Low-cost Manufacturing, Usability, and Security: An Analysis of Bluetooth Simple Pairing and Wi-Fi Protected Setup

Cynthia Kuo
Carnegie Mellon University
cykuo@cmu.edu

Jesse Walker
Intel Corporation
jesse.walker@intel.com

Adrian Perrig
Carnegie Mellon University
perrig@cmu.edu

Abstract. Bluetooth Simple Pairing and Wi-Fi Protected Setup specify mechanisms for exchanging authentication credentials in wireless networks. Both Simple Pairing and Protected Setup support multiple setup mechanisms, which increases security risks and hurts the user experience. To improve the security and usability of these specifications, we suggest defining a common baseline for hardware features and a consistent, interoperable user experience across devices.

1 Introduction

Bluetooth- and Wi-Fi-enabled devices are increasingly common. Already, manufacturers ship around 10 million Bluetooth units and 4 million Wi-Fi units *each week* [1, 2]. Inevitably, consumers will perform security-sensitive transactions – including financial transactions – over Bluetooth or Wi-Fi. Thus, institutions should demand a basic level of assurance: that these technologies do not expose their systems or their customers' accounts to additional risks. This implies that (1) the security mechanisms in Bluetooth and Wi-Fi should be at least as strong as the rest of the system; and (2) the mechanisms should be easy to use so that consumers can configure and use them correctly.

We evaluate the security and usability of setup in the Bluetooth SIG's Simple Pairing specification (August 2006) [3] and the Wi-Fi Alliance's Protected Setup specification (released December 2006) [4]. These specifications were developed with two goals in mind: first, to make the technologies easy for non-expert users; and second, to address vulnerabilities in earlier versions of the technology. Simple Pairing and Protected Setup are not yet available in consumer products at the time of this writing; we present analysis based on the specifications.

Our description and analysis focus on the introduction of one device to another. In Simple Pairing, introduction enables two devices to communicate with one another via Bluetooth. In Protected Setup, it occurs when a device enrolls in an existing Wi-Fi network; we presume the initial setup of an access point has already taken place. Both Simple Pairing and Protected Setup specify multiple methods for introduction. This creates a number of security and usability issues, which we examine in detail.

InterDigital Exhibit 2009, p. 1
Microsoft v. InterDigital
IPR2024-00537

2 Properties of Secure and Usable Setup

In this section, we define properties required for the secure and usable setup of two wireless devices. From a security perspective, setup establishes a secure channel that provides secrecy and authenticity – even in the presence of an active adversary. From a usability perspective, the entire user experience should be intuitive, consistent, and robust. The following subsections will address each set of requirements.

2.1 Secure Setup Requirements

We evaluate the security of Bluetooth and Wi-Fi setup against three factors: 1) conformance to a standard model for authentication; 2) simplicity; and 3) level of security provided. We explain each factor below.

1) Conforms to the standard model for establishing authentication credentials. Wireless communication is inherently vulnerable to message injection and eavesdropping attacks; we cannot rely on the wireless channel alone for establishing credentials. Thus, we rely on an additional out-of-band channel.

The *standard model* for establishing authentication credentials consists of the two devices being introduced, the wireless communication channel (called the in-band channel), the additional out-of-band channel, and an active adversary that controls the in-band channel. In this paper, the in-band channel is Bluetooth or Wi-Fi. We adopt a Dolev-Yao active attacker, who can eavesdrop, insert, modify, delay, and reorder messages sent in the in-band channel. In the standard model, it is assumed that the active adversary cannot control the out-of-band channel.

Because devices are rarely asked to establish authentication credentials, some might argue that our threat model is too strong. Today, the chances that an attacker is present during setup is small. However, this may change: networking technologies, such as Bluetooth and Wi-Fi, are quickly becoming ubiquitous and inevitably will be used for sensitive (e.g., financial) transactions.

2) Preserves simplicity to reduce vulnerabilities. Experts can better find and correct vulnerabilities in simpler security designs. When a design is too complex, understanding whether it excludes all important vulnerabilities is infeasible; thus, the design must be assumed insecure.

3) Provides a high level of security. According to a NIST recommendation for key management, today's cryptographic mechanisms should require an attacker to perform at least 2^{80} operations through 2010. At least 2^{112} operations should be required to provide secure operation through 2030 [5].[1] If we require 2^{80} operations and we assume that an attacker can perform 2^{50} operations, the attacker has no more than a 2^{-30} probability of success. For our analysis, expecting one guess in a billion to be successful (on average) is an acceptably low probability of attack success.

[1] The NIST recommendations are intended for unclassified government data. However, the ANSI X9 standards, which govern the use of cryptography by the financial industry, historically align with NIST guidelines.

2.2 Usable Setup Requirements

A usable setup experience refers to all of the end user-facing details preceding, during, and following credential exchange. For example, a usable setup experience helps an end user: initiate credential exchange; identify precisely which devices are communicating (to the exclusion of other devices in range); understand whether the wireless connection is functional and secure; and recover from errors. We highlight three critical factors in usable network setup below:

1) Maintains a consistent user experience across devices. The setup process should be similar for any two devices. For instance, pairing a Bluetooth headset with a cell phone should feel congruous to enrolling a laptop in a Wi-Fi network. A consistent experience provides two main benefits. First, end users can learn how to perform the setup process and apply this knowledge to subsequent setup attempts. Second, vendors can better support their products. Nearly all network-enabled devices need to interoperate with devices from other manufacturers. By implementing the setup process in a consistent manner – for example, using the same user interaction flow – vendors will be able to anticipate how other devices behave. This facilitates producing more accurate documentation and providing better technical support.

2) Provides confirmation of which parties are communicating. End users need to be confident that the devices which are configured to communicate are the intended devices. Schemes such as Talking to Strangers [6] or Seeing-is-Believing [7] achieve this property through *demonstrative identification*, i.e., identifying which devices are communicating based on physical context. Also, devices should confirm the in-band connection is functional.

3) Incorporates robust error handling. Failure is a common outcome when adding new devices to a wireless network – even for experts. End users need comprehensible error messages when errors occur. This helps users troubleshoot the errors themselves and helps technical support staff with troubleshooting.

3 Bluetooth Simple Pairing

Bluetooth is a Personal Area Networking standard based on short range radios [8]. Devices such as phones, printers, modems, game consoles, and headsets use Bluetooth to communicate among themselves. Bluetooth is useful when two or more devices are in close proximity and require only modest bandwidth.

A Bluetooth device plays the role of either "master" or "slave." A master can communicate with up to seven slave devices, and a Bluetooth network consisting of one master and its slaves is called a piconet. The master controls the timing of all Bluetooth communications on a piconet.

The process of adding a new slave device to a Bluetooth piconet is called pairing. Bluetooth Simple Pairing [3] is a set of security enhancements to the Bluetooth pairing mechanism. The goal of Bluetooth Simple Pairing is to establish authentication credentials between the Bluetooth master and slave devices.

Bluetooth Simple Pairing supports four different pairing models: "Numeric Comparison," "Just Works," "Out of Band," and "Passkey Entry."

The **Numeric Comparison** model is intended when both devices can display a six digit number and both provide "Yes" and "No" buttons. For example, a PDA can use this pairing scheme with a PC. During the pairing process, each device displays a six digit number computed from the pairing protocol. The user of each device is supposed to compare the two numbers and select "Yes" if they match and "No" if they differ. Numeric Comparison is executed over Bluetooth, which is the in-band channel in the standard model for authentication. The display of the number on each device, the visual comparison of the numbers by human beings, and the Yes/No selection together comprise the out-of-band channel. Since there are six digits in the PIN ($= 10^6 \approx 2^{20}$ possibilities), an attacker can compromise the PIN with a probability of at least 2^{-20}.

The **Just Works** method is intended when at least one of the devices has no display or "Yes/No" buttons. A common use case is the pairing of a Bluetooth headset with a cell phone. This method uses Numeric Comparison internally, but does not display the six digits for comparison, even if one of the devices has a suitable display. Indeed, displaying the number is not useful, since the corresponding value cannot be compared on the putatively paired device. Because the Just Works method lacks any out-of-band channel required by the standard model, this method provides no security against active attack.

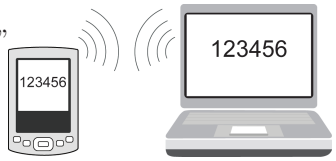

Fig. 1: Numeric Comparison

The **"Out-of-band"** method can be used when an alternate communication medium exists on both devices, such as Near Field Communication (NFC). The alternate communication medium transfers a key between the intended devices and functions as the out-of-band channel in the standard model. Two parameters determine the amount of security possible with this pairing method. First, transfer of a larger key can provide more security, particularly when compared to other methods. Second, the efficacy of the alternate communication channel to resist adversarial control is important in determining security. If an attacker can read or write the transferred data, then the credentials established by the method may be compromised. Hence, the security of this method depends fundamentally on the user properly exercising the alternate communication channel.

The **Passkey Entry** method is intended when one of the devices has a display and the other a keypad. The device with the display randomly generates a six-digit number, and the user enters this on the other device using the keypad. The displayed six-digit number, keypad, and human user together constitute the out-of-band channel for this method. Like Numeric Comparison, an attacker can compromise the six-digit passkey with a probability of at least 2^{-20}. However, this is only true the first time a passkey is used. The protocol splits the passkey into 20 bits and reveals one bit over 20 rounds of exchanges. An eavesdropper can compute each bit of the passkey after it has been sent. Thus, a passkey can only be used (securely) once.

Fig. 2: Passkey Entry

4 Wi-Fi Protected Setup

IEEE 802.11, commonly called Wi-Fi, is a Local Area Network standard [9]. It is widely used in laptop computers, PDAs, cell phones, bar code scanners, and other mobile devices with significant bandwidth requirements.

Wi-Fi is usually deployed as an infrastructure network, which consists of one or more access points, and one or more mobile devices called stations. Each station forms a connection, called an association, with a single access point.

Wi-Fi uses the 802.11i standard [10] for security. 802.11i is also called WPA2. WPA2 uses the IETF EAP protocol [11] to mutually authenticate a station and the network and to derive a session key. The session key provides confidentiality, integrity, and origin authenticity for each frame that a station and its access point exchange. Thus, Wi-Fi security relies on a long-lived authentication credential being established between the station and the network.

Wi-Fi Protected Setup was developed to address consumers' credential configuration problem. It is more complex than Bluetooth Simple Pairing; the host/peripheral model constrains the Bluetooth approach, while Wi-Fi attempts to address more complex relationships among wireless devices. The Wi-Fi scheme uses three different devices: the registrar, which is the network enrollment center; an access point; and an enrollee, which is the device being added to the network.

Wi-Fi Protected Setup supports three setup methods: Push Button Configuration, PIN entry, and Out-of-band channel.

Push Button Configuration (PBC) has no security in the standard model. The user pushes buttons on both the registrar and the enrollee devices. The button push causes both to initiate an unauthenticated Diffie-Hellman exchange. The method assumes that the Diffie-Hellman peer is the correct device, i.e., that a malicious active attacker is not present. There is no out-of-band channel.

The **PIN** method is the Wi-Fi Protected Setup default. The enrollee device has a four- or eight-digit PIN which is entered on the registrar's keypad. The PIN method uses the PIN as an authentication key to protect a Diffie-Hellman exchange. The transfer of the PIN from the enrollee device to the registrar is the out-of-band channel for the PIN method.

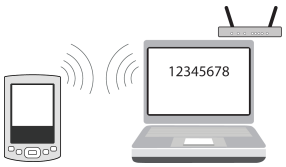

Fig. 3: PIN

A random eight-digit PIN represents $10^8 = 2^{26.65}$ possibilities. However, the PIN protocol splits the PIN into two four-digit numbers. Each side commits to its value for each half of the PIN and exchanges information to reveal the PIN value. A man-in-the-middle attacker can guess each half of the PIN separately [12]. This means the attack success probability for an 8-digit PIN is at least 2^{-14} (to establish a connection with the registrar). Like Bluetooth's Passkey Entry, a PIN can only be used once; otherwise, a man-in-the-middle attacker can reconstruct the PIN and establish a connection with both parties.

The Wi-Fi "**Out-of-band**" method is similar to the Bluetooth out-of-band method. An alternate communication channel, such as an NFC channel, transfers

	Out-of-band Channel in Standard Model	Probability of Attack Success
Bluetooth Simple Pairing		
Numeric Comparison	Display of 6-digit number on both devices; visual comparison of numbers and response input by human	2^{-20}
Just Works	None	Very likely
"Out-of-band"	The alternate communication channel	Depends on the selected channel
Passkey Entry	Display of 6-digit passkey on one device; entry of number into second device	2^{-20} for the first time a passkey is used
Wi-Fi Protected Setup		
Push Button Configuration	None	Very likely
PIN	Display of 4- or 8-digit PIN on one device; entry of PIN into second device	2^{-14} for the first time an 8-digit PIN is used; 2^{-7} for the first time a 4-digit PIN is used
"Out-of-band"	The alternate communication channel	Depends on the selected channel

Table 1: Security Characteristics of Individual Setup Models
Our secure setup requirement looks for no more than a 2^{-30} probability of attack success.

some information between the registrar and the enrollee. This transfer constitutes the out-of-band channel for the method. It is possible to obtain an arbitrary amount of security in the standard model, provided the user actively participates in protecting the alternate channel from attack.

Thus, like Bluetooth Simple Pairing, Wi-Fi Protected Setup can meet commonly accepted security levels only in the case of its Out-of-band method, and then only with the active cooperation of the user.

5 What Causes Poor Security?

Table 1 summarizes our discussion of each setup model. Not only are there security issues in individual setup models, the multitude of setup methods also introduces unnecessary complexity: any two given devices may support two arbitrary sets of setup models. In system safety engineering, this problem is called *interactive complexity*. A system is *interactively complex* "when the level of interactions reaches the point where they cannot be thoroughly planned, understood, anticipated, and guarded against. In interactively complex systems, designers find it difficult to consider all the potential system states and operators have difficulty handling all normal and abnormal situations and disturbances safely and effectively" [13].

Simple Pairing and Protected Setup are interactively complex. Bluetooth Simple Pairing specifies four pairing models, which mean there are $2^4 - 1 = 15$ combinations of setup models from which each vendor chooses. (Fifteen combinations is the power set of the four models, minus the empty set.) Between two Bluetooth devices, there are 120 possible setup combinations, which is the number of combinations (between two devices that each have 15 possible combinations of setup models) with repetition: $\binom{15+2-1}{15-1} = 120$. Similarly, Wi-Fi

Protected Setup supports three setup models, and there are $2^3 - 1 = 7$ combinations from which each vendor chooses. Between two Wi-Fi devices, there are $\binom{7+2-1}{7-1} = 28$ possible setup combinations.

The specifications need to anticipate how these different combinations may interact with one another. However, it is challenging to thoroughly evaluate 120 or even 28 combinations.

While accommodating the needs of many vendors, these options make any design and implementation more prone to mistakes. The specification contains more details, which security experts must review. Vendors must decide how many and which of the setup models to implement.

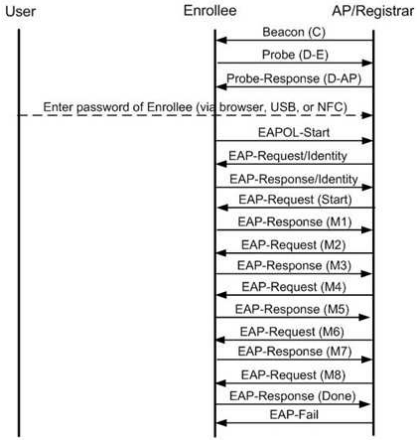

Fig. 4: Typical Protected Setup Protocol Diagram: PIN Enrollment in Wi-Fi Network

The number of combinations could be reduced by prioritizing the setup methods. For instance, suppose two Wi-Fi devices each support Push Button Configuration (PBC) and Out-of-band. Out-of-band should receive higher priority than PBC. Otherwise, an attacker could implement a dumbing-down attack, forcing the two devices to use the insecure PBC method. Wi-Fi does not – but Bluetooth does – prioritize setup models.

There is another security issue that deserves discussion. Four setup methods do not require screens: Simple Pairing's Just Works and Out of Band methods; and Protected Setup's Push Button Configuration (PBC) and Out-of-band methods. The lack of screen-based feedback to the user could magnify errors and facilitate attacks.

Just Works and PBC were designed specifically for devices without screens, such as Bluetooth headsets or Wi-Fi-enabled printers. Both methods rely on timing and proximity for their security. As long as there are no other devices in setup mode and in wireless range of the intended devices, setup occurs between the intended devices. As long as there are no malicious devices in wireless range, setup is secure. Clearly, the potential for unintended outcomes exists. For example, imagine using push button configuration on Christmas morning in New York City – neighbors might connect to each other's Wi-Fi networks accidentally.

The Out-of-band methods in Simple Pairing and Protected Setup also rely on device proximity, but the risk depends on the particular technology.

The security issues raised in this section can all be traced to the explosion of setup options. Each option increases the complexity of the setup process – and increases the possibility of mistakes in both design and implementation.

6 What Causes Poor Usability?

The multitude of setup methods not only detracts from the security of Simple Pairing and Protected Setup, it diminishes the usability as well. Simple Pairing and Protected Setup have not been introduced in consumer devices yet, but there are indications that they are too complicated. This is evident not in specification – but in what is *missing* from the specification. Many critical design choices remain undefined.

Both specifications focus on a narrow subset of the setup experience: the exchange of cryptographic keys. For example, Figure 4 shows a protocol diagram for enrollment in Wi-Fi Protected Setup using a PIN. Figure 4 indicates that the user only needs to enter the PIN number. Thus, the enrollment process appears simple. However, the diagram omits all the steps leading up to and following the credential exchange.

Appendix A lists some of the questions that implementers and end users will face. Unless vendors coordinate their implementation efforts (which is unlikely), many of the implementation questions will be pushed to end users. This means that the setup process may be far more involved than Figure 4 indicates. Figure 5 shows one plausible scenario for the end user experience of Wi-Fi Protected Setup. Note that Figure 5 is extremely optimistic, *ignoring potential errors* and questions of which device is the registrar.

Figure 5 also ignores subtleties in the Wi-Fi Protected Setup specification. For example, the end user decides whether a PIN will be copied from the enrollee or the registrar. This has important implications for network setup. Suppose an end user has an uninitiated device and an access point. Entering the AP's PIN onto the device means that the device will be authorized to act as an external registrar. Entering the device's PIN onto the AP means that the device will be enrolled in the network without registrar authority. The distinction is subtle, but the security implications may be significant.

Moreover, failing to address the questions in Appendix A could lead to non-interoperable software. For example, a potential enrollee may only support push button configuration; the registrar, produced by a security-conscious vendor, may only support PIN and NFC configuration. Setup will clearly fail. Without detailed specifications, implementers may make decisions that are incompatible with one another. This has the potential to create a non-interoperable system – *even if the underlying protocols interoperate*.

With the current specifications, we expect the following four usability issues will arise:

1. More setup models reduces consistency. Consistency allows users to apply what they learn from one situation to another, similar situation. It also increases users' confidence in their abilities, as they master applications quickly. Specifying several setup models reduces the consistency of the end user experience. The interaction flow from one setup model will be different from the flow of another model. As a result, the investment that users make in learning how to perform setup may not be fully leveraged. For example, learning how to compare numbers on Bluetooth devices may not benefit users when they set up

Bluetooth devices via NFC. Users may be further confused when they use the PIN method for Wi-Fi devices.

2. The quality of error handling, documentation, and technical support suffers. Without a consistent user experience on every device, a vendor cannot anticipate the setup process that users will encounter. Thus, each vendor can only build error handling mechanisms for one-half of the setup process. Each vendor can only document one-half of the setup process. If the product documentation fails, users call technical support. Technical support may not be able to support products from different vendors. Technical support then shifts the blame, instructing users to call the other vendor.

3. A failure to require confirmation of which devices are communicating may lead to confusion and errors. Many setup scenarios will include devices without screens. Without clear feedback from a screen, users may not receive confirmation that a wireless connection was successfully established. Without screens and adequate error messages, troubleshooting errors is nearly impossible. Also, if the credential exchange occurred via out-of-band channel, the devices should verify that the in-band connection was established successfully.

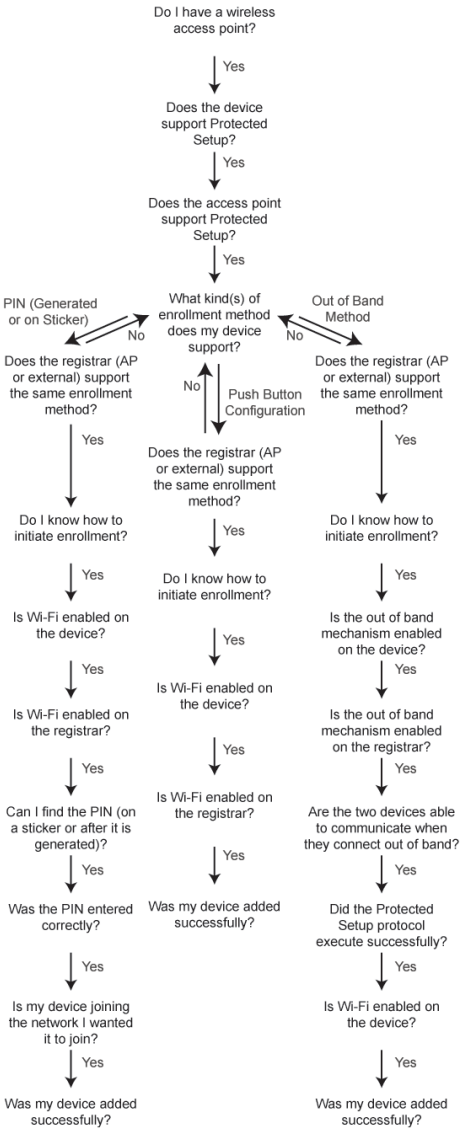

Fig. 5: Example End User Decision Tree in Wi-Fi Protected Setup (Omitting Potential Errors)

4. Users will not understand the level of security assurance associated with each setup model. As discussed in Sections 3 and 4, some setup models provide greater levels of security assurance than others. Simple Pairing and Protected Setup need to communicate that some connections are relatively secure, while others are not. Users should not be conducting sensitive business, for example, over connections established with Just Works or Push Button Configuration.

		In-band Channel (Bluetooth, Wi-Fi)					
		Higher Manufacturing Cost			**Lower Manufacturing Cost**		
	Output	Screen	Screen	Screen	LED	LED	LED
	Input	Keypad	2 Buttons	1 Button	Keypad	2 Buttons	1 Button
Higher Cost	Screen Keypad	4 | H	4 | H	4 | H	2 | M	1 | M	1 | M
	Screen 2 Buttons		3 | H	3 | H	2 | M	0 | M	0 | M
	Screen 1 Button			3 | H	2 | M	0 | M	0 | M
Lower Cost	LED Keypad				1 | L	1 | L	1 | L
	LED 2 Buttons					0 | L	0 | L
	LED 1 Button						0 | L

Secure Setup Mechanism (Please see Table 1 for attack success probabilities.)

0	N/A. A pair of devices that lack both screens and keypads will be set up insecurely.
1	One device has a keypad with which the user can enter a PIN number (or alphanumeric string). This device may or may not have a screen. The other device has neither a screen nor a keypad, but only a static PIN number (i.e., printed on a sticker). Cryptographically, a static PIN can only be considered secure the first time it is used.
2	One device has a screen on which to display a generated PIN. The other device has a keypad but no screen. Users type the generated PIN on the second device.
3	Both devices have screens but lack a full keypad. Both devices display a generated PIN, and users compare whether the PIN is the same.
4	Both devices have screens, and at least one has a keypad. Setup occurs in one of two ways: the comparison method in option (3), or the PIN input method in (2).

Feedback Capability

L	Low. Neither device has a screen to display success or error messages.
M	Medium. One device has a screen to confirm a successful setup or to display error messages. The size and capabilities of the screen obviously limit the quality of user feedback; this indicator focuses on feedback capability, *not* general usability.
H	High. Both devices have screens for displaying feedback.

Table 2: Tradeoffs between Manufacturing Cost, Secure Setup Mechanism, and Feedback Capability

7 Discussion

As we discussed in the previous two sections, interactive complexity will cause numerous security and usability issues. The number of setup models in Bluetooth Simple Pairing and Wi-Fi Protected Setup needs to be reduced – preferably to one or two scenarios. Preferably the scenario(s) would be consistent across Bluetooth, Wi-Fi, and other technologies.

We argue that the security and usability of Simple Pairing and Protected Setup can be improved by two simple actions: specifying (1) a common denominator for hardware features and (2) a common user interaction flow for application software. We elaborate on these two points below.

1. Common denominator of hardware features. First, devices must ship with a common feature set. For example, suppose that we require a level of security where an attacker needs to perform at least 2^{80} operations to break a system, or, assuming that an attacker can perform 2^{50} operations, that an attacker has an attack success probability of at most 2^{-30}. The Out-of-band method is the only one capable of meeting such stringent requirements. Suppose vendors choose to use NFC. Ideally, all devices ship with a screen, 2 buttons, and NFC capability. At the minimum, one device ships with a screen, 2 buttons, and NFC capability; the other device possesses at least an LED, 2 buttons, and NFC capability.

We summarize the tradeoffs between in-band device capabilities, secure setup models, and feedback capability in Table 2. The table for out-of-band setup differs in that the security mechanism is identical for all combinations. However, the feedback capability is identical to what is shown in Table 2.

At least one screen is needed so that success confirmations and descriptive error messages can be relayed to the user. Ideally, both devices possess a screen; this would increase consistency, enable better confirmation of which two devices are communicating (by displaying information about the other device on the screen), and facilitate error handling.

Less capable devices – such as devices without screens or keypads – cannot be introduced to each other securely. Vendors should consider whether it is worthwhile to push Bluetooth or Wi-Fi into these devices.

Retrofitting pre-existing hardware for a new security solution always requires some compromises. Bluetooth Simple Pairing and Wi-Fi Protected Setup are reasonable first steps for securing credential exchange. However, they should be viewed as transitory specifications. In the long run, the industry should aim for a single out-of-band channel that will be used for setup, whether it is NFC, USB, or some other technology.[2] Complementary technologies, such as decoy devices or scanners to detect attackers, can also be used to strengthen solutions.

2. Common user interaction flow for consistency in user interfaces. A common feature set is a necessary but not sufficient condition for creating a consistent user experience. Consistency does not mean that every user interface must be identical. Wireless setup can become more consistent simply by: ensuring that the setup application appears in the same location and has the same name and icons across devices; implementing a similar interaction flow across devices; and specifying a framework for error messages and troubleshooting procedures.

Relying on user interfaces as market differentiators is a dated concept. The setup operations for networked devices must be interoperable on the user experience level.

8 Related Work

To date, researchers have focused more on the security of Bluetooth Simple Pairing and Wi-Fi Protected Setup than the usability. Suomalainen et al. analyze the security of several setup methods, including Simple Pairing and Protected Setup [12]. Nyberg presents a Man-in-the-Middle attack on (an earlier version of) Protected Setup [14]. (Researchers also noted vulnerabilities in earlier versions of Bluetooth pairing [15–17].) Uzun et al. compare the usability of different pairing methods and their implementations [18].

Newman et al.'s description of setup in HomePlug AV raises many design issues also discussed here [19]. In HomePlug, users select from two setup modes: Simple Connect Mode and Secure Mode. Simple Connect Mode is similar to Wi-Fi's Push Button Configuration. Secure Mode is analogous to Bluetooth's

[2] There is a precedent for hardware changes in the design of 802.11i. 802.11i uses AES in CCM mode as its long-term solution. However, when 802.11i was designed, the majority of legacy Wi-Fi devices had microprocessors with insufficient available MIPs to support AES. 802.11i provides TKIP as a patch that can be deployed on legacy hardware.

	Length	Composition	Printed on Sticker or Generated on Screen?
HomePlug	12	Alphanumeric	Printed on sticker
Bluetooth	6	Numeric	Generated on screen
Wi-Fi	4 or 8	Numeric	May be printed on sticker, but generated on screen preferred; 8-digit PINs recommended; 4-digit PINs acceptable for less capable screens

Table 3: HomePlug Secure Mode vs. Bluetooth Passkey Entry and Wi-Fi PIN Methods

Passkey Entry and Wi-Fi's PIN methods; the three setup methods are compared in Table 3.

Like Bluetooth and Wi-Fi, HomePlug is designed to support a wide range of device capabilities, from computers to devices without screens and keyboards. Simple Connect Mode can be used with any device. If a device is accidentally recruited into the wrong network, a user resets the station until the correct network is found. If a rogue device is detected on a network, a user must reform the entire network to remove the device. Accidental recruitment will not occur with Secure Mode, but a sufficient user interface must be available.

Other schemes for exchanging authentication credentials using demonstrative identification include Stajano and Anderson's Resurrecting Duckling [20], Balfanz et al's Talking to Strangers [6], Balfanz et al's Network-in-a-Box [21], and McCune et al's Seeing-Is-Believing [7]. In Resurrecting Duckling, an uninitialized network node uses the first key that it receives. Ideally, the imprinted key is transferred using an out-of-band channel. Talking to Strangers proposes using location-limited channels, such as audio or infrared, for credential exchange. This idea is extended in Network-in-a-Box, which uses infrared to secure a wireless network. In Seeing-Is-Believing, cell phone cameras take pictures of 2D barcodes, which encode public keys. For mutual authentication, each device displays its unique 2D barcode, and the opposite device takes a picture of the barcode. Devices can also act as intermediaries for less capable devices.

9 Conclusion

This focus of this paper is not security setup per se; it is about making setup processes consistent. Consistency makes setup more usable – and by extension, more secure. Security features can only benefit consumers if setup is successful.

Many of the problems in Bluetooth Simple Pairing and Wi-Fi Protected Setup stem from the multitude of setup methods available. Several methods exist to accommodate vendors who opt for lower manufacturing costs. The feature sets selected for lower costs *force* system designers to use setup methods that are neither usable nor secure.

Simple Pairing and Protected Setup could be improved by:

1. requiring a common set of hardware features for compliant devices; and
2. specifying a consistent user experience, via common menu options, common user interaction flows, and a common framework for error logging.

This requires that the specifications converge to a small number of setup scenarios – preferably one, maybe two. It may raise some vendors' manufacturing costs, but consumers will be better able to setup wireless devices themselves.

Acknowledgements

The authors would like to thank the anonymous reviewers and N. Asokan for their helpful comments. This research was supported in part by CyLab at Carnegie Mellon under Grant DAAD19-02-1-0389 from the Army Research Office, a National Science Foundation Graduate Research Fellowship, Grant CNS-0627357 from the National Science Foundation, and by a gift from Intel. The views and conclusions contained here are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either express or implied, of ARO, CMU, Intel, NSF, or the U.S. Government or any of its agencies.

References

1. Bluetooth SIG: Authorities raid Chinese factory suspected of infringing on Bluetooth SIG registered trademarks. `http://www.bluetooth.com/Bluetooth/Press/SIG/AUTHORITIES_RAID_CHINESE_FACTORY_SUSPECTED_OF_INFRINGING_ON_BLUETOOTH_SIG_REGISTERED_TRADEMARKS.htm` (September 2006)
2. In-Stat: Year over year Wi-Fi chipset sales. Personal Communication, Kelly Davis-Felner (October 2006)
3. Linsky, J., Bourk, T., Findikli, A., Hulvey, R., Ding, S., Heydon, R., Singer, S., Kingston, S., Wenham, S., Suvak, D., Edlund, M., Chen, P., Aissi, S., Hauser, P., Benaloh, J., Yuval, G., Yacobi, Y., Lafky, J., Simon, D., Roberts, D., Stanwyck, D., Lauter, K., Muchnik, G., Kerai, K., Nyberg, K., Asokan, N., Lobo, N., Ginzboorg, P., Everaere, D., Meindl, R., Bertoni, G., Reuveni, E., Shimojo, Y.: Simple Pairing Whitepaper, revision v10r00. `http://www.bluetooth.com/NR/rdonlyres/0A0B3F36-D15F-4470-85A6-F2CCFA26F70F/0/SimplePairing_WP_V10r00.pdf` (August 2006)
4. Lortz, V., Roberts, D., Erdmann, B., Dawidowsky, F., Hayes, K., Yee, J.C., Ishidoshiro, T.: Wi-Fi Simple Config Specification, version 1.0a. (February 2006)
5. Barker, E., Barker, W., Burr, W., Polk, W., Smid, M.: National Institute of Standards and Technology (NIST) Special Publication 800-57 (Draft): Recommendation for Key Management - Part 1 General (Revised) (May 2006)
6. Balfanz, D., Smetters, D., Stewart, P., Wong, H.C.: Talking to Strangers: Authentication in ad-hoc wireless networks. In: Proceedings of the Symposium on Network and Distributed Systems Security (NDSS 2002), San Diego, CA, Internet Society (February 2002)
7. McCune, J.M., Perrig, A., Reiter, M.K.: Seeing-is-believing: Using camera phones for human-verifiable authentication. In: Proceedings of the IEEE Symposium on Security and Privacy. (May 2005)
8. IEEE: IEEE 802.15.1-2005 – IEEE Standard for Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific Requirements. Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Networks (WPANs(tm)) (2005)
9. IEEE: IEEE 802.11-1999 – IEEE Standard for Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific Requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications (2003)

10. IEEE: IEEE 802.11-1999 – IEEE Standard for Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific Requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications – Amendment 6: Medium Access Control Security Enhancements (2004)
11. Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., Levkowetz, H.: RFC 3748: Extensible Authentication Protocol (June 2004)
12. Suomalainen, J., Valkonen, J., Asokan, N.: Security associations in personal networks: A comparative analysis. Technical report, Nokia Research Center, Technical Report NRC-TR-2007-004 (2007)
13. Leveson, N.: System Safety Engineering: Back to the Future. `http://sunnyday.mit.edu/book2.pdf` (2002)
14. Nyberg, K.: Connect Now to MitM. Presentation at Crypto 2006 Rump Session (August 2006)
15. Jakobsson, M., Wetzel, S.: Security weaknesses in Bluetooth. In: CT-RSA 2001: Proceedings of the 2001 Conference on Topics in Cryptology, London, UK, Springer-Verlag (2001) 176–191 LNCS 2020.
16. Whitehouse, O. Presentation at RUXCON, `http://www.ruxcon.org.au/files/2004/12-ollie_whitehouse.pdf` (2004)
17. Shaked, Y., Wool, A.: Cracking the Bluetooth PIN. In: The Third International Conference on Mobile Systems, Applications, and Services (MobiSys). (June 2005) 39–50
18. Uzun, E., Karvonen, K., Asokan, N.: Usability analysis of secure pairing methods. In: Usable Security (USEC'07). (February 2007)
19. Newman, R., Gavette, S., Yonge, L., Anderson, R.: Protecting domestic power-line communications. In: Symposium On Usable Privacy and Security (SOUPS). (July 2006)
20. Stajano, F., Anderson, R.: The Resurrecting Duckling: Security issues for ad-hoc wireless networks. In: Security Protocols—7th International Workshop. Volume 1796., Cambridge, United Kingdom (2000) 172–194
21. Balfanz, D., Durfee, G., Grinter, R.E., Smetters, D.K., Stewart, P.: Network-in-a-Box: How to set up a secure wireless network in under a minute. In: Proceedings of the 13th USENIX Security Symposium, USENIX (August 2004)

A Questions Left Unanswered in the Bluetooth and Wi-Fi Specifications

A.1 Implementers

- Who initiates the setup process? Does the user set both devices into setup mode? Does one device always look for other devices in setup mode?
- Where is Protected Setup/Simple Pairing application located in OS menu? How does a user initiate setup?
- Does the application check if wireless is enabled? Should wireless be turned on automatically?
- If multiple setup methods are available, which method will be used for setup? Who will decide? The devices? Will more secure methods take precedence over less secure methods? Will users decide?
- For Bluetooth Numeric Comparison and Wi-Fi PIN methods: Which device (if any) generates the PIN? How is this decided? By the devices or the user?
- Is there a timeout value for a generated PIN? What is it?
- For the Bluetooth Just Works scenario, should a device just accept a connection, or prompt the user?
- Is there a timeout value for Just Works mode? What is it?
- Which device or manual (if any) provides directions on what the user should do?
- Which setup methods will a device support?
- Which device (if any) is logging data to aid troubleshooting?
- For Wi-Fi Protected Setup: Does the access point need to be present during enrollment? What happens if the enrollee and the registrar are out of WLAN range?
- For Wi-Fi Protected Setup: What device keeps a record of the keys that have been issued?
- If the out of band channel is used for setup, will the in-band connection be verified?

A.2 Users

- My AP has a PIN, my phone has a PIN, my computer has a PIN. Which PIN do I enter where?
- Which devices or manuals should I consult to confirm whether setup succeeded?
- Whose tech support line should I call if setup failed?