ for their friends, $d = 3$ for their friends, and so on. The social network used to keep track of social distances is described in Section II-D.

When the social distance $d > 1$ of the sender is known, it is used to compute a social priority factor $P_s = 2^{1-d}$. This priority drops quickly with social distance. The size of a social network may be expected to grow nearly exponentially with social distance, so the computation of P_s exponentially decreases the priority with increases in social distance. Also, $P_s \leq 0.5$ for $d \geq 2$, so the priority of friends of friends ($d = 2$) is always less than the priority for friends.

If the social distance is not known, and if AllNet keeps track of the social network up to distance $d = n$, the distance used for someone who does not appear in the social network is $d = n + 1$. The current design keeps track of identifiers up to distance $d = 3$, so a stranger is arbitrarily assigned $d = 4$, giving $P_s = 2^{-3} = 0.125$.

3) Other Priority Factors: The remaining P_i factors in equation (1) are based on:

- P_m the maximum number m of times a packet may be forwarded. This field is set by the original sender and never changes. Network resource may increase exponentially with m, so $P_m = 2^{1-m}$.
- P_h the number of hops h already traveled. Packets that have traveled longer distances would need more resources if retransmitted, but have also had more chances to reach their destination. In general it is wiser to favor local traffic, so we use $P_h = 1 - (h - 1)/8$ for $h \leq 4$, and $P_h = 0.5$ for $h > 4$.
- P_b the number of bits b in the destination address. AllNet uses $P_b = 1 - 2^{-b}/2$, which gives $P_b = 0.5$ for $b = 0$, $P_b = 0.75$ for $b = 1$, $P_b = 0.875$ for $b = 2$. This reflects the fraction of recipients who will not attempt to decrypt this packet.
- P_r the rate r at which packets from the same sender as this packet have recently been received. This factor discriminates in favor of known senders from which we have not forwarded many packets recently. Unknown senders have $P_r = 0.5$, whereas a known senders that has recently used fraction r of the bandwidth gets $P_r = 1 - r/2 \geq 0.5$.

These functions use only local information and information obtained from the packet being forwarded.

In the absence of social network information, the priority factors P_m, P_h, and P_b favor and encourage transmission of packets that will consume fewer network resources.

To obtain higher priority (P_s and P_r), a sender must be known from the social network, and have placed into the

packet a digital signature of the packet body, signed with the sender's private key. In our tests, verifying a 128-bit ECDSA signature took 0.3ms on a modern processor, and less than 3ms on an 800-MHz celeron.

The next section explains the design of the distributed social network mechanism of AllNet.

D. Anonymous Social Network

Distributed social networks have been and continue to actively be developed, including for example Diaspora [3], Friendica [4], and DiSo [5] (Distributed Social Network). As for AllNet, the goal in these networks is to foster decentralized interpersonal communication.

Another goal of AllNet is to to allow devices to determine the degree of connectedness within the social network and the extent to which to devote limited resources to forwarding each packet. This should not require revealing to others the identity of my friends.

The social network graph is easily built in a distributed fashion in a manner analogous with link-state routing. When I connect with a new person, Alice, I can send her information about individuals in my social network, including Bob, Charlie, Donna and Eve, who are in the set f_{me} of my friends.

The information I send includes destination addresses and public keys used to verify packets sent by the people in my social network. For example, it includes the address ID_B that I (and perhaps others) use when exchanging packets with Bob, and a corresponding public key PK_B that can be used to verify whether a packet is from Bob. The information does not include Bob's name or other personally identifiable information.

If Alice already has contact information for Bob, she can tell that Bob and I are in each others' social network. If she does not know Bob, she has a key that she can associate with one of my friends, but without knowing who that friend is. In general, she can add the set f_{me} into the set of her friends of friends, f_A^2. If I send her information about the set of my friends of friends, f_{me}^2, she can add the information to her set of friends of friends of friends, f_A^3.

When I send Alice information about f_{me}^2, I send her only the initial few bits of each destination address, which she adds in her f_A^3. For example, this may include the first few bits of the destination address used by Bob's friend Frank, who is in f_{me}^2. I don't know Frank, but if he sends a packet through my device, my device can tell that he is in f_{me}^2, and give the packet the corresponding priority. Alice can prioritize the message knowing that Frank is in her f_A^3.

Each AllNet device keeps track of f, f^2, and f^3. Assuming each person averages a few hundred contacts, each device only needs to store a few million contacts.

It is possible [15] to leverage this information to establish social distance beyond $d = 3$.

E. Secure Public Key Exchange

On the World Wide Web, secure exchange of public keys is mediated by a centralized Public Key Infrastructure (PKI) that depends on a number of trusted certificate authorities (CAs). This PKI is used routinely and is extremely reliable as long as the CAs can indeed be trusted, which unfortunately is not always true [8].

The Web of Trust, introduced by Zimmerman for PGP [9], allows individual users to certify other users. A recipient Romeo of a public key alleged to be from Juliet may trust that this is indeed Juliet's key if the key is signed by Mercutio or Benvolio, whom Romeo trusts to certify keys. While the decentralized nature of the Web of Trust makes it perfectly suitable for AllNet, this model is still somewhat more heavyweight than needed within a social network. Specifically, each certification alleges that the person is indeed who they claim to be.

In a true social network, people generally know each other informally, and frequently have out-of-band ways of exchanging information.

For example, when Romeo and Juliet met at a social event, they were able to exchange contact information using only their voices, rather than a secure network. Juliet never checked Romeo's ID, so she may trust him with her love, but perhaps not with her money. When Juliet later learns that Romeo is a Montague, she might still continue to trust him and love him, yet change her behavior in other ways, such as not meeting him in public. It is challenging to use the Web of Trust correctly because human trust is very nuanced, evolving, and implicit. This is hard and tedious for people to encode in computer software.

If Romeo and Juliet were using AllNet to communicate, when they met at they exchanged the secret string s randomly generated by Juliet's device. Romeo enters s into his device, which sends his public key PK_R and a Hash-Based Message Authentication Code (HMAC) $H = HMAC_s(PK_R)$. The HMAC is different for different strings s, so Juliet can tell that this is Romeo's public key even though all Juliet and Romeo know in common is the secret string s.

Juliet's device, on receiving a key exchange packet (PK', H'), can verify whether $HMAC_s(PK') = H'$. If so, Juliet can be confident that $PK' = PK_R$ is Romeo's public key. Juliet's device then sends a response encrypted with PK_R. This response carries her own public key PK_J, $HMAC_s(PK_J)$, and may also carry a user profile, security information, and information about her social network.

The string s can be short and easily communicated if there is little risk of accepting a packet from an attacker, for example when a device is only accepting wireless packets with a hop count of 1. When exchanging the key over the Internet, s must be long enough to prevent random attackers from successfully having their keys accepted. Six characters

may be enough in the first situation, a dozen or more is needed in the second case.

Without s, Romeo's rival Paris is unable to convince Juliet's or Romeo's devices to accept his public key, and therefore unable to pretend to either that he is the other. Even if Paris learns s at a later date, once the exchange is complete, neither device will accept a new public key.

When Mercutio needs to communicate with Juliet, Romeo can forward Mercutio's public key to Juliet and Juliet's public key to Mercutio. This is similar to the PGP Web of Trust.

III. Performance and Evaluation

A. Latency

Since AllNet is designed primarily for traffic (such as chat that normally requires low bit rates, latency is more important than throughput. Latency varies dramatically depending on the circumstances of the communicating hosts. Each of the next sections considers a different scenario.

Exchange of packets over AllNet was tested under different circumstances, including two clients on the same physical host, two clients on hosts connected by the Internet, two clients connected via an intermediate host across the Internet, and wireless ad-hoc connection between two clients. The results for all but the ad-hoc communication are predictable, with latency within a few milliseconds of that returned by `ping`.

For wireless ad-hoc communications, we measured the time to turn the wireless interface on and off. The current implementation has yet to be optimized. For example, the interface is placed into ad-hoc mode by calling `iw` and `ifconfig`. In 10 tests on a modern system with a 2.5GHz dual-core pentium E5200 running 64-bit Ubuntu Linux 12.04, calling `iw` and `ifconfig` to turn on the interface and place it in ad-hoc mode, then turning the interface off again, took between 98ms and 190ms, with a mean of 160ms and a median of 164ms.

Using the median 164ms and the target of $p = 1\% = 0.01$ of keeping the radio on, the analysis in section II-B suggests a receiver sleep for 16.4s between announcements. Senders wishing to hear announcements keep their radios on for the same length of time hear the announcement. A sender wishing to keep the same 0.01 duty cycle will then sleep $t/p^2 = 1,640$s, or about 27 minutes. This is the maximum one-hop latency among connected systems on the wireless ad-hoc network.

B. Efficiency

Broadcast is seen as an inefficient technology because it delivers packets to those that have no use for them. But broadcast also has advantages, including that addressing is optional. A security benefit of broadcast is that a packet is delivered to its destination without revealing which of the receiving nodes is the destination.

Ad-Hoc networks are inefficient because each packet is retransmitted on the same medium, and because multiple nodes may retransmit each packet.

AllNet puts a limit on how many resources will be used to forward packets. Even if efficiency is low, at worst this limits the number of AllNet packets that are successfully sent, rather than affect the resources (battery and spectrum) used for AllNet.

If AllNet is used mostly for interpersonal communication of text messages, the inefficiency is not likely to be a concern.

C. Security Considerations

All user data in AllNet that is not sent in the clear, is encrypted and signed, so the data is kept confidential and the recipient knows that the sender has a private key corresponding to the public key of one of its contacts. Unless the keys are compromised (or the algorithms are broken), the data is secure. AllNet applications currently use 4,096-bit RSA keys and AES for longer messages, but this choice can be changed easily and without impacting the underlying implementation of AllNet. Only recipients need to decrypt a message, so any algorithm that is used by both sender and recipient can be used over AllNet.

Good key management practices require that keys be backed up securely. For AllNet, we plan to allow one device to advertise as its own multiple keys, including both multiple keys on the same device, and also other keys on other devices belonging to the same owner. Anyone sending to this owner would normally send to all the owner's devices. Recipients of packets signed by any of the keys can trust that the packet is sent by the owner of all the keys. In case one device is compromised, the other device(s) can still send secure and authenticated packets to the owner's contacts, informing them of the situation and helping to alleviate any problems.

Data that is sent in the clear is not kept confidential. While further experience is needed, data sent in the clear will normally be ignored by users. Exceptions include messages signed by a recognized authority using a known key, or users searching for nearby businesses, emergency situations, or other reasons for wanting to read packets from strangers.

Assuming that these mechanisms function as designed, remaining security challenges include traffic analysis and denial of service attacks. The wireless medium is particularly vulnerable to these attacks, since attackers can overhear or inject traffic without a physical connection. Traffic analysis has already been discussed in Section III-B.

Denial of service is in some ways the opposite of traffic analysis. The denial of service attack may be stopped if the source of the attack is found, so an attacker is likely to want to send untraceable packets. However, AllNet forwards these packets only after higher priority traffic has been sent, giving the attacker a choice between effectiveness and

untraceability. As a result, an effective denial of service attack probably requires, as it often does at present, taking over devices belonging to others. With AllNet, even this is not a very good strategy for the attacker, since the recipient of the attack can use authentication to identify the sending device.

One final security consideration is the concern that if mobile devices are used as identification and keys to access resources, they become more valuable targets of attack. Manufacturers of mobile devices and mobile operating systems have been making progress in securing devices, and continued progress may be expected.

D. Ethical Considerations

Any powerful technology, including encryption, can be used for positive as well as negative purposes. Encryption is widely used for defense and offense by people such as bankers, criminals, whistleblowers, terrorists, military, and human rights campaigners.

To the extent that AllNet is successful and properly implemented and used, it will provide a large number of individuals with the ability to hold private conversations. It will give recipients the ability to discriminate based on the sender of packets, if known, or based on the sender not being known. The only way for even legitimate governments to directly obtain this information would be to run software on the mobile device or obtain the physical device and defeat its security.

Recent news have indicated that powerful organizations may be engaged in widespread eavesdropping by obtaining unencrypted data from the servers of centralized communication systems. Being fully distributed, AllNet is not subject to this kind of eavesdropping. Even those who control the infrastructure may not be able to prevent communication among peers or control the contents of the communication. AllNet enhances privacy, allowing people to communicate with less concern for political considerations, lessens the opportunity to spy on other's communications, and makes it harder to associate packets with people.

Whether this enhanced privacy is beneficial or not depends on the context. A benevolent government or good parents can spy to improve overall and individual welfare, whereas illegitimate governments or abusive parents can use the same powers to oppress people.

One notable effect is that when non-experts maintain their own secure system, the chances of accidental loss of security or loss of data are higher than when experts provide the security.

IV. Related Work

Most of the related work used in the design of AllNet was described in Section II. This section compares AllNet to similar projects. Section IV-E describes a previous publication on AllNet, and outlines the substantial changes made to the project since that time and the substantial differences between this paper and the prior.

A. Ad-Hoc Networks, DTNs, and DHTs

There has been much research on Ad-Hoc Networks, beginning with the early work in MANETs [10] and including fundamental work on Ad-Hoc networks [11]. The design of AllNet builds on these and many more results.

There is a plethora of routing protocols for wireless networks. While this version of AllNet uses broadcasting instead of routing on the ad-hoc portion of the network, future updates may use protocols such as OLSR [12] or AODV [13].

Delay Tolerant Networks [14] have also been the subject of much research, which again the design of AllNet builds on.

The same is true for Distributed Hash Tables [16], [17], [18], [19]. The field of DHTs has had many developments to support active peer-to-peer communities, and many systems used on a daily basis. Although the DHT for AllNet is not yet implemented, we plan to follow the design of Kademlia [1] because of its flexibility and redundancy.

While AllNet builds on previous work in all these areas, it is unique in combining these areas to create a new network with the specific purpose of providing interpersonal communication both with and without the infrastructure.

B. Emergency Wireless Networking

Many people have known for a long time that wireless communications can be useful when the infrastructure fails, and especially in emergencies. Even before the era of digital wireless networks, amateur radio operators provided communications in case of earthquakes or other major disasters. The characteristics of these (often ad-hoc) systems shared by the wireless portion of AllNet include communication over low bit rate channels.

Specific recent projects in this field include Lifenet [20] and the CDAC TERA network [21], [22]. The latter is infrastructure-based and relatively expensive, but has the advantage of working with unmodified mobile devices and being available to relief agencies.

Lifenet, like AllNet, is designed to work well even without the infrastructure yet can use the infrastructure when available. Lifenet also has focused more than AllNet on good routing protocols to support larger ad-hoc networks and higher-bandwidth applications than would work well using only broadcast. In the future, we may adapt AllNet to use this routing protocol, called simply Flexible Routing.

Unlike Lifenet, AllNet focuses on providing at least low-bandwidth communication whenever possible. We also strongly believe that a protocol, to be useful in emergencies, should also be useful in daily lives. Only by having users accustomed to the software and prepared to use it under

normal circumstances will they be able to make good use of it in an emergency.

To support daily use, the design of AllNet has focused strongly on security and the maintenance of social networks. Both security and social networks can be beneficial in emergencies as in daily lives. The development of specific applications is also essential to daily use.

C. Secure Decentralized Networks

Three main efforts towards providing secure, anonymous networks are Tor, Freenet, and Bitcoin.

Tor [23] is designed to provide anonymity and security for Internet access, and does so through onion routing, where messages are repeatedly encrypted and slowly decrypted as they make their way through the network. Routers may issue fake messages to try to defeat traffic analysis.

Unlike Tor, AllNet does not decrypt packets as they are forwarded, instead relying on end-to-end encryption between trusted hosts. Fundamentally, in Tor a user must, to a however minimal extent, trust the routers in the network, whereas in AllNet the user must trust his or her device and the device(s) of the recipient of the message. Trusting the routers might be a good strategy in a fixed network with known routers, but is not likely to work well in a dynamically changing ad-hoc network such as envisioned for AllNet.

Freenet [24] is a content distribution network designed to automatically distribute content while concealing both the source and consumers of the content. These anonymity goals resemble the anonymity goals of AllNet, where a device forwarding a packet may not know where the packet is coming from or who the intended recipient might be. Somewhat similar to the social network in AllNet, Freenet has a Darknet mode where communication is routed through people to whom one has manually set up a connection. Also like AllNet, Freenet uses a system similar to DHTs to locate data, and has a notion of broadcasting requests until they either arrive or time out.

Like Tor, and unlike AllNet, Freenet is designed primarily to be supported by infrastructure networks. Also, Freenet focuses more on storing and delivering immutable data objects rather than facilitating dynamic and time sensitive communication among individuals. Also, Freenet tries to optimize paths to specific destinations by bringing nodes closer together when they exchange information. In AllNet, nodes aware of how they can be reached explicitly communicate that information to other nodes in their social networks.

Intriguingly, the Bitcoin system [25] provides anonymity without resorting to encryption, and provides authentication without any need for personal identifiers.[2] Like AllNet, Freenet, and Tor, Bitcoin is completely decentralized. Like AllNet, data in Bitcoin is time sensitive but some delay is normal. Like all these other technologies, Bitcoin is most effective on systems that are well connected to the global Internet, but unlike these technologies, AllNet is designed to also work well with intermittent or no connection to the Internet.

And while the Bitcoin network can be used to store and communicate arbitrary information, its main purpose is to store and transfer value, and in that, Bitcoin is quite different from AllNet (and Tor and Freenet). The lack of identities and of a social network are further distinctions between Bitcoin and AllNet.

[2]Just as intriguingly, the original author of Bitcoin has managed to maintain his personal anonymity, being known only by the pseudonym Satoshi Nakamoto.

D. Interpersonal Communication Systems

A number of infrastructure-based systems provide human-to-human text based communication using short messages. The two most famous are Short Message Service, also known as SMS or text messaging, and Twitter. Both have been tremendously successful while severely limiting the length of individual messages.

These systems have provided substantial inspiration in the development of AllNet, forever reminding us that even short communications can be extremely useful.

Unlike AllNet, these services require infrastructure, are proprietary, and in the case of SMS, are often quite expensive. For example, a single SMS message requires fewer bits and lower quality of service than one second of voice calling, but is sometimes priced higher than one minute of conversation.

And this is the other inspiration for AllNet: an awareness that such services can be widely available, without the expense and control that sometimes accompanies the need for infrastructure.

E. Previous Publication on AllNet

AllNet was first described at a conference in 2012 [26]. The substantial documentation and the source code (available under a BSD-style licence) have also been posted to the AllNet main web site [27].

The fundamental idea has not changed since then. We are developing a technology to support interpersonal communication over both the Internet and ad-hoc networks of personal mobile devices. However, we have done a lot of work to improve AllNet since that publication. In particular, the earlier paper described version 0 of the protocol, whereas this paper describes version 2. The changes are numerous, and though many are small, some are significant.

Version 2, for example, has a completely redesigned header and address. In version 0 there was no DHT and forwarding was by limited broadcast or to a specific address. The applications had to do source routing, forwarding by the AllNet daemon was very complicated, and addresses were complicated as well. Specific kinds of addresses in Version 0 included IPv4 and IPv6 addresses and several

different addresses for different kind of broadcasts. Version 0 also required senders to know a lot about the topology of the network and how to reach a peer.

Supporting arbitrary bitstrings for addresses, and optional mappings from bitstrings to IP addresses, makes the system more robust and allows for improved performance when better information is available, for example about RPs. Likewise, the DHT will improve performance by reducing the need for broadcasting.

This paper also includes over 6 months' worth of experience with both implementing Version 1 and beginning to implement Version 2, as well testing and learning what works and what doesn't. None of this was available at the time the previous paper was written or presented.

This paper also provides more details about the design of AllNet.

V. Future Work and Conclusion

A. Implementation Status

This paper reflects the design and preliminary implementation of Version 2 of AllNet.

What has been implemented and tested is the AllNet daemon, which forwards packets based on priority both on the Internet and across wireless interfaces. This preliminary implementation includes dropping duplicate packets, supporting listeners and mappings, and forwarding cached packets on demand.

Functionality that has not yet been integrated into the Version 2 daemon includes the DHT, the social network, verifying packet signatures, and keeping wireless interfaces off most of the time. There is a primitive chat client and key exchange program, though at the time of writing, more work needs to be done to make them fully functional.

In contrast, Version 1, which is not interoperable with Version 2, has a chat client and key exchange program. The chat client keeps data persistently and offers a very simple textual interface.

B. Future Work

Future applications of AllNet include text-only browsing and email access, and distributed password-less authentication.

The first two provide web and email access for transfers of relatively few bytes (text only) whenever richer media cannot be supported. Where the web site or email provider supports it, encryption can guarantee that intermediate devices are unable to see the data in transit. Email might be delivered without multimedia attachments.

Once a user's device has keys, and the user has become accustomed to managing them wisely, such keys can be used more widely for authentication, particularly for anything which currently requires a password.

Ideally, it seems feasible to leverage the social network to provide interpersonal incentives to help others. It would be interesting to study how much people participate in AllNet to build a community and its complement of how much people participate for their own benefit, and perhaps, how much the two overlap.

C. Summary and Conclusion

It seems worthwhile to support exchanges of text messages among people whenever that can be done, with or without the infrastructure. Such low bit rate communication can be extremely useful in a number of cases, most dramatically in emergencies, but will only be widely adopted if it is useful for daily communications.

At this point, the design of AllNet is elegant and very effective, even though a lot remains to be done. For example, routing is challenging when networks are decoupled from the network through which the node communicates. Yet this is necessary to support mobile devices. AllNet broadcasts within the local area, with hop count limitation to reduce resource usage, and uses Distributed Hash Tables (DHTs) within the Internet. The DHTs have not yet been implemented.

People often get excited about AllNet, believing that it will lead to a revolution in how everyone communicates. But even if AllNet is wildly successful, ad-hoc networks don't scale well. We will still need, use, pay for, and get the benefits of the infrastructure. We might see evolution in the market for Internet access rather than revolution.

What AllNet can and will do is set a baseline of providing interpersonal communication securely and for free whenever and wherever possible, motivated not by profit maximization but by the desire to help people communicate and to build better communities.

References

[1] P. Maymounov and D. Mazières, "Kademlia: A Peer-to-peer Information System based on the XOR Metric", 1st Int'l Workshop on Peer-to-Peer Systems (IPTPS '02), March 2002, Cambridge, MA, USA.

[2] M. T. Hansen and E. Biagioni, "BTP: a Block Transfer Protocol for Delay Tolerant Wireless Sensor Networks", 5th IEEE Int'l Workshop on Practical Issues in Building Sensor Network Applications (SenseApp 2010), October 2012, Denver, CO.

[3] "Diasporia – All about the Diaspora Social Network", `http://diasporial.com/whats-diaspora`

[4] "friendica", `http://friendica.com/`

[5] "DiSo Project", `http://diso-project.org/`

[6] F. Chung and L. Lu, "The Average Distance in a Random Graph with Given Expected Degrees", Internet Mathematics, vol. 1 (2003).

[7] D. Bloom, "A Birthday Problem", American Mathematical Monthly, vol. 80 (1973).

[8] "Hackers issue fake security certificates for CIA, Google", Electronista, 05 September 2011, http://www.electronista.com/articles/11/09/05/diginotar.hack.tied.to.iranian.government/

[9] Philip Zimmermann, "Why I Wrote PGP", in the Original 1991 PGP User's Guide (updated in 1999).

[10] Broch, Maltz, Johnson, Hu, and Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols, Proceedings (ACM MOBICOM 98), October 1998.

[11] Gupta and Kumar, "The Capacity of Wireless Networks", IEEE Transactions on Information Theory, vol. 46, March 2000.

[12] Clausen and Jacquet, Editors, "Optimized Link State Routing Protocol (OLSR)", Experimental RFC 3626, Oct. 2003.

[13] Perkins, Belding-Royer, and Das, "Ad hoc On-Demand Distance Vector (AODV) Routing" Experimental RFC 3561, July 2003.

[14] Kevin Fall, "A Delay-Tolerant Network Architecture for Challenged Internets", SigCOMM, Aug 2003.

[15] Edoardo Biagioni, "AllNet: using Social Connections to Inform Traffic Prioritization and Resource Allocation", unpublished (2012), available from http://alnt.org/social-distance.pdf

[16] Ratnasamy, Francis, Handley, Karp and Shenker, "A scalable content-addressable network", ACM SigCOMM 2001.

[17] Stoica, Morris, Karger, Kaashoek, and Balakrishnan, "Chord: A scalable peer-to-peer lookup service for Internet applications", ACM SigCOMM 2001.

[18] Rowstron and Druschel, "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems", 18th Int'l Conf. on Distributed Systems Platforms, 2001.

[19] Zhao, Huang, Stribling, Rhea, Joseph, Kubiatowicz, "Tapestry: A Resilient Global-Scale Overlay for Service Deployment", IEEE JSAC, vol. 22, 2004.

[20] "About LifeNet", http://www.thelifenetwork.org/about.html

[21] Int'l Federation of Red Cross and Red Crescent Societies (IFRC), "TERA (Trilogy Emergency Relief Application) and Beneficiary Communication", http://www.ifrc.org /en /what-we-do /beneficiary-communications /tera/

[22] CDAC Network, "The CDAC Network: Improving the Effectiveness of Humanitarian Response", white paper, http://www.cdacnetwork.org

[23] R. Dingledine, N. Mathewson, P. Syverson, "Tor: The Second-Generation Onion Router", Usenix Security 2004.

[24] Ian Clarke, "A Distributed Decentralized Information Storage and Retrieval System", University of Edinburgh, 1999. https://freenetproject.org/papers/ddisrs.pdf

[25] Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", made public May 24 2009. Available from http://bitcoin.org/bitcoin.pdf

[26] Edoardo Biagioni, "A Ubiquitous, Infrastructure-Free Network for Interpersonal Communication", 4th Int'l Conf. on Ubiquitous and Future Networks (ICUFN), July 2012.

[27] http://alnt.org/