Spontnet: Experiences in Configuring and Securing Small Ad Hoc Networks

Laura Marie Feeney, Bengt Ahlgren, Assar Westerlund†, Adam Dunkels‡
Swedish Institute of Computer Science
Box 1263, SE-164 29 Kista, Sweden
Email: lmfeeney@sics.se

Abstract—**In contrast with work focusing on routing problems in mobile ad hoc networks, this work addresses the problem of system configuration in such networks.**

In particular, we are interested in ways to instantiate the configuration infrastructure - naming, addressing, authentication, and key distribution - needed to establish small-to-medium scale ad hoc networks supporting collaborative applications.

We argue that, in such *spontaneous networks* , much of the necessary infrastructure can be derived from the face-to-face human interactions that these networks are intended to facilitate. This approach has the additional advantage of being intuitive for the non-expert user.

In this paper, we describe Spontnet, our prototype implementation of a simple ad hoc network configuration utility based on these ideas. Spontnet allows users to distribute a group session key without previous shared context and to establish shared namespace. Two applications, a simple web server and a shared whiteboard, are provided as examples of collaborative applications that could be useful in a spontaneous networking environment.

I. BACKGROUND

An infrastructureless (also known as ad hoc) wireless network is one in which nodes cooperatively establish a network independently of any base station infrastructure, fixed common computational or storage elements or centralized management. One key advantage of infrastructureless wireless networks is their potential for "anytime, anywhere" operation. Such networks have been proposed for use in a variety of environments, including data fusion in sensor networks and tactical communication in military or disaster-relief operations. It is anticipated that such networks will also play an important role in the development of pervasive computing environments.

In the absence of a base station infrastructure, nodes must cooperate to forward traffic within the wireless cloud in which they operate. Mechanisms for creating a dynamic routing infrastructure for multihop wireless networks have been widely studied[1]. Several routing protocols are under consideration by the Mobile Ad Hoc Networking (MANET) working group in the IETF[2].

By contrast, there exist very few mechanisms for meeting the "service infrastructure" requirements of applications operating in this environment. Without trusted centralized administration, problems such as access control, address allocation, namespace management, authentication and key distribution are very difficult to solve in the general case, because the nodes have no prior shared context.

We therefore turn our attention to the more limited problem of supporting collaborative applications for a small group of people who come together to participate in some activity. The small scale and intentional nature of the network make first-person authentication and distribution of keying material a practical technique for defining network boundaries and configuring basic services.

No pre-planning or pre-configuration of any sort is required — participants create these "spontaneous networks" at will. We believe that the spontaneous networking scenario limits the problem space in a way that makes the service infrastructure more tractable, while still allowing for many useful applications.

Having introduced the non-trivial problem of creating a service infrastructure in ad hoc networks and defined a tractable subset problem, section II expands on the notion of spontaneous network and provides a concrete example of its practical application. The main part of the paper describes our current prototype implementation of Spontnet. Sections III and IV describe security and namespace management in a spontaneous network in more detail. Section V shows a spontaneous network in action. Finally, sections VI and VII describe existing related work and provide some conclusions and directions for further work.

II. SPONTANEOUS NETWORK

The defining requirement in a spontaneous network is that of intentional face-to-face interaction: the activity has a well-defined duration and membership. As a corollary, even participants that are untrusted in other respects can be expected to participate cooperatively in establishing the network infrastructure.

As a concrete example, let us consider a small group of people, representing different organizations, who are meeting together to work on a joint project proposal. These people, naturally, bring their wireless-enabled laptop computers to the meeting and would like to be able share information and jointly edit project documents. How can they do this?

If the meeting is being held on-site at one of the organizations, the visitors' computers could be configured to operate in the host organization's network, Unfortunately, this requires administrative intervention (e.g. enabling wireless access, distributing passwords, configuring the visitors' computers) and is a security issue for both the host and visiting nodes.

Alternatively, the meeting could take place at a location, such as an airport lounge, that provides (possibly for a fee) an IEEE 802.11 "hotspot". In this case, the hotspot infrastructure will take care of name and address configuration and users will be able to access the global Internet and possibly their home networks. However, users will need to configure their systems

†Currently at Permabit, Inc., Cambridge, MA.
‡Also at Mälardalen University, Sweden.

0-7803-7686-2/02/$17.00 © 2002 IEEE

carefully to ensure that their interactions are secure. This may be particularly difficult in the case of a NAT-based hotspot.

In the (not-too-distant) future, the visitors might also be able to use 3G devices and secure MobileIP protocols to access their own organizations' networks via the cellular infrastructure, relying on "home agents" in their home networks to moderate their communication over the global Internet. Such services will certainly not be free, and may prove too costly for activities such as meetings that may last for an hour or more.

In short, these solutions are neither failure resilient, nor resource efficient, nor cost efficient. Why involve remote systems (possibly distributed around the world) to enable collaboration among people who are all sitting in the same room? It seems clear that, even in the future world of ubiquitous wireless access, small isolated networks will be a common occurrence.

To configure such networks, it is necessary to leverage the human interactions associated with collaborative activities to create the needed security infrastructure. Humans are quite good at performing "authentication" and "access control" in face-to-face interaction. In human terms, a handshake may be said to convey an identity and indicate (some) trust. We expand on this notion by using a short-range point-to-point IR "handshake" to initialize a name service and establish a session key for the group.

In addition to providing privacy for group traffic, this session key serves the essential purpose of defining a network boundary for the group. This is important for preventing problematic interactions between nearby networks, which might be caused by broadcast configuration traffic or overlapping address spaces.

III. Spontnet Authentication and Key Distribution

In Spontnet, authentication and key distribution works as follows:

Each user initializes his or her device with his or her name and email address. A user might choose to use a different "identity" for different situations, depending on the organization he or she represents (e.g. a professional or personal meeting). While the format of the name is relatively unconstrained, it is assumed that user selects a valid email address (see discussion below).

One user is selected to initiate the session key. This is a quite informal operation, e.g. "Alice, can you please get the network started?", which is easy to manage in small groups. Once the key has been initialized and distributed to at least one other person, there is no further requirement on the initiator, who may leave at any time.

The key exchange operation is designed to be as explicit as possible, requiring active participation from both the sender and the receiver. A second user obtains the key by pointing the IR port on his or her device at the IR port on the initiator's device and transmitting a [name, request] tuple. A small popup window informs the initiator of the name of the requester. The initiator can then visually "authenticate" the identity of the requester and check the alignment of the IR link. The initiator must then explicitly confirm the transfer of the session key to the second user. The key is then transmitted, in the clear, over the IR link between the two users. Since any user who has the key can redistribute it to any other user, the time to distribute the key through a small group is quite short.

Once the session key has been received via the short range IR link, it is used to secure traffic on the RF (i.e. IEEE 802.11) link. The session key is used as an IPSec key. The IPSec policy is set such this key is required for traffic to and from all addresses in the spontaneous network, as well as for multicast addresses used in the spontaneous network.

Each node periodically transmits its name and address information on the secure multicast channel, so that any node that has the key quickly (within some seconds) learns the identity of all the other participants in the group. If participant information is not refreshed in this way, it eventually times out. The spontaneous networking "daemon", maintains a small window, displaying the current population of the network.

It is important to emphasize that the transmission of keying material in the clear is not secure. Any attacker that can eavesdrop on the key exchange will be able to listen to any traffic in the spontaneous network, as well as injecting arbitrary traffic into the network. However it is physically rather difficult to unobtrusively intrude on an IR link, which tends to be at least somewhat directional and have fairly short range. Because, unlike RF energy, IR energy does not pass through walls, transmitting a session key in a closed room is more secure than transmitting it in an open area. This corresponds nicely with our natural intuitions about security.

The goal is to provide a level of security consistent with the type of information that may reasonably be exchanged during meetings held in an unsecured location. Those who must seriously consider an attack by an adversary with access to equipment such as an IR sensitive camera are extremely unlikely to hold such meetings. Similarly, attacks in which someone physically impersonates another human being are (far) out of scope of this work.

Alternatively, a public key exchange can be used to protect the session key from eavesdropping, though not from a man-in-the-middle attack. In this case, the requester generates a public/private key pair and includes the public key in the request. The initiator uses this key to encrypt the response containing the session key. Note that this is a completely arbitrary key; there is no outside agency by which to verify that a key belongs to a user. In this case, an attacker cannot learn the key by eavesdropping on the IR link, but must instead substitute the requester's public key with its own. This kind of active attack is physically quite difficult over an IR link. Such an attack could also be thwarted by requiring that each pair of users manually compare a hash of the public key used in the exchange; however, this begins to raise ease-of-use issues.

Because all instances of the session key originate from the initiator of the session key, this mechanism could also be used to create a tree containing a chain of signed public-key certificates for each node in the network.

IV. Spontnet Addressing and Naming

Addressing and naming are fairly straightforward in the Spontnet system, primarily because the implementation uses IPv6 [3]. IPv6 provides a 128-bit address space, in which the lower order bits of the address space may be derived from the

Fig. 1. A handshake is a form of authentication and indicates a minimal level of trust.

Fig. 2. Key transfer requires careful alignment of laptops.

MAC address of the device or may be randomly generated. Although the IPv6 specification [4] requires that duplicate detection be performed on addresses generated in this way, this is currently not implemented. The periodic announcements described in Section III above could be used for this purpose.

Node names in Spontnet are derived from the user's email address, which was specified in the initialization information described above. Email addresses have several properties which make them suitable for this purpose: they are globally unique, they are understood even by inexpert users and they are usually reasonably mnemonic, being of the form *name@organization*

In the current implementation of Spontnet, the small number of nodes makes it feasible to obtain the network namespace via periodic broadcast of member information and store it in a local database (/etc/hosts). Multicast DNS lookups are a viable alternative, although this level of scalability is not required in the Spontnet case.

V. Demo Applications

Our prototype implementation of the Spontnet system was recently demonstrated at [5]. It can be seen running in the photograph in Figures 1, 2, and 3.

The demo runs on a collection of five IBM Thinkpad 560 laptops running the FreeBSD[6] operating system and the KAME[7] implementation of IPSec and IPv6. The software also includes a very limited IPv6 implementation of the well-known DSR[8] ad hoc routing protocol. However, the expectation is that in the spontaneous networking scenario, multihop routing will only be used to route between incompatible link layers, e.g. between IEEE 802.11 and Bluetooth devices.

The demo system supports two simple applications. The first is a very primitive shared whiteboard based on *wbd*. The whiteboard application uses a pre-defined address, although it could easily be configured to use zeroconf protocols such as the multicast address allocation and service location protocols.

The second application is a limited web server. The spontaneous networking daemon maintains a list of the current participants in the network, whose home pages can be accessed by clicking on the appropriate name.

Web-based services are the best model for applications that are appropriate for use in a spontaneous network. Although participants in the network are trusted to act cooperatively in the formation of the network — after all, they are voluntarily participating by explicit invitation — they are not trusted in any other way. However, there is no reason to assume that network participants will not try to obtain unauthorized access to other nodes in the network.

Therefore, in addition to filtering out all traffic not secured with the session key, nodes should strictly limit access in all other respects. In this regard, we note that configuring secure web services for a machine operating in a hostile environment is a well-understood, if not completely solved, problem.

Although the current applications are primitive, it seems clear that our service infrastructure makes it possible to develop sophisticated secure collaborative applications for use in a spontaneous network.

Users who have experimented with the system have had generally positive reactions. As can be seen in Figure 2, aligning the laptops to perform the key exchange can be rather clumsy. Work is currently in progress to migrate the key exchange to a smaller device (iPAQ). Once the key transfer is completed, the user can download the key to his or her laptop in a separate operation. This second transfer can be made arbitrarily secure, because the security association between the user's laptop and PDA (or mobile phone) is long lived.

VI. Related Work

Minimal configuration networks are an active topic of research, particularly for the case of the small office and home environments.

For these situations, many problems, such address allocation and duplicate address detection, have been (at least partially) solved, particularly for the case of IPv6 address autoconfiguration. The IETF's Zeroconf Working Group is chartered to address problems in name resolution and service location, drawing on other IETF sponsored protocols where appropriate. To

Fig. 3. Web based applications are a good basis for secure collaboration.

some extent, Zeroconf and IPv6 solutions are directed toward a single link or collection of links sharing a common router.

Because a multihop ad hoc network is subject to arbitrary partitions and merges, duplicate address detection is a more difficult problem in this environment [9]. Because a small spontaneous network more closely resembles the former case — multiple link technologies are the only source of multihop requirements — Spontnet is based on IPv6 address autoconfiguration as discussed in Section IV.

A number of proposed methods for securing various aspects of ad hoc networks are described below. For the most part, these methods assume the existence of a secure external mechanism for configuration and key distribution.

The idea of leveraging first-person authentication and identifying it with a short-range handshake as a means of configuring a small, isolated network appears to have been first proposed in [10] (a precursor to this work) and in a slightly different context, in [11].

Zeroconf

The IETF Zeroconf Working Group [12] is chartered to develop techniques for minimal configuration techniques for interface configuration and multicast address allocation, name-to-address mapping and service location. The working group charter only covers the special cases of a single network segment or a collection of segments connected by a single router. However, it is noted that mechanisms which do not depend on these assumptions are valid in arbitrary topologies.

Key elements of zeroconf include IPv6 address autoconfiguration (and its IPv4 counterpart), multicast DNS (DNSext) extensions, a multicast-based service location protocol (SLPv2), and multicast address allocation (ZMAAP). Some portions of the zeroconf protocols are commercially available in Apple Rendezvous[13].

The zeroconf protocols do not specifically address security infrastructure, although there is a clear requirement that the use of zeroconf protocols not compromise existing levels of security.

Access Control

Access control is a particularly important problem in infrastructureless wireless systems. Access control is needed to define network boundaries for both operational and security purposes.

For a node to be connected to a wired network segment, the node (or user) must have physical access to that segment. A wireless infrastructure network does not provide this implicit access control. A network administrator can obtain a similar result by limiting access to the base station. In IEEE 802.11[1], this can be done by means of a network password (which is distributed to users via some external mechanism) or by configuring the access point with a list of interfaces that are allowed to communicate via the base station.

Because an ad hoc network does not have a base station infrastructure, base stations cannot be used as an access control mechanism. Moreover, the decentralized nature of an ad hoc network limits the applicability of any kind of centralized access control mechanism.

Secure Routing

Because an ad hoc network cooperatively creates its routing infrastructure, it is important to be able to recognize legitimate route information or to detect misbehaving nodes.

One approach is to cryptographically secure the routing information exchanged by nodes. SEAD [14] uses a hash chaining technique that allows (possibly computationally limited) nodes to validate a sequence of ad hoc (DSDV-based) routing table updates with low computational cost. SEAD assumes that initial hash values are distributed via some secure external mechanism.

In addition to injecting invalid routing information, a node can also attack the network by refusing to forward traffic along an assigned route. In [15], snooping is used to detect and route traffic to avoid such misbehaving nodes. In [16], responsible behavior is encouraged by exchange of a virtual currency called "nuglets". Again, it is assumed that necessary cryptographic infrastructure is created via a secure external mechanism.

Distributed Certificate Authorities

In [17], a distributed certificate authority based on *threshold cryptography* is used to guard against the situation in which some certificate authorities may be compromised. The private key of the certificate authority is divided among n servers in such a way that $t+1 < n$ servers are needed to sign a certificate.

This scheme assumes there is an external administrative infrastructure for configuring the certificate authorities and database containing the identities of validated users.

Self-organized Public Key Infrastructure

Reference [16] describes a self-organized public key infrastructure, which is rather like a decentralized variant of PGP (Pretty Good Privacy). If one user accepts another user's public key as valid, he or she signs a public-key certificate for that user's key.

[1] We do not argue that such mechanisms are foolproof as currently deployed.

Each user maintains a repository of public-key certificates, including all those which he or she has signed, as well as a small number of certificates signed by other users. The union of these collections of certificates forms a trust graph over the users of the network.

Because the trust graph behaves like a small-world graph, it is possible (with high probability) to efficiently discover certificate chains to verify any public key. This mechanism is intended primarily for large, long-lived campus or metropolitan area ad hoc networks, such as those described in the Terminodes[18] project.

Resurrecting Ducklings

Spontnet is perhaps closest in spirit to [11], which describes a mechanism for creating a "secure transient association" between a user (or a device, such as a PDA, which is more or less permanently associated with the user) and a simple device, which may be shared by a number of users in sequence. The paper presents an extended example of a wireless enabled thermometer being used in a medical environment.

When a user begins to use the device, it is "imprinted" with the identity of its owner, in the same way that a new duckling is bonded to its mother. Only the current mother of the device can manipulate the device. The imprinting may be explicitly erased when the device is returned to storage or it may time out after some interval, allowing the duckling to resurrect itself in a new context.

In order to ensure the bonding between a user and device is well defined, the authors suggest physical contact as the best mechanism for transferring key information.

VII. Conclusions

This paper describes our experiences building Spontnet, a simple demonstration system that shows how to use face-to-face authentication and a short-range link with easily identifiable endpoints to distribute a session key and namespace information.

Future work includes support for more complex subgroups within the spontaneous network and the development of more sophisticated web based collaborative applications for use in spontaneous networking scenarios.

VIII. Acknowledgements

Franklin Reynolds, of Nokia Research, Boston, has contributed to this work with much helpful discussion.

The authors would like to thank Kersti Hedman of SICS for her excellent photography. We would also like to thank Petra Fagerberg and Anders Andersson, also of SICS, for taking time out to be our models.

References

[1] E. M. Royer and C.-K. Toh, "A review of current routing protocols for ad-hoc mobile wireless networks," *IEEE Personal Communications Magazine*, pp. 46–55, Apr. 1999.

[2] S. Corson and J. Macker, "Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations," Tech. Rep. RFC 2501, Internet Engineering Task Force, 1999.

[3] S. Deering and R. Hinden, "Internet protocol, version 6 (ipv6) specification," Tech. Rep. RFC 2460, Internet Engineering Task Force, 1998.

[4] S. Thomson and T. Narten, "Ipv6 stateless address autoconfiguration," Tech. Rep. RFC 2462, Internet Engineering Task Force, 1998.

[5] L. M. Feeney, B. Ahlgren, A. Westerlund, and A. Dunkels, "Spontaneous networking for secure collaborative applications in an infrastructureless environment," in *Demo Abstracts, Pervasive 2002*, Aug. 2002.

[6] http://www.freebsd.org.

[7] http://www.kame.net.

[8] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing* (Imielinski and Korth, eds.), vol. 353, Kluwer Academic Publishers, 1996.

[9] N. Vaidya, "Weak duplicate address detection in mobile ad hoc networks," in *Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, June 2002.

[10] L. M. Feeney, B. Ahlgren, and A. Westerlund, "Spontaneous networking: an application–oriented approach to ad hoc networking," *IEEE Communications Magazine*, vol. 39, pp. 176–181, June 2001.

[11] F. Stajano and R. Anderson, "The resurrecting duckling: Security issues for ad-hoc wireless networks," in *Proceedings of 3rd AT&T Software Symposium*, pp. 172–194, 1999.

[12] E. Guttman, "Autoconfiguration for ip networking: Enabling local communication," *IEEE Internet Computing*, vol. 5, pp. 81–86, May 2001.

[13] http://www.apple.com/macosx/jaguar/rendezvous.html.

[14] Y.-C. Hu, D. B. Johnson, and A. Perrig, "SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks," in *Proceedings of the 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2002)*, pp. 3–13, June 2002.

[15] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Mobile Computing and Networking*, pp. 255–265, 2000.

[16] J.-P. Hubaux, L. Buttyan, and S. Capkun, "The quest for security in mobile ad hoc networks," in *Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC 2001)*, Oct. 2001.

[17] L. Zhou and Z. J. Haas, "Securing ad hoc networks," *IEEE Network*, vol. 13, no. 6, pp. 24–30, 1999.

[18] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J.-P. Hubaux, and J.-Y. L. Boudec, "Self-organization in mobile ad hoc networks: the approach of terminodes," *IEEE Communications Magazine*, vol. 39, June 2001.