Secure pairing of interface constrained devices

Claudio Soriente, Gene Tsudik and Ersin Uzun*

Computer Science Department,
University of California, Irvine, USA
E-mail: csorient@ics.uci.edu
E-mail: gts@ics.uci.edu
E-mail: euzun@ics.uci.edu
*Corresponding author

Abstract: Secure initial pairing of electronic gadgets is a challenging problem because of the usual lack of a common security infrastructure and the threat of so-called Man-in-the-Middle (MiTM) attacks. A number of techniques have been proposed to address the problem, but many are not applicable to devices lacking required interfaces, such as displays or speakers. In this paper, we introduce a new secure device pairing concept that involves using the human body either as the communication medium for – or the source of – the common secret. We implement the concept as a suite of practical pairing protocols for interface constrained devices.

Keywords: secure device pairing; human-assisted authentication; MiTM; man-in-the-middle attacks.

Reference to this paper should be made as follows: Soriente, C., Tsudik, G. and Uzun, E. (2009) 'Secure pairing of interface constrained devices', *Int. J. Security and Networks*, Vol. 4, Nos. 1/2, pp.17–26.

Biographical notes: Claudio Soriente is a PhD candidate in Computer Science at the University of California, Irvine. He received the MS Degree in Computer Science in 2004 from University of Salerno. His research interests include wireless sensor networks and applied cryptography.

Gene Tsudik is a Professor of Computer Science at the University of California, Irvine. He has been conducting research active in internetworking, network security and applied cryptography since 1987. He obtained a PhD in Computer Science at University of Southern California in 1991. Before coming to UC Irvine in 2000, he was a Project Leader at IBM Research, Zurich Laboratory (1991–1996) and USC Information Science Institute (1996–2000). His research interests include authentication, mobile/wireless network security, anonymity, secure group communication, digital signatures, key management, secure ad hoc network routing, database privacy, secure storage and usable security.

Ersin Uzun is a PhD candidate in Networked Systems at University of California, Irvine. He received his MS Degree in Computer Science in 2006 from University of California, Irvine and BSc Degree in Computer Engineering in 2004 from Bilkent University. His research interests revolve around building secure and usable systems and include usable security, authentication, mobile applications, network security, applied cryptography and digital rights management.

1 Introduction

Proliferation of personal gadgets, such as PDAs, cell-phones and media players, has brought new services and new possibilities to ordinary users. There are many common settings where two (or more) devices work together, e.g., a Bluetooth headset and a cellphone, a PDA and a wireless printer, or an access point and a laptop. Before two devices can begin working together, the user(s) must securely pair them. As part of the pairing process, devices discover each other via a common – usually wireless – communication channel. Unfortunately, traditional cryptographic means (such as authenticated key exchange protocols) are unsuitable for securing this initial channel, since unfamiliar devices have no prior secure context and no common point of trust: no on-line Trusted Third Party (TTP), no off-line Certification Authority (CA), no Public Key Infrastructure (PKI) and no common secrets.

The core problem is how to establish a secure communication channel between two previously unassociated devices. Since wireless communication is, by its very nature, human-imperceptible, there is a very real threat of Man-in-the-Middle (MitM) attacks. Such attacks can occur whenever unauthenticated communication is involved. A ready example is the textbook Diffie-Hellman

Key Exchange protocol (Diffie and Hellman, 1976). Since it is not authenticated, an attacker can easily impersonate either party, such that – at the end of the protocol – both parties think that they are talking to each other, whereas, in reality each is talking with (or through) the attacker.

Some initial pairing solutions require the user to put the two devices into scan/discover modes, respectively, and, once the channel is established, to secure it by entering a user-selected password (or PIN) into both devices. A number of security and usability issues arise with this general approach (see Uzun et al., 2007 for an in-depth discussion). Most early solutions failed to provide both usability and security at the same time. They are either secure but not user-friendly or vice versa. To this end, a number of recent proposals (Goodrich et al., 2006; Kindberg and Zhang, 2003a, 2003b; McCune et al., 2005; Saxena et al., 2006; Soriente et al., 2007b) take advantage of certain out-of-band channels (e.g., audio or visual) to provide secure, yet usable, device pairing. Proposed techniques vary greatly in assumptions about device capabilities, user competence and involvement as well as environmental factors.

Several standardisation bodies also recognised the importance of the pairing problem and have begun working on usable and secure procedures. For example, the Wi-Fi Alliance has published specifications for Wi-Fi Protected Setup (Alliance, 2007). Microsoft has released specifications for Windows Connect Now-NET (Microsoft, 2006), which is closely related to Wi-Fi Protected Setup. Bluetooth Special Interest Group has released specifications on Simple Pairing (Group, 2006). The Universal Serial Bus (USB) forum has recently released specifications for Wireless USB Association Models (Specification, 2006) which stipulate procedures for pairing two Wireless USB devices. Unlike research proposals, standards specifications have to consider devices with a wide range of hardware capabilities. Consequently, specifications do not dictate a single pairing method. All of them support the use of at least one type of auxiliary channel. For example, Bluetooth Simple Pairing supports the use of Near-Field Communication (NFC) and Wireless USB Association Models support the use of USB cables. However, trying to accommodate a wide range of hardware capabilities introduces security problems in such standards, (e.g., Suomalainen et al., 2007; Kuo et al., 2007).

Despite significant recent progress in secure device pairing, one main issue remains unresolved: exotic (or non-ubiquitous) device assumptions. All recent research proposals and standards require certain hardware or interfaces that are not commonly available across the entire spectrum of devices. Prior techniques envisage devices equipped with (at least one of): cameras, infrared or laser transceivers, accelerometers, speakers, microphones, NFC transceivers, USB ports, keypads and displays. Clearly such devices exist but they are not ubiquitous enough. Moreover, considering the extra cost as well as space and/or esthetic requirements, it seems unlikely that all electronic devices will have such capabilities in the near future.

This paper attempts to fill the gap left by prior techniques, by providing a secure solution that is usable and practical on a wide variety of devices. We start from the usability angle end and introduce two new approaches:

- transferring a secret from one device to another over the human body
- using the random aspects of a human body as the common secret.

Unfortunately, although these (conceptual, not-yet-realised) techniques require the absolute minimal human involvement, they require certain exotic hardware assumptions. We then focus on practicality and identify the most common interface in electronic devices that allows us to scale down our hardware assumptions. Finally, we combine our ideas in a suite of protocols called Button-Enabled Device Association (BEDA), which simulates the principle ideas in the two new approaches under more practical hardware assumptions. BEDA can accommodate any pair of devices by using a very basic interface: a functional input button (i.e., a single key) that is almost universally available. Our usability studies show that BEDA protocols are user-friendly and efficient even under such very limited hardware assumptions.

The paper is organised as follows: We introduce the two new approaches in Sections 2 and 3. Then, Section 4 describes the protocols and general operation of BEDA. Section 5 discusses implementation details, followed by results of our usability study. Limitations and possible improvements are addressed in Section 7. Previous work is summarised in Sections 8 and 9 ends the paper with some future research directions.

2 The human body as the communication channel

The idea of using the human body as a medium for electronic communication is not new, with the first proposals dating back to early 1990s (Shivers, 1993; Zimmerman et al., 1995; Zimmerman, 1995). However, initial realisation efforts are quite recent, e.g., as shown in Figure 1, NTT has developed a prototype for testing purposes with some limited (conjectured) commercialisation intentions (Shinagawa et al., 2005; Nippon Telegraph and Telephone Corporation, 2005).

Conceptually, using this technology would significantly improve user experience in secure device pairing. Usability is one of the key challenges in device pairing, mainly because users are very diverse and secure completion depends on the user's ability to perform certain tasks in order to establish a secure Out-of-Band (OOB) channel correctly. If we assume that traffic interception over the body 'channel' cannot go unnoticed, just touching both devices would be sufficient for the user to complete secure pairing. When user touches (or holds) both devices as shown in Figure 2, the ephemeral OOB channel can be quickly established to transmit cryptographic information. Usability of this protocol would be superior to any

currently available technique simply because the required task is much easier and much more intuitive.

Figure 1 PC card type RedTacton Transceiver from NTT technologies (see online version for colours)

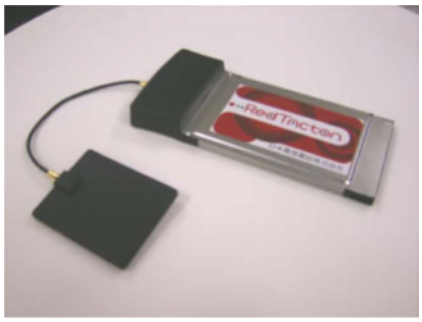


Figure 2 User holding two devices: devices exchange cryptographic information (see online version for colours)

Note that the general idea is nothing more than a speculation over recently available technology. Our main goal is to explore the design space for better usability. Otherwise, the security properties of the human body as the communication channel are not well-known. In fact, it might not be secure – with respect to MiTM attacks – for use in device pairing. This is because current proposals emulate the electric field over the body and transmission signals may continue to propagate over various conductors and dielectrics, e.g., the floor/carpet the user is standing upon. Most importantly, the need for specialised hardware and its high cost make this approach impractical for commodity devices, at least for the time being.

As shown later in the paper, certain BEDA protocols, use the basic idea of transmitting data over the human body. However, we obviate the need for specialised interfaces by requiring more user involvement. Specifically, we take advantage of the human nervous system. Since humans are quite capable of quickly reacting to conditional stimulus with a pre-conditioned action, we use this 'feature' to effectively transmit binary signals from one device to another.

3 Obtaining secrets from the human body

If two devices possess a short common secret, it can be used as a *seed* in device pairing. Password-based Authenticated Key Exchange (PAKE) protocols (e.g., Bellovin and Merritt, 1992) or one-time secret-based key exchange protocols (e.g., Gehrmann et al., 2004) can convert a short shared secret into a set of long-term keys. We postulate that the human body has some constantly changing variables (such as pH level, odour, humidity, temperature and other chemical properties) that are hard to predict for a (non-invasive) adversary. Thus, a combination of such variables – if they can be measured with enough precision – could be used to obtain a short secret.

Assuming such technology exists, we could easily develop a very user-friendly and secure device pairing method whereby the user's involvement is limited to touching the two devices one after the other. Conducting their measurements over the same human body, and within a short time interval, both devices would obtain the same secret which can be later used as the means of authentication for the subsequent key exchange.

Obviously, this general method is a total concoction; we consider it in order to better explore the design space of usability. Testing the reliability and security of this method would require extensive research and, even with the best possible outcome, the cost of measurement-taking interfaces would make it impractical for the foreseeable future.

However, the use of the human body as the source of initial secrets inspires one of the BEDA protocols (described later in the paper) called Button-to-Button (B-to-B). To eliminate the unrealistic hardware requirements, we, once again, impose a little more burden upon the user. Although it is well-known that the human brain is not a good random number generator, our field tests show that the observed waiting times between certain human actions appear random when users are instructed to behave arbitrarily.

4 General operation

As discussed earlier, our main goal is secure pairing of a widest possible range of devices with the emphasis on usability and cost-effectiveness, i.e., minimal features required to support pairing. To this end, BEDA uses the simplest user interface component, a single button, available on almost every device. The auxiliary channel enabled by a single button forms the basis for securing the main communication channel, such as Bluetooth or Wi-Fi. Note that this main communication channel may need to be initially set up in order to use BEDA. We consider this to be a reasonable prerequisite, since BEDA aims to secure the already available (but human-imperceptible and thus subject to MiTM attacks) communication channel.

The main BEDA protocol consist of two phases. In the first phase, a short *21-bit* secret is distributed to both devices over an auxiliary channel. In the second phase,

devices authenticate their respective Diffie-Hellman public keys by proving knowledge of the secret value in a 21-round protocol. The latter is a variant of the MANA III protocol by Gehrmann et al. (2004). In it, the secret is split into 21 pieces and knowledge of a single bit is proven in each round. The ith round between two devices (D1 and D2) is illustrated in Figure 3, where P_i represents the ith bit of the short secret P and PK_1 and PK_2 are the respective public keys to be authenticated.

Figure 3 Round i of authentication using the short secret P

1. D_1
   - generate a large random R_{i1}
   - compute $h_{i1} = h(1, PK_1, PK_2, P_i, R_{i1})$
   - send h_{i1} to D_2
2. D_2
   - generate a large random R_{i2}
   - compute $h_{i2} = h(2, PK_2, PK_1, P_i, R_{i2})$
   - send h_{i2} to D_1
3. D_1
   - send R_{i1} to D_2
4. D_2
   - if $\hat{h}_{i1} = h(1, PK_1, PK_2, P_i, \hat{R}_{i1})$ then ACCEPT else ABORT
5. D_2
   - send R_{i2} to D_1
6. D_1
   - if $\hat{h}_{i2} = h(1, PK_2, PK_1, P_i, \hat{R}_{i2})$ then ACCEPT else ABORT

In the first phase, we considered two approaches for the set up and the distribution of the short initial secret:

- both devices acquire it from the user
- one device chooses it randomly and the user transfers it to the second device.

In the first approach, both devices acquire the same secret through the use of a single functional button. This is achieved by measuring elapsed time between – and during – the button press operations and requiring the user to simultaneously press and release the buttons on both devices, until a long enough secret is acquired. Implementation details and usability analysis of this approach are discussed in the next two sections.

In the second approach, we assume that at least one device has an output interface not easily observable by the adversary. Such output interfaces include: vibration or a small display. The device with this type of output interface signals the user (at certain intervals) to press the button on the other device; idle times between button actions are used for transmitting the actual secret. In such a scheme, press-and-release of a button may or may not be considered as two different actions. In other words, the user may be asked to change the button state from press to release (or vice versa) at every signal, or to press and release the button at every signal. The former results in fewer button actions with longer pressing times, while the latter involves more button presses immediately followed by a release. We implemented several protocol variants (using different output interfaces and button actions) and performed comparative usability tests. The next section describes the implementation and our usability results are discussed in Section 6.

5 Implementation

We implemented and tested four BEDA variants on commodity cell-phones. We used the comparative usability testing framework described in Kostiainen et al. (2007) for fast protocol prototyping and testing. Pictures of devices executing our implementations can be found in Figure A1, in Appendix A.

In the first implementation, both devices acquire the secret directly from user. The user is required to press and release the buttons on the two devices simultaneously and wait for a random (though short) time interval between key-presses. Each device is programmed to start a timer with the first button press and the elapsed time between each button event (either press or release) is then used in determining the short value to be used as the shared secret. Elapsed times between events are kept concatenated until seven events are observed. Each device takes this secret value to the second phase of the protocol. We used 300 ms (0.3 s) as the smallest unit of measurable time. Exact times measured in milliseconds could not be used here due to the less-than-perfect synchrony between the two hands of an average human user. However, less sensitive (longer) time unit selection tolerates such imperfections and delays. Our choice of 300 ms was determined empirically after conducting a small initial study.

We measured elapsed time between each event and reduced it modulo 8 (to obtain a 3-bit value). Over seven button actions we thus collected the total of 21 bits of data. Our choice to construct the secret in 3-bit binary increments was determined after observing (during our pilot study) that users tend not to wait longer than 3–4 s (on average) after they get comfortable with the protocol. Acquiring a secret in 3-bit increments assures the randomness of the resulting secret, even if a user is fast-paced and does not wait more than 2.1 s between successive events. Note that these values can be further adjusted in individual implementations. We use B-To-B in the rest of the paper to refer to this protocol variant.

For the scenario of one device choosing the secret, we considered two modes of output: vibration and display. In the display implementation, one device shows a black square on its screen and the user is instructed to press a button on the other device whenever the square turns white. After the user starts the protocol, the display-equipped device generates a 21-bit random number and waits 3 s before giving the first signal (by colouring the square white for 0.5 s). It gives seven more such signals, such that each signal is separated by idle time determined by ith 3-bit segment of the secret. The receiving device,

on the other hand, measures intervals between button presses in milliseconds and rounds them to the closest full second. This is needed to tolerate up to 500 ms of fluctuation caused by the user reaction/reflex times. We use Display-to-Button (D-To-B) to refer to this variant.

The vibration variant employs the same algorithm as D-To-B but exhibits its signal by vibrating for 500 ms (instead of displaying a square). We refer to it as Short-Vibrations-To-Button (SV-To-B). The last variant takes a slightly different approach and requires fewer button presses. In it, the user is asked to press-and-hold the button on one device **while** the other one vibrates. This final variant is called Long-Vibrations-To-Button (LV-To-B). To transfer the ith segment of the secret, the sending device either vibrates or remains idle (in alternating order) for t seconds, where t is the integer value of ith 3-bit segment of the secret and the sequence starts with vibration. The receiving device considers the press and release of the button as different events and computes each 3-bit segment by rounding the measured time between those events, as described earlier.

6 Usability analysis

Having prototypes all four aforementioned protocol flavours: B-To-B, D-To-B, SV-To-B and LV-To-B, we investigated their respective usability factors by performing a number of experiments discussed in this section.

A total of 20 subjects were recruited. Subjects were chosen on a first-come first-serve basis from the respondents to recruiting posters. Subjects were mainly university students which resulted in a fairly young, well-educated and technology-savvy participant group. The demographics and related background information of the participants are summarised in Table 1.

Table 1 Participant profile

Gender	Male	75%
	Female	25%
Age	18–24	15%
	25-29	60%
	30–34	15%
	35+	10%
Education	Bachelor	50%
	Masters	25%
	PhD	25%
Any difficulty	YES (despite any aid)	10%
with visual abilities	No	90%
Any difficulty	Yes	5%
with reflex abilities	No	95%

Test procedure: Our usability study was conducted in a variety of campus venues (depending mainly on the subjects' preferences), including, but not limited to: cafés, student dorms/apartments, classrooms, office spaces and outdoor terraces. After giving a brief overview of our study and its goals, participants were asked to fill out the background questionnaire to collect demographic information. In this questionnaire, users were asked whether they had any visual impairment or any condition that might interfere with their sensing of vibration or reflexes. Next, users were given a brief introduction to the cell-phones used in the tests and the nature of BEDA protocols.

Each user was then given the two devices and asked to follow on-screen instructions shown before each task in order to complete it. Every user was asked to pair the devices four times in total, using each implementation described in the previous sections. To reduce the learning effect on test results, the four tasks were presented to the user in random order. User interactions throughout the tests were logged automatically by the testing framework. After completing the tasks, each user filled out a post-test questionnaire form and was given 5 min of free discussion time followed by a short interview.

Results: We collected data in two ways:

- by timing and logging user interaction
- via questionnaires and structured interviewing.

Completion time for each protocol was automatically logged by the software. According to this data, using a button on both devices was faster than all other variants, on average, although it needed more trials. Whereas, users successfully paired devices with short signalling vibrations with the least number of trials. However, average completion time hovered around roughly a minute in all methods, as shown in Table 2.

Table 2 Summary of the related logged data

Method	*Average completion time* (s)	*Average number of retrials for success*
B-To-B	53.2 (sd* = 32.5)	2.45 (sd = 1.43)
D-To-B	72.8 (sd = 39.4)	1.45 (sd = 0.89)
SV-To-B	60.1 (sd = 18.3)	1.35 (sd = 0.49)
LV-To-B	56.6 (sd = 19.4)	1.20 (sd = 1.41)

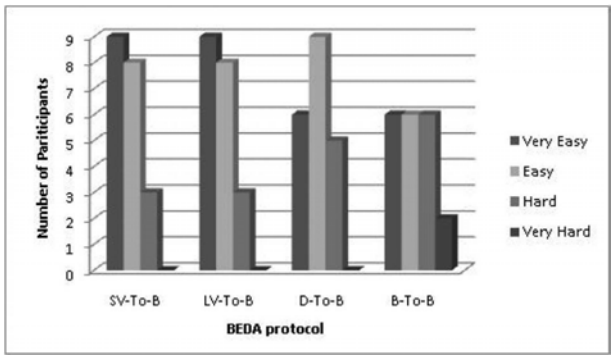
*sd = Estimated standard deviation.

In B-To-B, the secret is derived from user actions and there is hence an obvious risk of it being insufficiently random. Our choice of 300 ms as the measuring unit was made to reach an acceptable balance between security (randomness) and usability (short completion time). Logs generated by our testing software clearly indicate that the derived secrets were indeed random – each octal digit of every derived secret was uniformly distributed in the $[0, 7]$ interval (independent from its position and other numbers in the secret) over 49 protocol runs, including re-trials. Although B-to-B was not the top choice of most users, it does not seem to suffer (in terms of security) from the human serving as the source of randomness.

In the post-test questionnaire, we solicited user opinions about the tested methods. Participants rated each method for its ease-of-use and pointed out whatever

usability problems they experienced. They were also asked to compare each method to their previous experience (if any) with Bluetooth, Wi-Fi or infrared-based pairing. Users found BEDA variants using vibration to be easiest and commented that they required the least concentration. On the other hand, they found B-To-B to be fairly hard. Such results were not surprising considering the relatively delay-intolerant implementation of B-To-B and the more attention-demanding nature of D-To-B, as compared to vibration variants. The ease-of-use ratings given by participants are summarised in Figure 4.

Figure 4 Participant opinion (see online version for colours)



The post-test questionnaire also asked users to order the methods they are already familiar with (including Wi-Fi, Bluetooth and Infrared-based) and BEDA variants from easiest to hardest. Among 13 participants who were familiar with Wi-Fi pairing, 77% considered BEDA to be easier. Whereas, among 14 familiar with Bluetooth pairing, only 36% considered BEDA easier. During short post-test interviews, users explained the reason for Bluetooth being easier than BEDA: the former involving just typing in a few (usually four) digits. However, when the number of required digits gets only a little higher (as in WEP or WPA keys in Wi-Fi secure pairing), they find BEDA easier. The interviews also demonstrated that almost all users liked BEDA protocols and enjoyed using them. More interestingly, majority of users (even the ones that rated some BEDA protocols as being hard and found current methods easier) told us they would like to use BEDA instead of current techniques because they are simple and fun to use. (In fact, they emphasised the subtle difference between **simple** and **easy** and classified BEDA as simple).

7 Discussion and limitations

All BEDA protocol variants require devices with minimal interface capabilities: a single button on one device and a button, vibration capability or an LED/display on the other. In its simplest flavour, BEDA requires both devices to have a single button. Note that some forms of output might still be required for the user to acknowledge the outcome of the pairing process. An LED blinking with a certain pattern, or a simple display might tell the user that the protocol execution was successful and that both devices share the same secret key. Implementation and user-friendliness might vary depending on the device user interface capabilities. However, the conclusion is that BEDA provides pairing techniques for devices with the simplest form of user interface – a single button.

Of the four variants we studied, there is a clear distinction between B-To-B and the other three. The latter use "*human response to a stimulus*" as a conduit for transferring a random secret value chosen by one device to the other device. Whereas, in B-To-B, devices derive the secret value from user's actions.

At first, B-To-B might not look different from widely adopted secure pairing techniques that require the user to choose a random key and enter it into both devices. However, results from Uzun et al. (2007) clearly show that the key obtained in such protocols is far from being random. B-To-B, on the other hand, uses the human (re-)actions and their timings as the source of randomness. We believe (and experiments confirm it) that this data is more random and thus results in better overall security. With 95% statistical confidence, we could not find any evidence to reject the randomness hypothesis over the 343 3-bit segments forming the 49 keys collected on each device as part of our trials. Moreover, existing protocols require a full keypad on both devices, whereas, B-To-B requires only one button.

Average completion time for BEDA variants ranged between 53.2 s and 72.8 s. Although average completion times are expected to improve slightly as users get more experienced, BEDA protocols would still take longer than some other pairing techniques.

In all protocols, users' reflex time in reaction to different stimuli is very important. Our usability tests show that participants can easily accomplish the pairing. Although our participant group was fairly young and generalisation to other age brackets is premature, our subjects included two who had experienced visual difficulties (one with cataracts and another – with 60% loss of vision on one eye) as well as one who was taking prescription medication (Xanax). Although either of these factors can influence reflexes and coordination all three of these subjects performed as well as everyone else.

Our D-To-B implementation uses a square turning from black to white. However, we believe that the protocol is equally applicable to simpler devices only equipped with an LED or a primitive one-line display. Switching on and off an LED (or showing a one-line word 'PRESS' and 'RELEASE') would have a similar effect and offer similar usability features.

All BEDA protocols take advantage of the human user either as a conduit for transferring the secret or as a generator of the secret. This is resistant to MiTM attacks only if the transferred or generated secret cannot be observed. Assuming that participating devices are not compromised, the only way to mount a successful MiTM attack against BEDA is by being close enough to observe either user's or devices' actions. Since devices must be held in the user's hand and be physically close to the user, we claim that an MiTM attack can not remain

unnoticed if the attacker gets close enough to the user. Of course, the attacker can always try to observe the user and devices though a hidden camera or binoculars. Even in such cases, the user can take some obvious steps to conceal own actions and/or device output.[1] Recall that the short initial secret is only used for ephemeral authentication of respective Diffie-Hellmann public keys (exchanged via a human-imperceptible medium). To be successful, the attacker must discover the short secret before the devices move into the second phase of the protocol, where they prove knowledge of the secret. The attacker thus has very little time. Also, the attacker has only one chance of guessing the secret, since failing to prove knowledge of any bit results in the devices aborting the protocol immediately. Finally, once a protocol terminates, obtaining the short secret key is useless since the security of the subsequent session is based on the Diffie-Hellman key of adequate (much greater) size.

Note that security against a passive adversary in BEDA is roughly comparable to that of entering a password using a keyboard. While one enters a password, an adversary might attempt to 'shoulder-surf' but such actions are hard not to notice. If better security is required, all BEDA protocols (except B-to-B) can be modified to resist shoulder-surfing attacks. Instead of a secret-based protocol, one can modify BEDA to use one of the SAS-based protocols (Laur and Nyberg, 2006; Pasini and Vaudenay, 2006) to authenticate the key exchange. After exchanging the cryptographic material, a short authenticated secret may be communicated by one device and transferred over human reflexes to the other. At this point, the receiving device can detect any attack and end the protocol if only unidirectional authentication is needed. However, if mutual authentication is desired, an extra step is needed whereby the receiving device communicates its 1-bit accept/reject decision and the user transfers this bit to the other device. However, the resulting protocol would require both a button and some form of output interface on each device (to support mutual authentication).

8 Related work

There is a fairly large body of relevant prior work on secure device pairing.

The earliest work by Stajano and Anderson (1999) made a seminal contribution by bringing the problem into the spotlight. The proposed techniques, however, required the use of standardised physical interfaces and cables. Follow-on methods by Balfanz et al. (2002) and Feeney et al. (2002) made progress by using infrared communication as the human-verifiable side-channel. Though timely in its day, this approach is no longer viable since:

- few modern devices are equipped with IrDA interfaces (they are too slow, short-distance, require line-of-sight and manual start-up)
- the infrared channel itself is not fully immune to MiTM attacks.

Another approach involves graphical visualisation of the hash of the exchanged cryptographic material. The user then needs to compare the output on both devices. In order to make the comparison easier, researchers devised visual metaphors to represent the hash. Levien and Golberg proposed a 'snowflake' mechanism (Goldberg, 1996; Levien, 1996; Perrig and Song, 1999) used 'Random Art', while Dohrmann and Ellison devised a colourful 'flag' representation (Ellison and Dohrmann, 2003). Although these schemes avoid the cumbersome and error-prone process of comparing two hashes byte-by-byte, they require high-resolution displays, making the approaches suitable for only certain types of devices, such as laptops, PDAs and high-end phones.

The Seeing-is-Believing (SiB) technique by McCune et al. (2005) uses the visual channel to perform secure device pairing. The visual channel is established between the visual transmitter (bar-code displayed on a screen or a sticker) of one device and the visual receiver (camera) of the other device and devices take turn of taking pictures when mutual authentication is needed. The protocol does not rely on the human visual abilities (except that the human needs to focus the camera to take a picture) since the devices themselves compare the bar-codes. SiB is applicable to scenarios where at least one device has a camera. Saxena et al. (2006) developed an SiB extension which achieves secure pairing if one device is equipped with a light detector or a camera, while the other has at least a single LED. The LED device uses its 'blinking' capability to transmit authentication data, while the other device records the blinking pattern, extracts the data and compares it with its own computed value. This protocol requires less in terms of device features, but not all devices have a light detector or a camera. Moreover, the comparative usability study in Uzun et al. (2007) indicates that users are generally not adept in following the prescribed order of interaction if it involves more than one device.

Another pairing approach uses a different human-perceptible channel – audio – in the Loud-and-Clear system (Goodrich et al., 2006). As usual, the proposed protocols involve two devices exchanging their keys and computing the hash of the exchanged cryptographic material. The hash is later translated in a syntactically correct English-like 'Madlib' (gibberish) sentence that can be either played or displayed depending on the available hardware and the user compares the sequences to verify the key exchange in a user friendly way. The authors consider many other scenarios and variations of the protocol, but each device is required to have a speaker or a display even at the simplest of them. Recently, Soriente et al. (2007b) took the approach of using audio one step further and realised the secure device pairing over the audio channel where no other common interface, such as Bluetooth or 802.11, is needed. Although using the audio channel for key transmission increases usability, by taking away the burden of establishing another channel, it is only applicable when both devices have a microphone and a speaker.

Other proposals suggested the use of technologies that more expensive and relatively exotic. Kinberg et al.

suggested an approach requiring RF and ultrasound receiver/transmitters on both devices in Kindberg and Zhang (2003b) and laser technology (each device must be equipped with a laser transceiver) in a more recent proposal (Kindberg and Zhang, 2003a). Holmquist et al. (2001), proposed the use of a common movement pattern as the security initiator when the two devices are shaken together. A similar approach was proposed by Mayrhofer and Gellersen (2007). This requires both devices to be equipped with two-axis accelerometers; it is also unsuitable for physically large/bulky devices.

Recently, some industrial research and standardisation bodies have also published specifications for secure device pairing (Alliance, 2007; Group, 2006; Specification, 2006). These emerging specifications take the typical approach of doing Diffie-Hellmann key agreement over the insecure channel and then authenticating it using an auxiliary channel. Although the implementation is not specified, each specification supports different hardware configurations at the first look. Bluetooth Simple Pairing (Group, 2006) requires a display on one device and a display (or a keypad) on the other. Wi-Fi Protected setup (Alliance, 2007) requires a display on one side and a keypad on the other and Wireless USB (Specification, 2006) supports devices with a display. Each of these specifications also support at least one OOB channel which is usually even more demanding in terms of required hardware, e.g.; USB ports, NFC transceivers or cables.

In summary, aforementioned techniques and specifications require particular hardware and/or interfaces that are simply not available on many devices. There are common pairing scenarios, such as a wireless printer and a laptop, an access point and a PDA, or a wireless headset and a desktop, which are not supported by any of the previously mentioned protocols. Even in some pairing scenarios where the previous schemes seem to apply, one would still need a combination of several such schemes to accommodate a considerable fraction of possible pairing scenarios. Moreover, the usability of such a combination would be very questionable, especially, since no comprehensive usability study has been performed for many of these complex schemes. Moreover, even the very basic pairing methods have not fared well when employed by ordinary (technically non-savvy) users (Uzun et al., 2007).

9 Future work

Our usability experiments show that the relatively short 300 ms interval used with the synchronous button press variant does not provide enough error tolerance for all users and sometimes requires several re-tries. We are in the process of performing further tests to help enhance the usability of this variant. There is an obvious trade-off between increased error tolerance and the number of button presses (which influences completion time) in acquiring 21 bits of random data. We anticipate that further testing and experimentation will aid in determining the optimal parameters. We also plan to conduct more usability experiments in participants' own environments with more comprehensive task scenarios, such as setting up a complete wireless home network with several types of devices, i.e., not just cell-phones. This will offer better insight into the usability of BEDA and a more comprehensive comparison with the current techniques.

Acknowledgement

A preliminary version of this work appeared in Soriente et al. (2007a).

References

Alliance, W. (2007) *Wi-fi Protected Setup Specification*, WiFi Alliance Document, January.

Balfanz, D., Smetters, D., Stewart, P. and Wong, H. (2002) 'Talking to strangers: authentication in ad-hoc wireless networks', *Symposium on Network and Distributed Systems Security (NDSS '02)*, San Diego, California, USA.

Bellovin, S. and Merritt, M. (1992) 'Encrypted key exchange: password-based protocols secure against dictionary attacks', *Research in Security and Privacy, 1992, Proceedings 1992 IEEE Computer Society Symposium on*, pp.72–84.

Diffie, W. and Hellman, M.E. (1976) 'New directions in cryptography', *IEEE Transactions on Information Theory*, Vol. IT-22, No. 6, pp.644–654.

Ellison, C. and Dohrmann, S. (2003) 'Public-key support for group collaboration', *ACM Trans. Inf. Syst. Secur.*, Vol. 6, No. 4, November, pp.547–565.

Feeney, L., Ahlgren, B. and Westerlund, A. (2002) 'Demonstration abstract: spontaneous networking for secure collaborative applications in an infrastructureless environment', *International Conference on Pervasive Computing (pervasive 2002)*, Zurich, Switzerland.

Gehrmann, C., Mitchell, C. and Nyberg, K. (2004) 'Manual authentication for wireless devices', *RSA CryptoBytes*, Spring, Vol. 7, No. 1, pp.29–37.

Goldberg, I. (1996) *Visual Key Fingerprint Code*, Available at http://www.cs.berkeley.edu/iang/visprint.c

Goodrich, M., Sirivianos, M., Solis, J., Tsudik, G. and Uzun, E. (2006) 'Loud and clear: Human-verifiable authentication based on audio', *ICDCS '06: Proceedings of the 26th IEEE International Conference on Distributed Computing Systems*, Lisboa, Portugal, pp.10–17.

Group, B.S.I. (2006) *Simple Pairing Whitepaper*, http://www.bluetooth.com/Bluetooth/Apply/Technology/Research/Simple_Pairing.htm

Holmquist, L., Mattern, F., Schiele, B., Alahuhta, P., Beigl, M. and Gellersen, H. (2001) 'Smart-its friends: a technique for users to easily establish connections between smart artefacts', *UbiComp '01: Proceedings of the 3rd International Conference on Ubiquitous Computing*, Springer-Verlag, London, UK, pp.116–122.

Kindberg, T. and Zhang, K. (2003a) 'Secure spontaneous device association', in Dey, A., Schmidt, A. and McCarthy, J. (Eds.): *Ubicomp*, Volume 2864 of Lecture Notes in Computer Science, Springer, pp.124–131.

Kindberg, T. and Zhang, K. (2003b) 'Validating and securing spontaneous associations between wireless devices', in Boyd, C. and Mao, W. (Eds.): *ISC*, Volume 2851 of Lecture Notes in Computer Science, Springer, pp.44–53.

Kostiainen, K., Uzun, E., Asokan, N. and Ginzboorg, P. (2007) *Framework for Comparative Usability of Distributed Applications*, Technical Report NRC-TR-2007-005, Nokia Research Center.

Kuo, C., Walker, J. and Perrig, A. (2007) 'Low-cost manufacturing, usability, and security: an analysis of bluetooth simple pairing and wi-fi protected setup', *Usable Security Workshop (USEC'07)*, Scarborough, Trinidad and Tobago.

Laur, S. and Nyberg, K. (2006) 'Efficient mutual data authentication using manually authenticated strings', *The 5th International Conference on Cryptology and Network Security, CANS 2006*, Volume 4301 of Lecture Notes in Computer Science, Springer, Suzhou, 8–10 December, pp.90–107, A shortened version of ePrint Report 2005/424.

Levien, R. (1996) *PGP snowflake*, Source code available at: http://packages.debian.org/testing/graphics/snowflake.html

Mayrhofer, R. and Gellersen, H. (2007) 'Shake well before use: authentication based on accelerometer data', *Proc. Pervasive 2007: 5th International Conference on Pervasive Computing*, Toronto, Canada, pp.144–161.

McCune, J., Perrig, A. and Reiter, M. (2005) 'Seeing-is-believing: using camera phones for human-verifiable authentication', *2005 IEEE Symposium on Security and Privacy*, Oakland, CA, USA, pp.110–124.

Microsoft (2006) Windows Connect Now-UFD and Windows Vista Specification, Version 1.0. http://www.microsoft.com/whdc/Rally/WCN-UFDVistaspec.mspx

Nippon Telegraph and Telephone Corporation (2005) *Redtacton Website*, Available at http://www.redtacton.com

Pasini, S. and Vaudenay, S. (2006) 'SAS-based authenticated key agreement', in Yung, M. (Ed.) *Public Key Cryptography – PKC'06, 9th International Workshop on Theory and Practice in Public Key Cryptography*, Volume 3958 of Lecture Notes in Computer Science, Springer-Verlag, New York, USA, pp.395–409.

Perrig, A. and Song, D. (1999) 'Hash visualization: a new technique to improve realworld security', *Proceedings of the 1999 International Workshop on Cryptographic Techniques and E-Commerce (CrypTEC '99)*, pp.131–138.

Saxena, N., Ekberg, J., Kostiainen, K. and Asokan, N. (2006) 'Secure device pairing based on a visual channel', *2006 IEEE Symposium on Security and Privacy*, Oakland, California, USA, pp.306–313.

Shinagawa, M., Ochiai, K., Sakamoto, H. and Asahi, T. (2005) 'Human area networking technology: RedTacton', *NTT Tech. Rev.*, Vol. 3, No. 5, pp.41–46.

Shivers, O. (1993) *BodyTalk and the BodyNet: A Personal Information Infrastructure*, Personal Information Architecture Note 1.

Soriente, C., Tsudik, G. and Uzun, E. (2007a) 'BEDA: button-enabled device pairing', *International Workshop on Security for Spontaneous Interaction, UbiComp 2007 Workshop Proceedings*, Innsbruck, Austria.

Soriente, C., Tsudik, G. and Uzun, E. (2007b) *HAPADEP: Human Assisted Pure Audio Device Pairing*, Cryptology ePrint Archive, Report 2007/093.

Specification, W.U. (2006) *Association Models Supplement*, Revision 1.0, USB Implementers Forum, http://www.usb.org/developers/wusb/

Stajano, F. and Anderson, R. (1999) 'The resurrecting duckling: security issues for ad-hoc wireless networks', *Security Protocols, 7th International Workshop*, Cambridge, UK.

Suomalainen, J., Valkonen, J., Asokan, N., Stajano, F., Meadows, C., Capkun, S. and Moore, T. (2007) 'Security associations in personal networks: a comparative analysis', *Security and Privacy in Ad-hoc and Sensor Networks 4th European Workshop, ESAS 2007*, Cambridge, UK, pp.43–57.

Uzun, E., Karvonen, K. and Asokan, N. (2007) Usability analysis of secure pairing methods, Technical Report NRC-TR-2007-002, Nokia Research Center.

Zimmerman, T. (1995) *Personal Area Networks (PAN): Near-Field Intra-Body Communication*, PhD thesis, Massachusetts Institute of Technology.

Zimmerman, T., Smith, J., Paradiso, J., Allport, D. and Gershenfeld, N. (1995) 'Applying electric field sensing to human-computer interfaces', *Proceedings of the SIGCHI Conference on Human factors in Computing Systems*, Denver, Colorado, USA, pp.280–287.

Note

[1] For example, the user may press devices' buttons in his pockets.

Appendix A: Pictures from our implementations

Figure A1 From top to bottom: B-To-B, D-To-B and SV-To-B (see online version for colours)