Validating and Securing Spontaneous Associations between Wireless Devices

Tim Kindberg, Kan Zhang
Mobile and Media Systems Laboratory
HP Laboratories Palo Alto

HPL-2002-256
September 12th , 2002*

E-mail: {timothy,kzhang}@hpl.hp.com

secure
association

One of the sought-after characteristics of mobile and ubiquitous computing environments is for devices to become spontaneously associated and interoperate over wireless networks. However, unlike the cable that connects two devices in a wired association, a wireless network does not provide a physical indication of which device is on the other end of the association. Further, the messages sent over a wireless network are readily accessible to other devices on the same network. Hence, a spontaneous wireless association is subject to various spoofing and replay attacks. We introduce protocols to thwart these attacks by physically validating the two devices in a wireless association and, in so doing, exchanging a shared session key between them for subsequent secure communication.

* Internal Accession Date Only
Approved for External Publication

Validating and Securing Spontaneous Associations between Wireless Devices

Tim Kindberg & Kan Zhang
Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304, USA
{timothy, kzhang}@hpl.hp.com

Abstract

One of the sought-after characteristics of mobile and ubiquitous computing environments is for devices to become spontaneously associated and interoperate over wireless networks. However, unlike the cable that connects two devices in a wired association, a wireless network does not provide a physical indication of which device is on the other end of the association. Further, the messages sent over a wireless network are readily accessible to other devices on the same network. Hence, a spontaneous wireless association is subject to various spoofing and replay attacks. We introduce protocols to thwart these attacks by physically validating the two devices in a wireless association and, in so doing, exchanging a shared session key between them for subsequent secure communication.

1 Introduction

We expect a frequently encountered task in mobile and ubiquitous computing will be to make spontaneous associations between devices over wireless networks. An association is a set-up between two parties that enables them to exchange data over a communication channel. The systems we are interested in support *spontaneous* associations between devices, over wireless networks. That is, devices such as personal digital assistants (PDA's) and printers become dynamically associated according to the demands of the circumstances, and interoperate without cables and without prior software configuration.

For example, a user who brings her PDA into a café is able to electronically discover and use a printer ('printer A' in figure 1) in the cafe without having to plug in a cable or change the configuration of her PDA. Similarly, if the user meets another user with a PDA, she may want to exchange a document between the PDA's. She may also want to play an electronic game with several other PDA-owners in the café.

In all three examples, the user wants to associate her PDA with another device according to her judgement about the trustworthiness of the other device (and its owner). She may be surrounded by other devices of dubious trustworthiness.

Thus, spontaneous interoperation is beneficial for the user but it raises the problems of how to validate and secure an association made over an invisible link. Validating an association means verifying the physical entity of the other party in an association. In this sense, validation can be seen as the physical counterpart of cryptographic authentication of identity. This problem arises because the wireless network doesn't provide a way for matching a 'virtual' network identity to a physical entity. Moreover, securing the association means exchanging session keys with the validated endpoint, for the construction of a secure channel.

Suppose there are two printers in the café (printers A and B in figure 1). They are both connected to a wireless network that runs throughout the café. First, how does the visitor know, especially in an unfamiliar environment, which printer is the printer named 'printer A' on the wireless network that she just made an association with, before she sends her document to print? It will be inconvenient, at best, if her document appears at the wrong printer. The printer A may have a physical label 'printer A' attached to it. However, such a printer can be easily spoofed. There is no way to tell if someone has replaced printer A with another device labelled exactly the same. Second, how can she make sure that her data sent to printer A cannot be eavesdropped or tampered with?

Similarly, two PDA-owners wishing to transfer a document between their devices need to be sure that just their devices are involved, and not, for example, a PDA belonging to a malicious owner nearby.

These problems would not arise if the visitor connected the PDA directly to the device of her choice with a short cable she had brought with her. In such a case, (a) she would know with which device she had associated her PDA; (b) she would be assured of a confidential and integrity-preserving connection to the device. Of course, the target device still has to prove worthy of the user's trust by, for example, maintaining the privacy of their data.

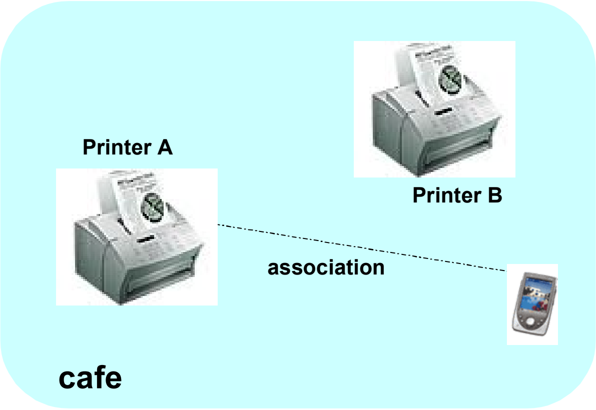

Figure 1. Associating to the right printer

The contribution of this paper is to describe a method for spontaneous device association that provides security guarantees similar to those of an untapped physical cable but does so over a wireless connection. We introduce solutions for (a) physically validating a wireless association such that one or both parties can verify with which device they have associated and (b) securing an association so that data exchanged using the association cannot be eavesdropped or tampered with. The method we use to physically validate an association is based on use of existing techniques to physically locate a device from its network address, which is of independent value.

The rest of the paper is organized as follows. Section 2 discusses related work. Our design for physically validating an association is presented in section 3. Section 4 gives the key exchange protocols that can be combined with the techniques from section 3 to achieve validated secure association. Some additional ideas are discussed in section 5.

2 Related work

Network discovery is where one device (e.g. a PDA) is used to discover others (e.g. printers) using a combination of network communication and access to service registries. Network-discovery systems such as Jini [1] and the Service Location Protocol [8] supply limited information about a service: a network address and (typically) a small amount of text about the service and its host device. There is often too little information for a user to relate the network address (for example, as represented by a link in a Web page to the device's URL) to a particular device in an unfamiliar physical environment. Moreover, information such as device location may be out of date.

Several projects have suggested how to validate an association. The 'resurrecting duckling' design for spontaneous networks [4] provides for 'secure transient associations'. The authors of that paper suggest that a key might be displayed on one of the devices but that option is rejected because it is 'tedious and error prone': the authors advocate 'physical' -- electrical -- contact as a means of key exchange.

The Smart-Its project [7] introduced a method to establish an association between two handheld devices by holding them together and shaking them. Each device captures and broadcasts its own movement pattern. By matching the received movement patterns with its own, one of them can find the other and establish a connection between them. This indeed is a clever idea for the purpose of ruling out unintended connections. However, it isn't suitable for validating wireless associations with large devices, or over a distance. Moreover, the security of this protocol depends heavily on timing. The two devices somehow have to be able to time their broadcasts as close as possible so as to prevent an attacker from replaying received broadcasts. To be fair, the protocol is designed for convenience rather than security.

It is preferable for devices to be securely associated at a more convenient distance. Feeney et al. [6] have suggested using a short-range infrared beacon on the target device. But precautions such as shielding would be required to protect the client from spurious infrared transmissions, whose source cannot be distinguished by distance and can only partially be distinguished by angle.

The SWAP-CA specification [5] for wireless networking in a home environment introduced a protocol, commonly referred to as the two-button protocol, for two wireless devices to find each other. The idea of the two-button protocol works as follows. The users simultaneously trigger the two devices into an 'association' mode, usually by pressing a button on each device. In the association mode, the master device

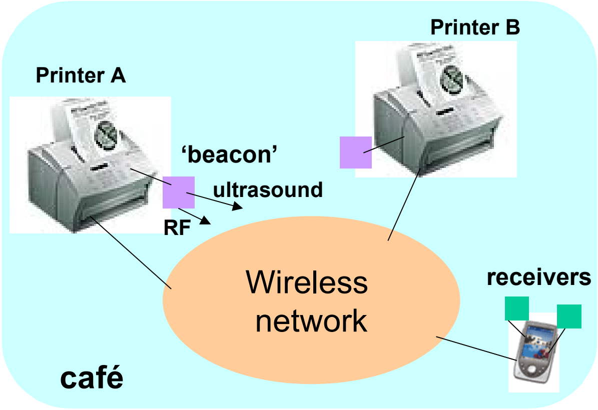

Figure 2. Locating the target device using ultrasound and RF signals

will transmit a special kind of packet and the slave device will listen for such special packets. Once a special packet is received, the slave device is said to have found the device that it is supposed to be associated with, which is the master device that sent the packet. In a peer-to-peer scenario, a device can be both a master and slave device. The two-button protocol has the following drawbacks. First, the validation of an association is based on timing - whoever sends the special packet first gets associated with the slave device. An observing active attacker can hijack the association by simply transmitting such special packets in a high frequency in anticipation of the button being pressed. Second, the two-button protocol requires out-of-band communication between the users of the devices. Third, the association established by the two-button protocol is not secure.

The technique we shall develop is to validate associations based on location information. While we are not aware of prior work taking this approach to validation, many have looked into techniques for locating physical objects. In particular, several (e.g. [9, 10, 11]) have investigated the combination of RF (radio frequency) and ultrasound signals to provide location information based on signal travel times. We use a similar technique to determine the location of the associated device.

3 Physically validating an association

In this section, we apply existing ultrasound-based techniques so that users who have network-discovered a device can physically locate that device and, when it is in line of sight, verify that they are communicating with the intended device. This technique, which is useful in itself, prepares the way for showing how to physically validate a *secure* association in the next section.

A discovery service provides the client with a list of participating devices that match the client's specification. In our example of a café, the network discovery service provides our visitor with a list of devices available in the café including printers A and B. The service may provide her with information about the capabilities and also the locations of the printers. But it may not be obvious to her which is which. She may not be able to see one or both of them.

Our design employs a combination of components to help the user locate her chosen ('target') device and, when she is in line of sight with it, validate an association with it (see figure 2). The client device, which we shall assume is a PDA, is connected to the target device via a wireless network such as IEEE 802.11 or Bluetooth.

In the simplest case, the PDA is connected to two ultrasound receivers (later we shall discuss using more). The receivers are mounted symmetrically on the PDA or even on the user, e.g. their shoulders. Correspondingly, there are 'beacons' connected to and mounted on the target devices in such a way that they cannot be dissociated from their device except at considerable expense. Beacons have ultrasound transmitters constructed to emit an ultrasound message. In addition, beacons have RF transmitters capable of direct RF transmission of negligible latency, like the type of RF transmission used to unlock a car door or open a garage. Correspondingly, the PDA is equipped with an RF receiver.

When the user presses a 'locate' button next to the required service listed on her PDA, the PDA sends a message to the corresponding device -- say 'printer A' in Figure 2. When, and only when, the target device receives a 'locate' message containing its designated identifier, it instructs its beacon to emit (a) an RF message acknowledging the 'locate' message and (b) at the same time, an ultrasound message. Both messages encode that same identifier.

The PDA listens for the RF message and, at each ultrasound receiver, the ultrasound message. When the PDA receives those three events, it records the time of receipt and verifies that the identifiers in the RF and ultrasound messages match. If the identifiers don't match, the user is asked to try again (a spurious or malicious message was received). Otherwise, the PDA calculates the time of flight of sound from the beacon to each ultrasound receiver, from the differences of arrival times of the RF message and the ultrasound message at each receiver. We suppose that the PDA has a clock with microsecond precision, and that it uses a nominal value for the speed of sound to calculate distances from times of flight.

The PDA is now in a position to compute the approximate orientation of the target device from the axis of the two receivers, using knowledge of which receiver heard the message first. At worst, the device can display arrows pointing right-left or front-back, together with approximate distance information. At best, through elementary trigonometry based upon the known distance between the receivers and the difference of the arrival times at the two receivers, it can show the location as lying along a particular line. However, in the latter case, the increased accuracy of pointing may be spurious since the ultrasound might emanate from a point that is not in line of sight (it may be received by reflection or diffraction).

Moreover, the location of the responding device is ambiguous since, with two ultrasound receivers, a given set of arrival times may in principle correspond to any location in a plane bisecting them. It is likely in practice that the target device lies roughly in a horizontal plane with the PDA device. However, the arrows displayed on the PDA should be double-headed to indicate the ambiguity between right-left and front-back locations.

We rely on the user to inspect for other potential sources based on the indicated distance, and to turn the PDA and/or walk and re-press the 'locate' button to disambiguate the source of the signals. As the user moves forwards or backwards in the arrow's direction, the PDA can change its double-headed arrow to a single-headed arrow pointing relative to the user's direction of motion.

The user re-presses the 'locate' button until the target device is unambiguously before them, as registered by an arrow pointing straight towards it from the axis of the ultrasound receivers, and a reading of distance that matches the user's estimation for the object of their attention. She then has a physically validated association with the device.

If the PDA is equipped with more than two ultrasound receivers, then it is possible to provide orientation information in three dimensions and hence provide less ambiguous feedback to the user. Moreover, the developers of the Cricket compass [10] report a directional accuracy of about 5 degrees and distance accuracy of 25 cm. with 5 ultrasonic receivers.

4 Secure association

The technique in the previous section validates a target device as the one the user has chosen, but it does not provide secure communication with that device. We now provide protocols to construct a physically validated secure association. These protocols securely exchange a session key between two devices while physically identifying the other device.

4.1 Secure association with one-way validation

The scenario in this case is of a user wanting to make a validated secure association with a target device in their line of sight. Such an association can be seen as equivalent to using a piece of cable between two devices: the user is assured that communication between the two devices is private and tamper-proof, and that communication takes place between her client device and the target device as opposed to any other device.

The way we achieve secure association is to exchange a session key between the client device and the target device. The session key is known only to the two devices and used for encrypting the communication between them. We use public-key cryptography techniques [2] to exchange a session key. Techniques from section 3 are combined with key exchange protocols to achieve validated key exchange.

As in section 3, we employ a client device with (in the simplest case) two ultrasound receivers and a RF receiver; and the target device has a beacon that emits ultrasound and RF signals. The target device also has a public-key pair (Kp, Ks). The public-key pair can be generated by the target device itself or be given to the target device during configuration. After the user has discovered the network address of the target device and is in line of sight with it, the following protocol can be used to set up a validated secure association.

(a) Client → Target: 'associate'

The user points his client device at the target device and presses an 'associate' button. An 'associate' message is sent to the target device from the client device over the wireless network.

(b) Target → Client: N1, Kp (RF message)

Target → Client: N1 (ultrasound message)

When the target device receives the 'associate' message, it emits a RF message and, at the same time, an ultrasound message.

The RF message contains public-key Kp and a random number N1 generated by the target device. The ultrasound message transmitted by the beacon on the target device need contain only the same random number N1.

(c) The client device receives the RF and ultrasound messages. It checks that the time of arrival of the ultrasound message at the ultrasound receivers is the same to make sure that the target device is what the client device is pointing at. It also checks that the ultrasound message contains the same random number N1 as in the RF message. If not, it indicates to the user that she should try again (a spurious or malicious message was received). Otherwise, it computes an approximate distance using the speed of sound and the time of flight of the ultrasound message, and asks the user to verify that it corresponds to what she perceives.

(d) Client $\rightarrow$ Target: N1, {K}Kp

If the user is satisfied with the physical parameters for the received public key, she presses a 'confirm' button. At that point, the client device sends a wireless network message back to the target device containing N1, and {K}Kp which is the encryption of session key K using Kp. The session key K is a random number generated by the client device and will be used to encrypt future communication between the client device and the target device over the wireless network.

(e) The target device receives the network message and checks if the random number contained in the message is the same as the one it sent to the client device in step (b). If so, the target device decrypts {K}Kp using Ks and obtains K. If not, it ignores the received message.

(f) If the above steps are completed successfully, both devices should have K. However, a malicious eavesdropper may send a spoofing message N1, {K'}Kp to the target device ahead of the client device in step (d). As a result, the target device may end up with a key K' chosen by the attacker. To defend against such an attack, the two devices can confirm if they have the same session key K by exchanging messages encrypted using K. For example, the target device can confirm to the client device by sending a wireless network message containing {N2, Kp, N2}K, where N2 is a new random number chosen by the target device and {N2, Kp, N2}K is the encryption of {N2, Kp, N2} using K. The client device decrypts the message and checks that the decrypted data {N2, Kp, N2} is properly formed. After confirming that they both have the same session key K, they have setup a secure association between them. Furthermore, this secure association has been physically validated by the user in that the user is sure that she is communicating with the device that she pointed her client device at.

Some variants of the above protocol can also be used. One variant is for the client device to send its public-key Kc to the target device in step (a) and let the target device choose a session K, encrypt it using Kc and send the encrypted session key to the client device in step (b). The client device should be able to decrypt the message and obtain K in step (c). Now the secure association is set up and validated. They may skip steps (d) and (e), and jump to step (f) to confirm the shared session key.

Another variant is to use the Diffie-Hellman key exchange algorithm [3] in place of an encryption-based public-key algorithm. Instead of sending its public encryption key Kp in step (b), the target sends its Diffie-Hellman public key g^{KA} (mod n), where KA is the corresponding private key and, n and g are Diffie-Hellman parameters such that n is a large prime and g is primitive mod n. Parameters n and g are sent together with g^{KA} (mod n). After the client device receives the message and validates it in step (c), the client device chooses KB at random and sends g^{KB} (mod n) to the target device in step (d). After the target device accepts the message in step (e), they both can compute the shared session key g^{KAKB} (mod n). Finally, they can proceed to confirm the session key in step (f).

4.2 Secure association with mutual validation

In the preceding protocols, one device is a 'client' and the other a 'server' and they are asymmetrically equipped. An important case, however, is the symmetrical one where both the devices are PDA's. Their owners wish to establish a secure association in order, for example, to exchange a document or play a game with one another in a potentially hostile environment.

In this scenario, the target device is also a client device that wants to validate the secure association being set up. Both devices are equipped with two or more ultrasound receivers, and also ultrasound transmitters and RF transceivers.

In this case, the two devices can use a similar protocol as in the first variant of section 4.1, except that, in this case, the users require validation of the association from both devices. Thus, in step (d) the 'client' device sends N1 in an ultrasound message simultaneously with (N1, {K}Kp) in a RF message (instead of over the wireless network). That allows the other device to validate the association. If the other user does not receive appropriate validation signals, she can abort the association.

5 Discussion

We have described the application of existing location techniques based on ultrasound and radio frequency signals to validate the location of a network-discovered device and, more significantly, to validate a secure association with that device.

The technique provides roughly equivalent validation and security to the use of a cable. Of course, we provide no security against the possibility that the target device has been subverted, despite the user's trust.

Techniques such as these will prove of increasing importance as researchers in mobile and ubiquitous computing devise more applications in which users spontaneously associate their devices with others, often in unfamiliar or untrustworthy environments. We have given examples of spontaneous printing, document exchange and game-playing. Other examples abound, such as securely associating the devices belonging to just one user, e.g. their PDA or mobile phone wirelessly connected to a 'personal server' containing their data and worn on their belt.

In principle, our techniques allow association with any device in line of sight: the user does not have to be next to the target device. It remains for us to implement our protocols to discover whether they really afford the claimed advantage.

In the meantime, we are also looking into other approaches. For example, a target device can display its public key as a barcode, which is read by a camera or scanner on the client device and used as a basis for key exchange. This would provide more definite validation than an infrared signal.

We are also looking into providing extra physical evidence of association, by making devices flash a light or emit a sound at the point when the user presses the 'locate' button. This might be helpful as a supplement to the ultrasound/RF-based technique for identifying a network-discovered device, depending upon the achieved accuracy and also on how well ordinary users react to the directional information.

References

[1] K. Arnold, B. O'Sullivan, R. Sheifler, J. Waldo, and A. Wollrath. The JINI Specification. Addison Wesley, 1999.

[2] Bruce Schneier, Applied Cryptography, John Wiley & Sons, Inc., 1994.

[3] Whitfield Diffie and Martin Hellman, "New Directions in Cryptography", IEEE Transactions on Information Theory, v. IT-22, n. 6, Nov 1976.

[4] Frank Stajano & Ross Anderson. "The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks". In B. Christianson, B. Crispo and M. Roe (Eds.) Security Protocols. 7th International Workshop Proceedings, Lecture Notes in Computer Science, Springer-Verlag, 1999.

[5] Shared Wireless Access Protocol (Cordless Access) Specification (SWAP-CA), Revision 1.0, The HomeRF Technical Committee, 17 December 1998.

[6] Laura Marie Feeney, Bengt Ahlgren, Assar Westerlund. Spontaneous Networking: An Application-oriented Approach to Ad Hoc Networking. IEEE Communications Magazine - volume 39 - issue 6, June, 2001.

[7] Lars Erik Holmquist, Friedemann Mattern, Bernt Schiele, Petteri Alahuhta, Michael Beigl, Hans-W. Gellersen. Smart-Its Friends: A Technique for Users to Easily Establish Connections between Smart Artefacts. Proc. Ubicomp 2001, Springer-Verlag LNCS 2201, pp. 273-291, 2001.

[8] E. Guttman, Perkins, C., Veizades, J., Day, M., "Service Location Protocol, Version 2", RFC 2608, June 1999.

[9] M. Addlesee, R. Curwen, S. Hodges, J. Newman, P. Steggles, A. Ward, and A. Hopper. Implementing a sen-tient computing system. *Compute*r, 34(8):50–56, August 2001.

[10] Nissanka B. Priyantha, Allen Miu, Hari Balakrishnan, Seth Teller, The Cricket Compass for Context-Aware Mobile Applications, Proc. 7th ACM MOBICOM, Rome, Italy, July 2001.

[11] C. Randell and H. Muller. Low Cost Indoor Positioning System. In proceedings Ubicomp 2001, Springer-Verlag, September 2001. pp. 42-48.