Initializing Security Associations for Personal Devices

N. Asokan
Nokia Research Center, Helsinki
TKK - Helsinki University of Technology

ZISC workshop on Wireless Security, September 2007.
Latest version of the presentation available at http://asokan.org/asokan/research/fc-tutorial.pdf

NOKIA Connecting People

Outline

- The problem: What is "First Connect" and why is it hard to secure?
- Proposed solutions: recent efforts addressing this issue in
  - research literature
  - standard specifications
- Usability analysis and some open issues

NOKIA Connecting People

The problem

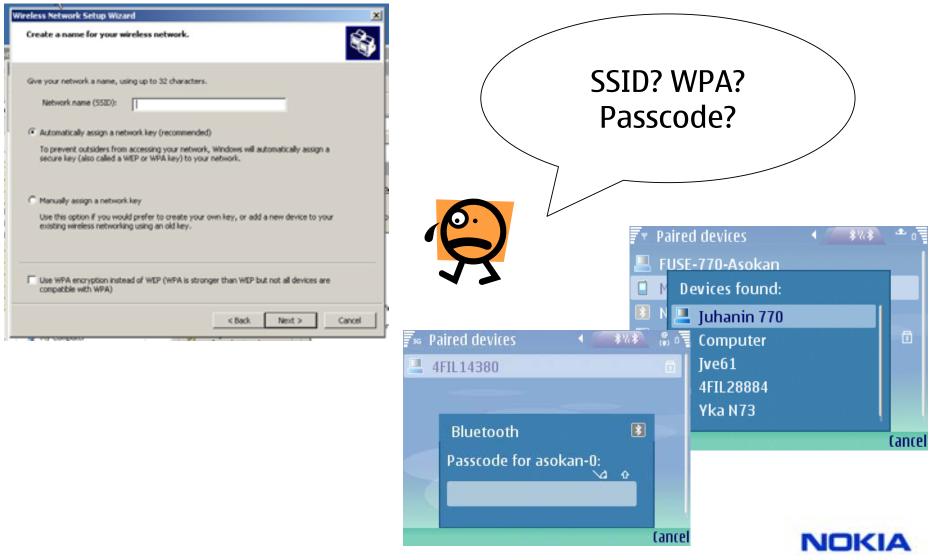
NOKIA Connecting People

Setting up the first connection

- **First Connect**: setting up contexts for subsequent communication.
  - Typically for proximity communications between personal devices, e.g.:
  - Pairing a Bluetooth phone and headset
  - Enrolling a Phone or PC in the home WLAN
  - More instances to come: Wireless USB, WiMedia
- **Problem**: Secure First Connect for personal devices
  - Initializing security associations (as securely as possible)
  - No security infrastructure (no PKI, key servers etc.)
  - Ordinary non-expert users
  - Cost-sensitive commodity devices

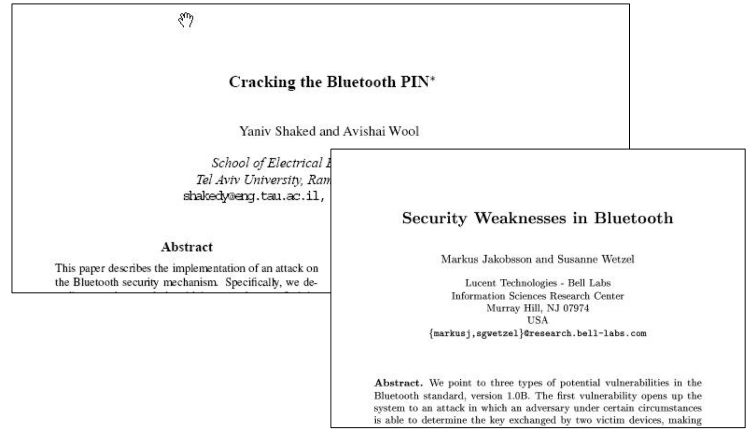
NOKIA Connecting People

Current mechanisms are not intuitive ...



NOKIA Connecting People

... and not very secure



Cracking the Bluetooth PIN*

Yaniv Shaked and Avishai Wool

School of Electrical E
Tel Aviv University, Ram
shakedy@eng.tau.ac.il,

Abstract

This paper describes the implementation of an attack on the Bluetooth security mechanism. Specifically, we de-

Security Weaknesses in Bluetooth

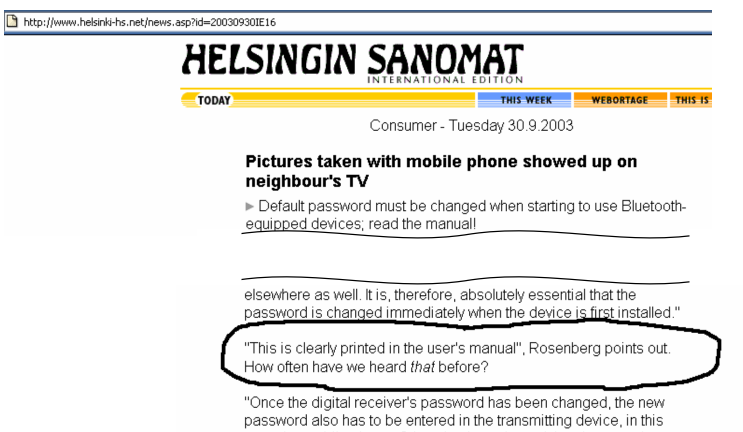
Markus Jakobsson and Susanne Wetzel

Lucent Technologies - Bell Labs
Information Sciences Research Center
Murray Hill, NJ 07974
USA
{markusj,sgwetzel}@research.bell-labs.com

Abstract. We point to three types of potential vulnerabilities in the Bluetooth standard, version 1.0B. The first vulnerability opens up the system to an attack in which an adversary under certain circumstances is able to determine the key exchanged by two victim devices, making

NOKIA Connecting People

Naïve usability measures damage security



http://www.helsinki-hs.net/news.asp?id=20030930IE16

HELSINGIN SANOMAT
INTERNATIONAL EDITION

TODAY | THIS WEEK | WEBORTAGE | THIS IS

Consumer - Tuesday 30.9.2003

Pictures taken with mobile phone showed up on neighbour's TV

▶ Default password must be changed when starting to use Bluetooth-equipped devices; read the manual!

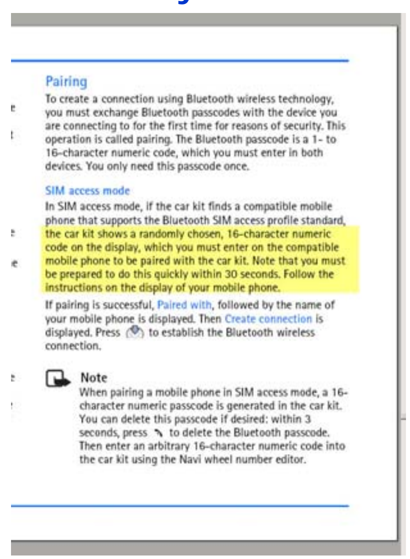
elsewhere as well. It is, therefore, absolutely essential that the password is changed immediately when the device is first installed."

"This is clearly printed in the user's manual", Rosenberg points out. How often have we heard *that* before?

"Once the digital receiver's password has been changed, the new password also has to be entered in the transmitting device, in this

NOKIA Connecting People

Naïve security measures damage usability



Pairing

To create a connection using Bluetooth wireless technology, you must exchange Bluetooth passcodes with the device you are connecting to for the first time for reasons of security. This operation is called pairing. The Bluetooth passcode is a 1- to 16-character numeric code, which you must enter in both devices. You only need this passcode once.

SIM access mode

In SIM access mode, if the car kit finds a compatible mobile phone that supports the Bluetooth SIM access profile standard, the car kit shows a randomly chosen, 16-character numeric code on the display, which you must enter on the compatible mobile phone to be paired with the car kit. Note that you must be prepared to do this quickly within 30 seconds. Follow the instructions on the display of your mobile phone.

If pairing is successful, Paired with, followed by the name of your mobile phone is displayed. Then Create connection is displayed. Press (✓) to establish the Bluetooth wireless connection.

Note

When pairing a mobile phone in SIM access mode, a 16-character numeric passcode is generated in the car kit. You can delete this passcode if desired: within 3 seconds, press ↖ to delete the Bluetooth passcode. Then enter an arbitrary 16-character numeric code into the car kit using the Navi wheel number editor.

- Car kits allow a car phone to retrieve and use session keys from a mobile phone smartcard
- Car kit requires higher level of security
  - ➢ users have to enter 16-character passcodes

More secure = Harder to use?

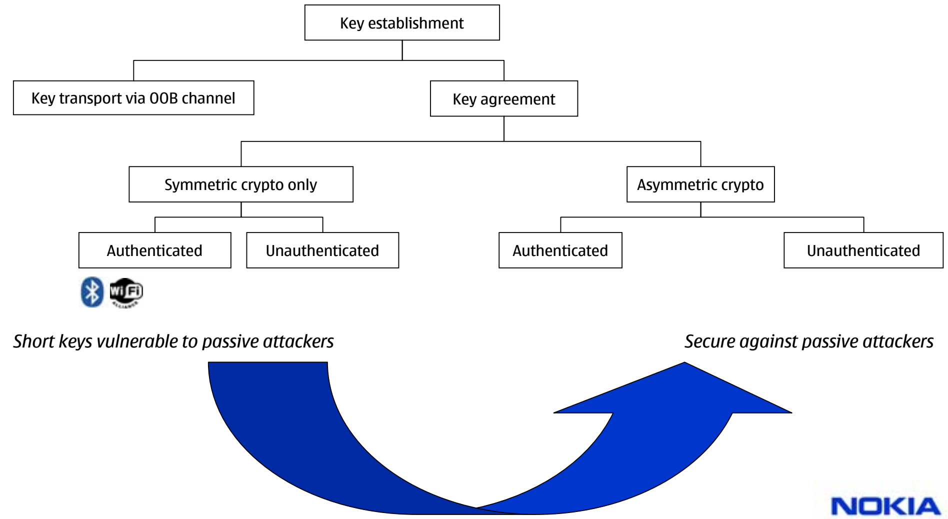
NOKIA Connecting People

Wanted: Secure, intuitive, inexpensive first connect

- Two (initial) problems to solve
  - Peer discovery: finding the other device
  - **Authenticated key establishment**: setting up a security association
- Assumption: Peer devices are physically identifiable

Key establishment protocols for first connect (1)

We will update this chart as we go along

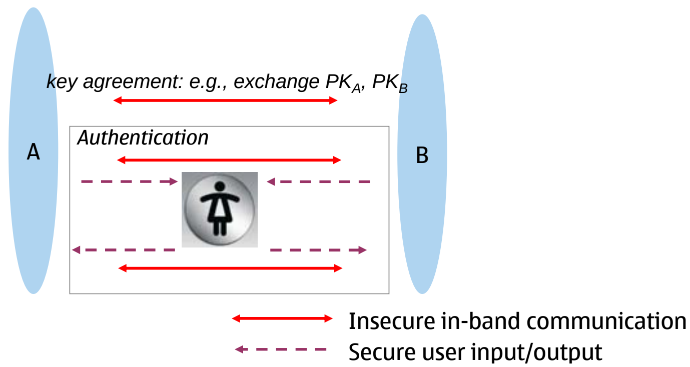


Proposed solutions: research literature

Authenticating key agreement

- Use an auxiliary channel to transfer information needed for authentication
- Two possibilities for realizing secure channel
  - User assistance
  - Out-of-band secure channels: physical communication channel
    - E.g., Near Field Communication, infrared, ...

Authenticating key agreement: user-assisted

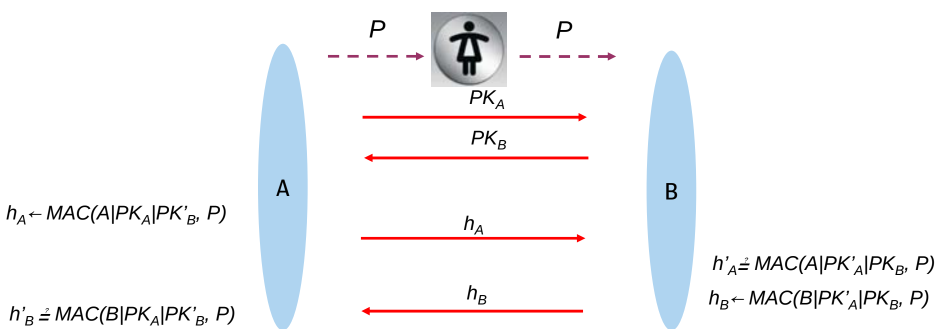


- User "bandwidth" is low (4 to 6 digits)
- Directionality depends on available hardware (1-way or 2-way)
- Security properties (integrity-only, or integrity+secrecy)

User as the secure channel

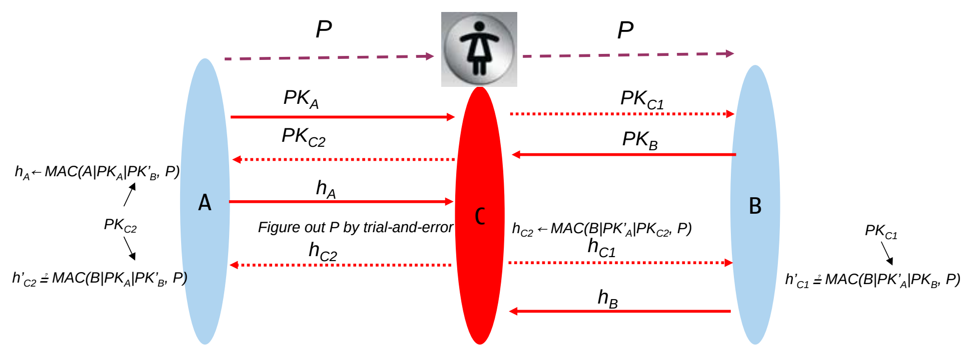
- Peer discovery by "user conditioning": introduce a special first connect mode
  - E.g., Press a button to put device into the special mode
  - Demonstrative/indexical identification
- Authentication by
  - entering a **short secret** Passkey, or
  - Comparing **short** non-secret check codes (aka "short authentication string")
- Short key/code should not hamper security
  - Standard security against offline attacks
  - Good enough security against active man-in-the-middle

Authentication using a short passkey: a first attempt



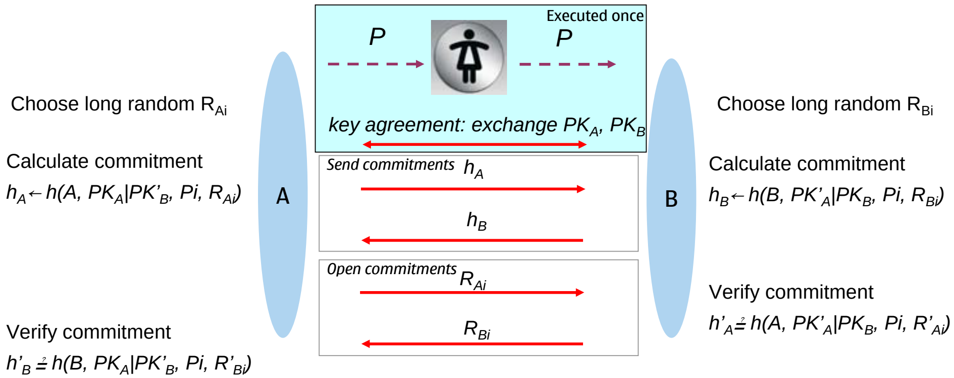
P is a short passkey (e.g., 4 digits)
MAC() is a message authentication code: e.g., HMAC-SHA1
But a man-in-the-middle can easily defeat this protocol!

Man-in-the-middle in authentication using a short passkey



Guess a value x for P; calculate h_x = *MAC(A|PK'$_A$|PK$_{C2}$, X)*; Check $h_A \stackrel{?}{=} h_x$
If P is a n-digit PIN, attacker needs at most 10^n guesses; Each guess costs one MAC calculation
A typical modern PC can calculate 100000 MACs in 1 second

Authentication using interlocking short passkeys

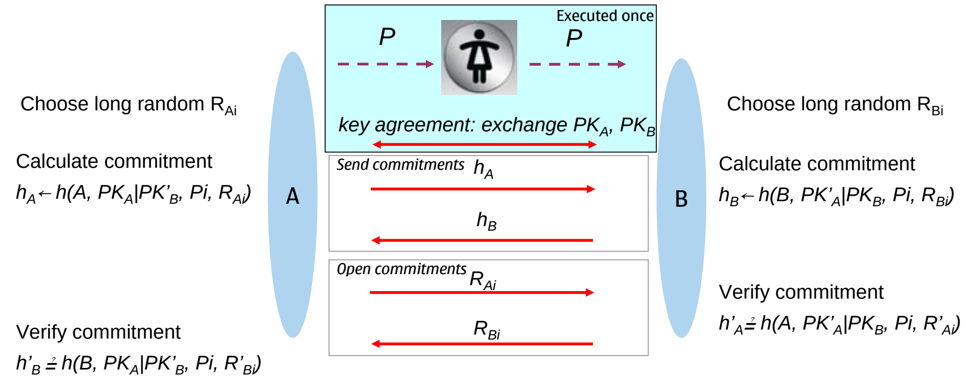


One-time passkey *P* is split into *k* parts ($k > 1$): next 4-round exchange repeated *k* times

h() is a hiding commitment; in practice SHA-256

Up to $2^{-(l-1)}$ ("unconditional") security against man-in-the-middle (l is the length of *P*)

Authentication using interlocking short passkeys



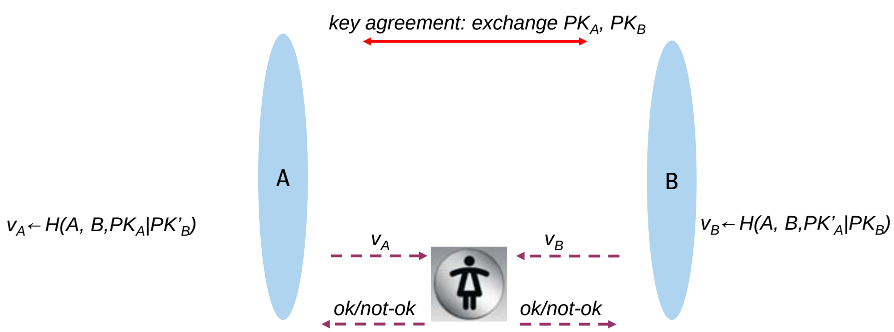
One-time passkey *P* is split into *k* parts ($k > 1$): next 4-round exchange repeated *k* times

h() is a hiding commitment; in practice SHA-256

Up to $2^{-(l-1)}$ ("unconditional") security against man-in-the-middle (l is the length of *P*)

Originally proposed by Jan-Ove Larsson [2001]: essentially multi-round MANA III

Authentication by comparing short strings: a first attempt

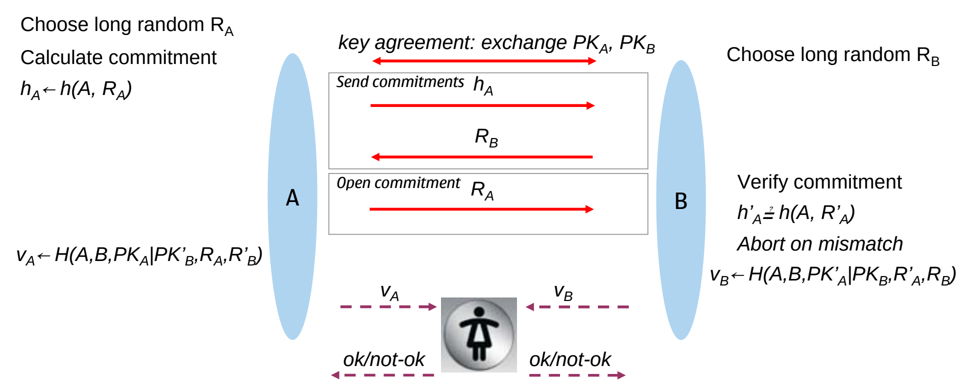


v_A and v_B are short strings (e.g., 4 digits),

User approves acceptance if v_A and v_B match

As before, a man-in-the-middle can easily defeat this protocol

Authentication by comparing short strings



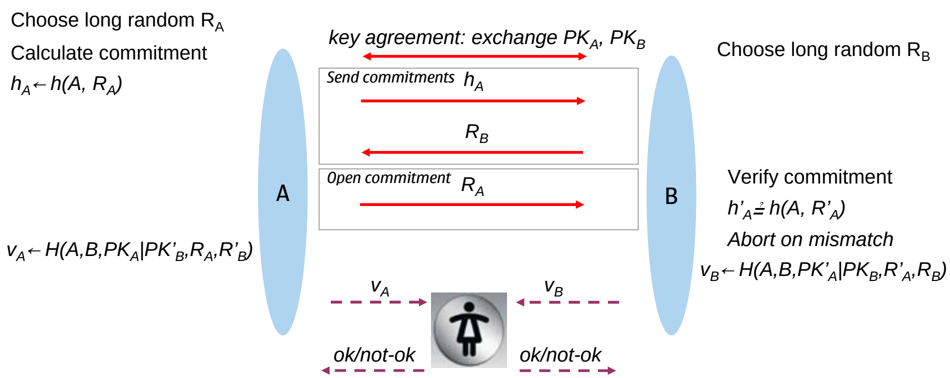
User approves acceptance if v_A and v_B match

2^{-l} ("unconditional") security against man-in-the-middle (l is the length of v_A and v_B)

h() is a hiding commitment; in practice SHA-256

H*()* is a mixing function; in practice SHA-256 output truncated

Authentication by comparing short strings



User approves acceptance if v_A and v_B match

2^{-l} ("unconditional") security against man-in-the-middle (l is the length of v_A and v_B)

h() is a hiding commitment; in practice SHA-256

MANA IV by Laur, Asokan, Nyberg [IACR report] Laur, Nyberg [CANS 2006]

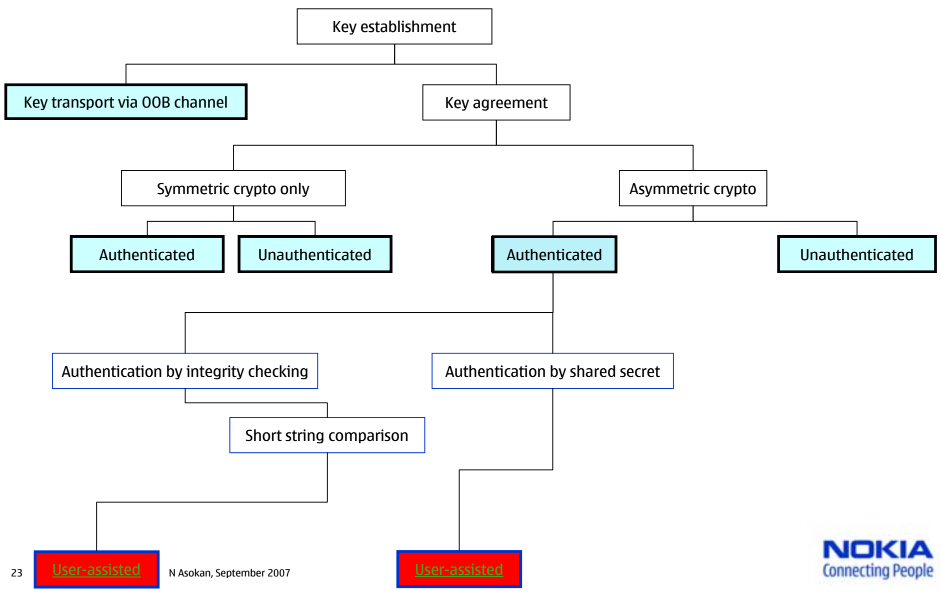
NOKIA Connecting People

Authentication by comparing short strings

- Initially due to Zimmerman in PGPfone biometric authentication [1996]
- Recent variations: reuse of public keys, formal analyses
  - Gehrmann et al, Čagalj et al, Vaudenay et al, Pasini et al, Laur et al, ...

NOKIA Connecting People

Key establishment protocols for first connect (2)



NOKIA Connecting People

Problems with user-as-secure-channel

- Relies on availability of specific hardware (display, keypad, buttons, ...)
- Needs a negotiation protocol
- What about usability?

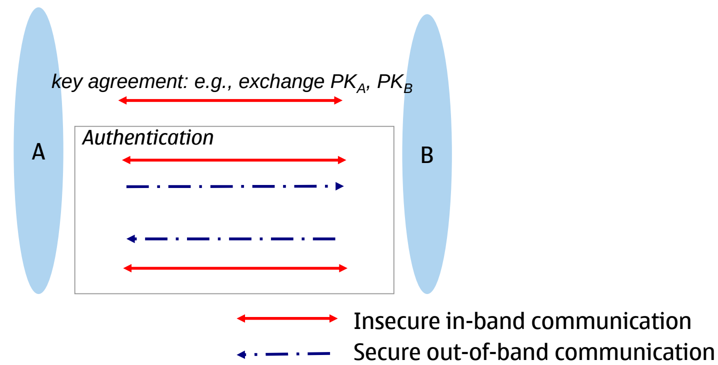
NOKIA Connecting People

Out-of-band secure channel

- Idea: use a physically secure channel to transfer security critical information
  - Minimize user involvement → better usability
- Peer discovery is intuitive
  - Demonstrative/indexical identification
- Channel must have certain security properties
  - integrity (tampering with messages can be detected)
  - Sometimes secrecy as well

NOKIA Connecting People

Authenticating key agreement: out-of-band channel



Different out-of-band channels have different
- Bandwidth
- Directionality (1-way or 2-way)
- Security properties (integrity-only, or integrity+secrecy)

NOKIA Connecting People

What out-of-band channels can you think of?

- Near Field Communication
  - "touch" to connect

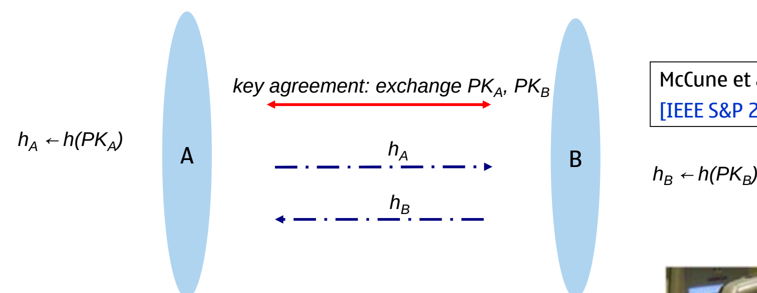
- Audio

- Visual

- Body-area communication
  - *touch* to connect
- ...

NOKIA Connecting People

Seeing Is Believing



McCune et al,
[IEEE S&P 2005]

Rohs, Gfeller
[PervComp'04]

NOKIA Connecting People

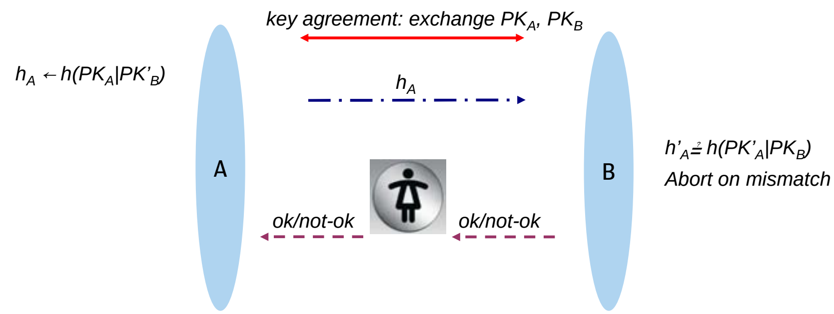
Drawbacks of SiB

1. Mutual authentication requires that <u>both</u> devices have cameras and switch roles
   - ➔ Slow and difficult for the user!

   Potential solution: one-way visual channel + user confirmation
2. Not all devices have big enough displays to show two-dimensional bar codes
   - Typically these constrained devices do not have cameras either

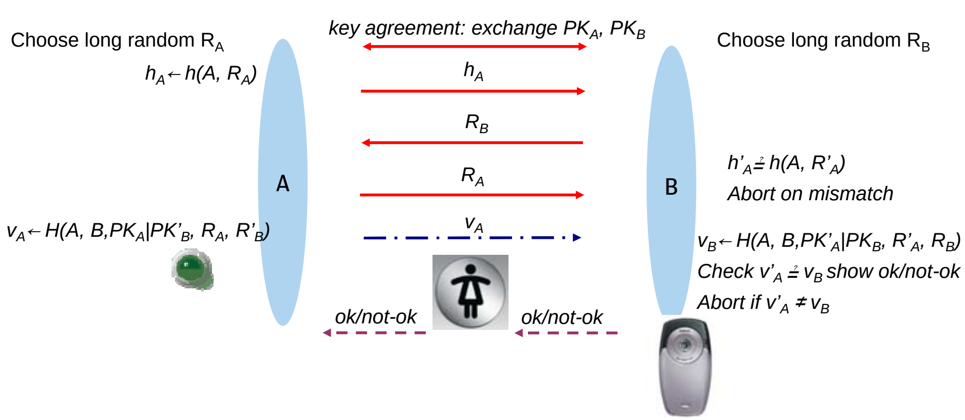
Problem: secure first connect for constrained devices with **minimal additional hardware**?

Mutual authentication with one-way visual channel



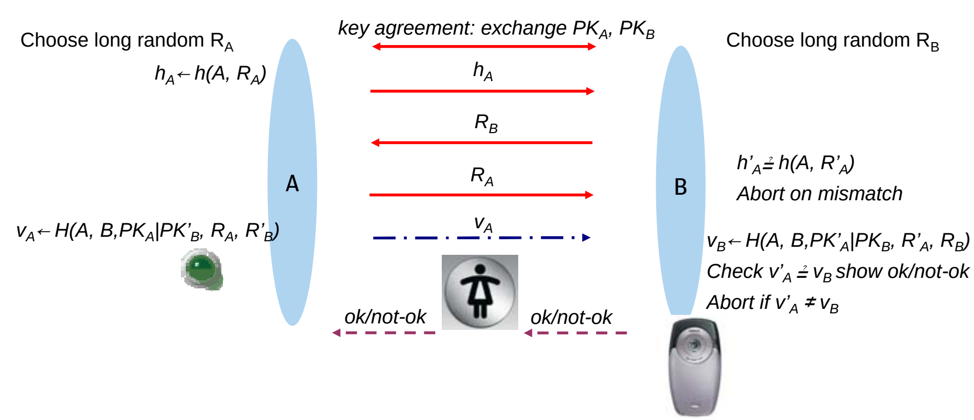
Supporting display constrained devices

Use a short authentication string protocol like MANA IV



Supporting display constrained devices

Use a short authentication string protocol like MANA IV



Saxena, Ekberg, Kostiainen, Asokan [IEEE S&P 2006]

Supporting display constrained devices

Pairing phone and laptop with LED

Pairing two phones

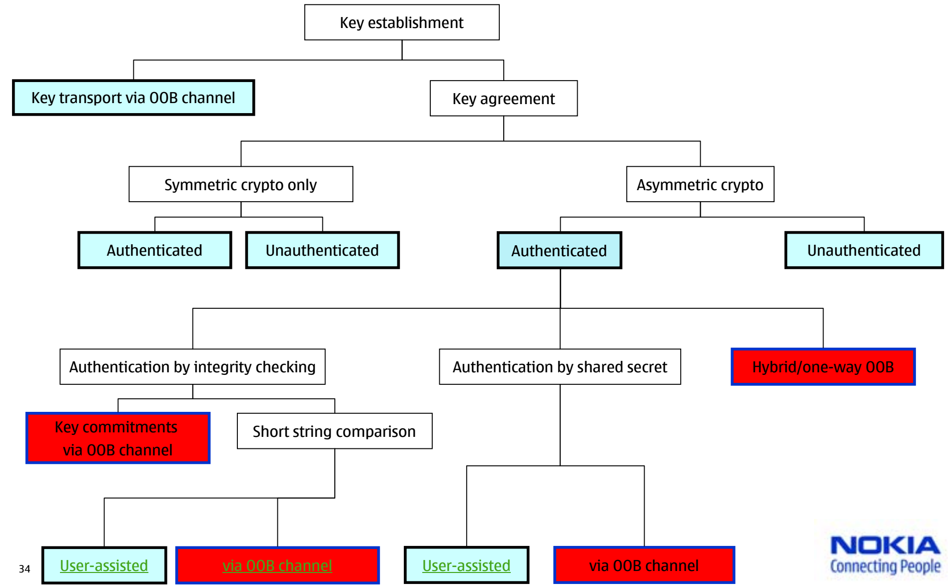
Suitable for access points, wireless headsets

Hardware needed:

- Single LED (cheap)
- Video camera (common on smartphones)

Saxena, Ekberg, Kostiainen, Asokan [IEEE S&P 2006]

Key establishment protocols for first connect (3)



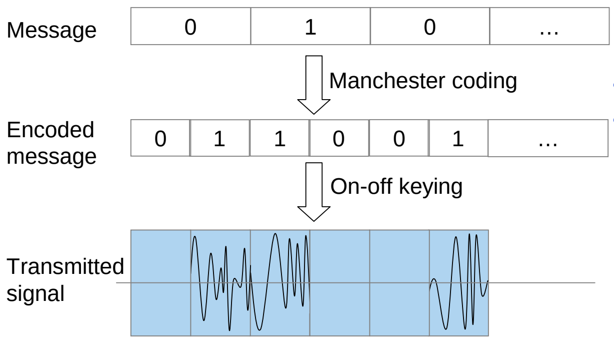
Problems with out-of-band channels

- Cost
  - Availability of specific (possibly new) hardware interfaces
- Deployability
  - Universally deployed auxiliary channel needed
  - Otherwise how to discover common auxiliary channels between the devices?
    - Leave-it-to-the-user: visible well-known logos
    - Negotiation protocol

Can we use the radio interface itself for authentication?

- In-band integrity checking
  - Assumption: genuine device emits energy during transmission; a distant attacker cannot easily drown this out
  - I-codes by Čagalj et al
- Common radio environment
  - Assumption: genuine devices hear the same radio signals; a distant attacker likely hears something different
  - Amigo by Varshavsky et al
- Spatial indistinguishability
  - Assumption: a distant attacker cannot tell which device is transmitting
  - Shake-them-up by Castelluccia et al

Integrity protection in-band: I-Codes

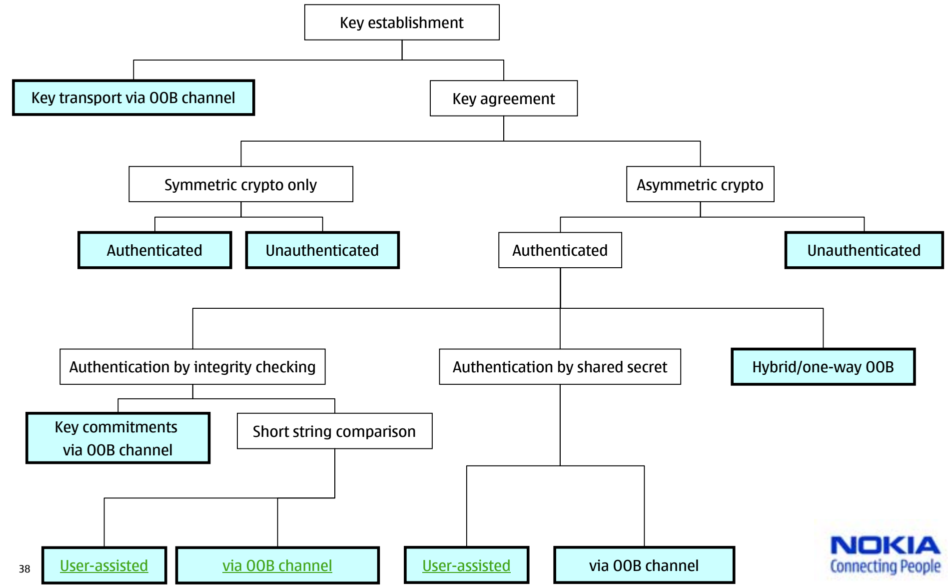


- Recipient measures the presence/absence of energy (1-bit/0-bit)
- Attacker cannot change 1→0
- Issues
  - Modifications to lower layers in the communication stack
  - No genuine radio interference

Čagalj, Čapkun, Rengaswamy, Tsigkogiannis, Srivastava, Hubaux [IEEE S&P 2006]

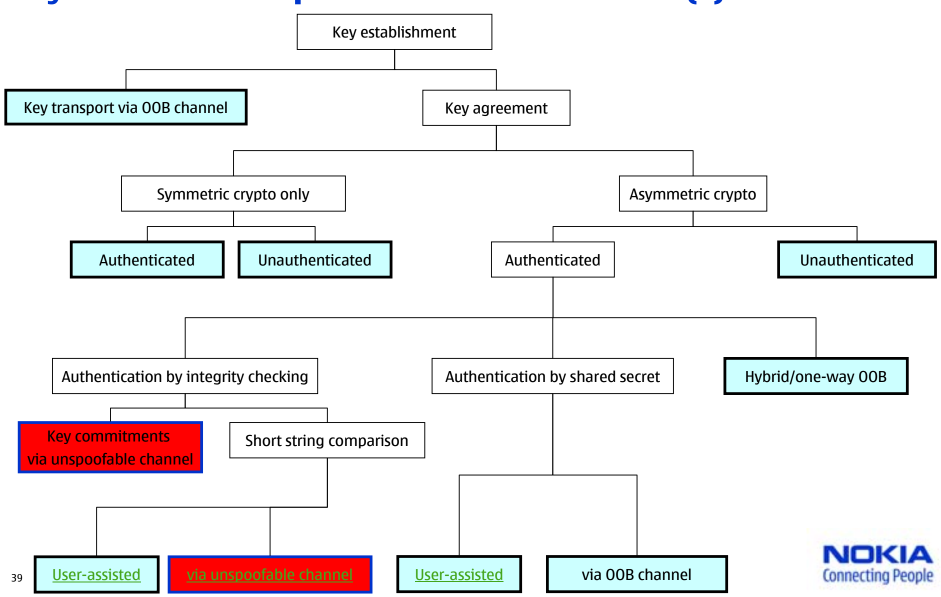
NOKIA Connecting People

Key establishment protocols for first connect (2)



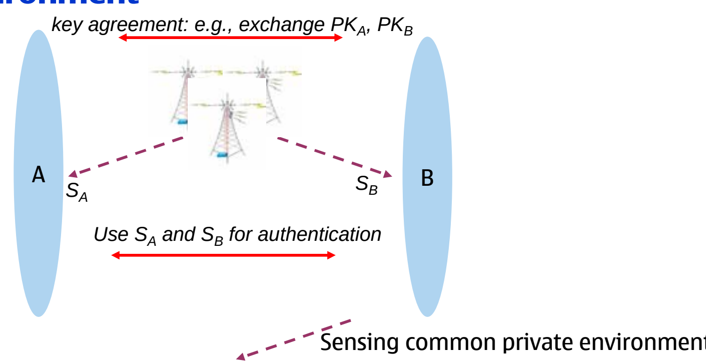
NOKIA Connecting People

Key establishment protocols for first connect (3)



NOKIA Connecting People

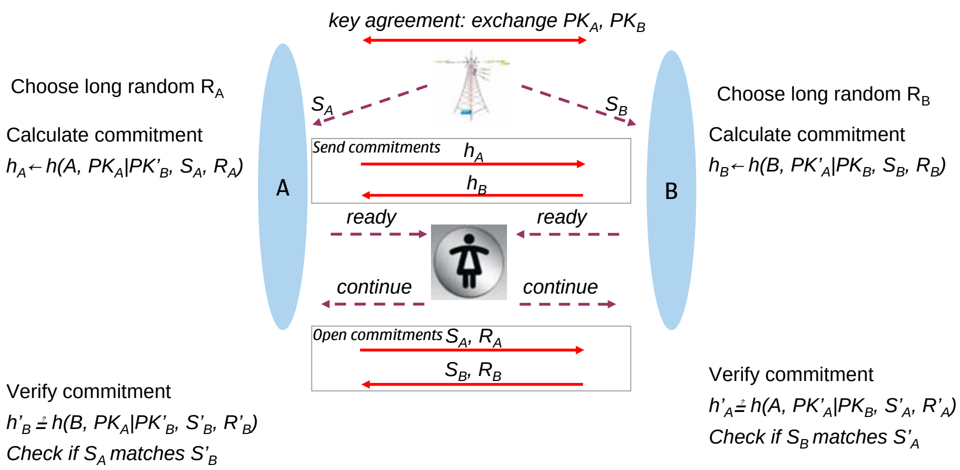
Authenticating key agreement: secret extraction from common environment



- Measure some environmental features
  - For co-located (in space and time) sensors measurements should be *almost* identical
  - For anyone else, measurement must be unpredictable
- Radio signal strength [Varshavsky, Scanneli, LaMarca, de Lara, HotMobile 2007, UBICOMP 2007]
- Accelerometer readings [Mayrhofer and Gellersen, Pervasive 2007]

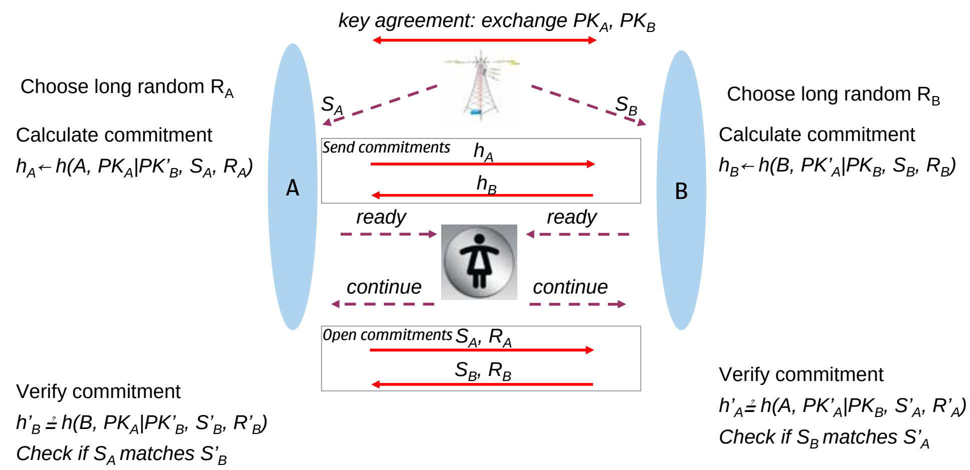
NOKIA Connecting People

Authentication using interlocking extracted secrets



h() is a hiding commitment; in practice SHA-256

Authentication using interlocking extracted secrets



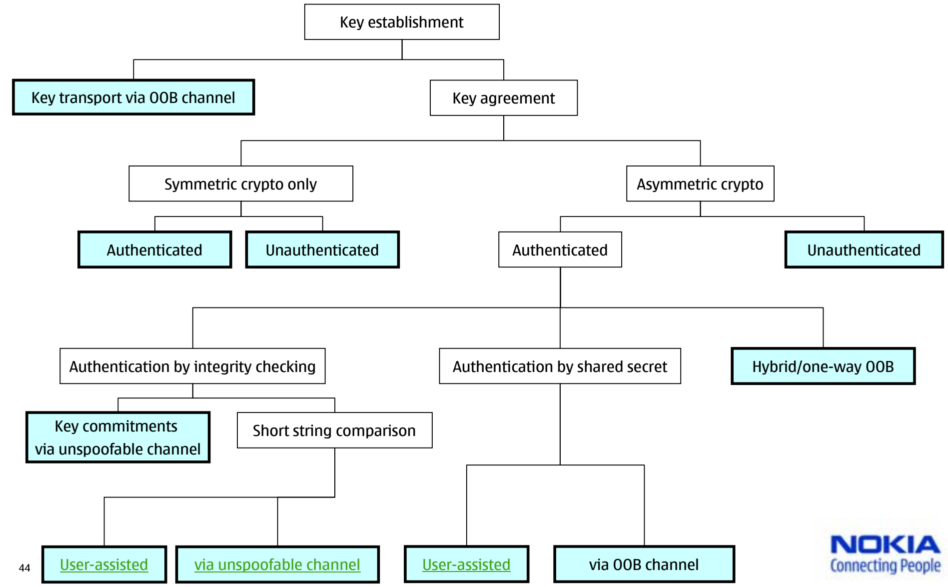
h() is a hiding commitment; in practice SHA-256

Application of MANAIII by Gehrmann, Nyberg, Mitchell [RSA Cryptobytes 2004]

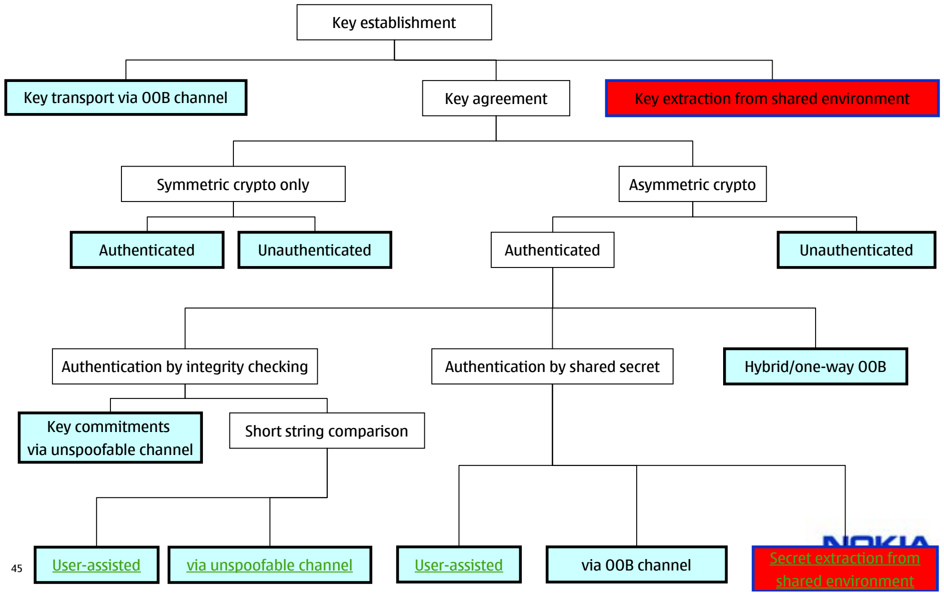
Issues with secret extraction

- User involvement
- Are the assumptions valid?
- If a long shared secret can be extracted, key agreement may not be necessary

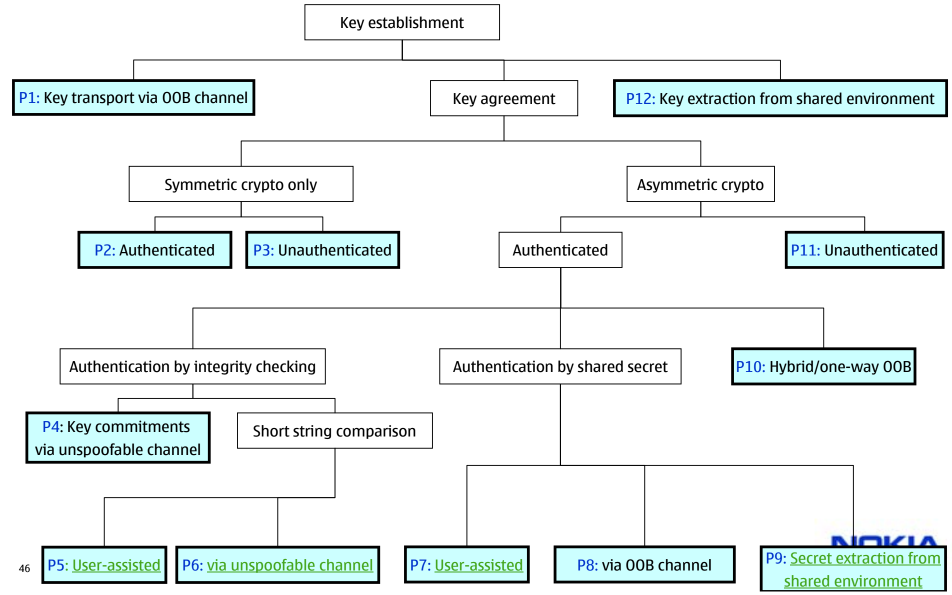
Key establishment protocols for first connect (3)



Key establishment protocols for first connect (4)



Key establishment protocols for first connect (5)



Proposed solutions: emerging standards

Emerging standards for first connect

- **Bluetooth Secure Simple Pairing** (released July 2007)
  - "Just works", 2-way NFC, Comparison of short check strings, 6-digit passkey (20 rounds), NFC tags
    - P11, P4, P5, P7, P10
- **WiFi Alliance Protected Setup** (released January 2007)
  - Flash drives, "Push button", 2-way NFC, short passkey (2 rounds), NFC tags
    - P1, P11, P4, P7, P10
  - Also Windows Connect Now: P1, P7 (released Summer 2006)
- **Wireless USB Association Models** (released early 2006)
  - USB cable, Comparison of short check strings
    - P1, P5
- Others in the works...

Suomalainen, Valkonen, Asokan [ESAS 2007]

Key establishment in Bluetooth pairing

- Key establishment is based on symmetric-key algorithms
- Authentication of key establishment based on a PIN
  - usually short, for usability
- All input to key establishment except PIN is visible to passive eavesdroppers
- When short PINs are used, passive attacker can mount a dictionary attack
  - Can recover PINs, encryption and authentication keys: 4 digit PINs in a few seconds
  - Needs to record messages exchanged using pairing
  - But an active attacker can force re-pairing

Bluetooth Secure Simple pairing

Bluetooth v2.1 *Secure Simple Pairing*

- Objectives
  - Make pairing easier for the end user
  - Improve its security
- Security goals
  - Strong security against passive attackers
  - Good-enough security against active attackers

Easier device discovery

Bluetooth v2.1 *Secure Simple Pairing*

- Out-of-band
  - E.g., BT device addresses exchanged via NFC
  - No need for Bluetooth Inquiry
- User conditioning
  - Devices participate in pairing only in response to user action

Protection mechanisms

Bluetooth v2.1 *Secure Simple Pairing*

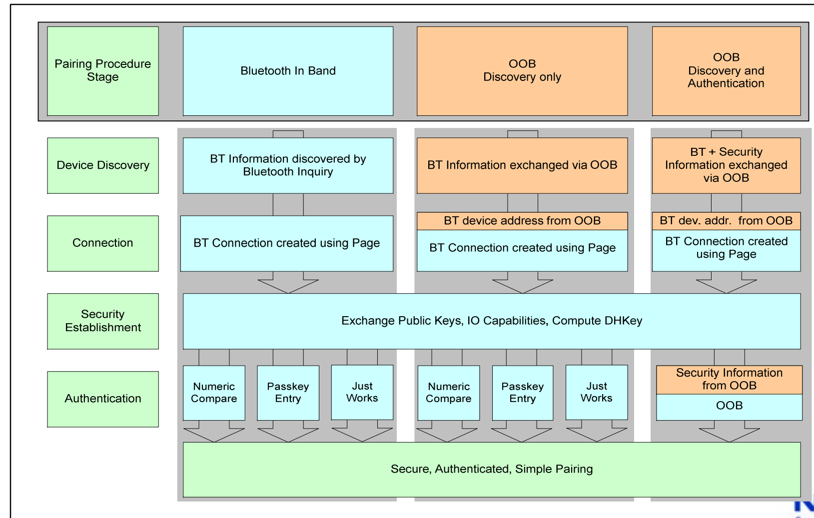
- Passive eavesdroppers: Diffie-Hellman key agreement
- Active attackers: Authentication of key agreement
  - Multiple options for authenticating: “association models”

Association Models (1/2)

- Out-of-band channel
  - User "touches" one device or its tag with another
  - Commitments to public keys and secret passkeys exchanged via out-of-band
- Numeric comparison
  - User compares 6-digit numbers displayed by each device
  - indicates if they are the same or not
- Passkey entry
  - One device shows a 6-digit number; user types it into the other device
- "Just Works"
  - No authentication (but still secure against passive attackers)

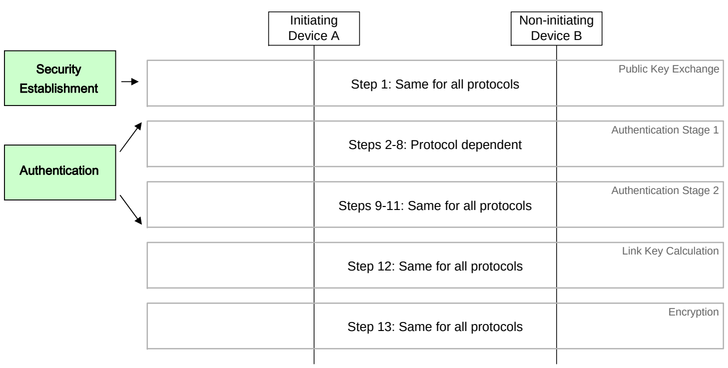
- Choice of model depends on I/O capabilities of devices

Association Models (2/2)

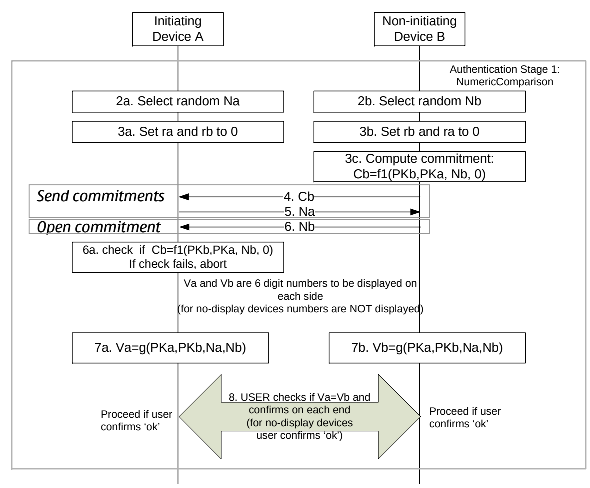


BT: Bluetooth

Phases in Secure Simple Pairing



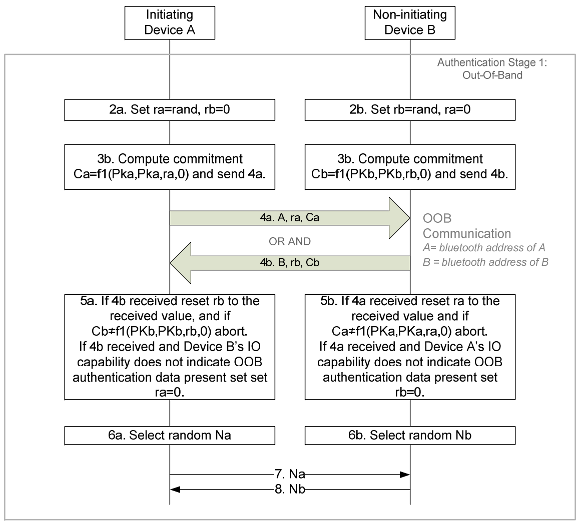
Stage 1 Protocol for numeric comparison



- Main idea
  - A must choose Na before knowing Nb
  - B must choose Nb before knowing Na
  - Attacker cannot control any input to g()
  - Based on MANA IV (6-digit checksum)
- Active attacker has 2^{-20} chance of succeeding
  - Not dependent on his computational resources

Stage 1 Protocol for out-of-band authentication

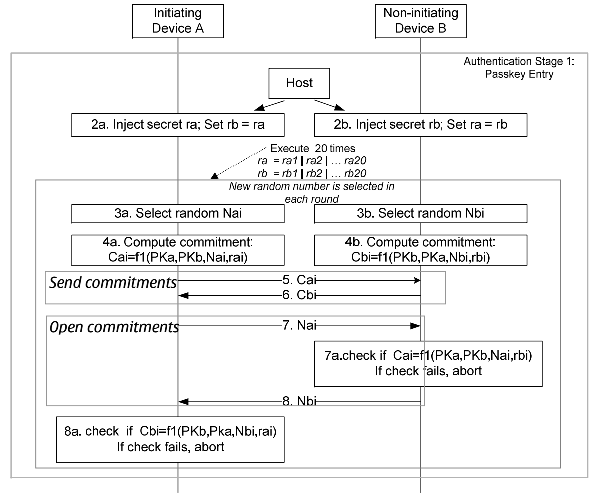
Bluetooth® v2.1 *Secure Simple Pairing*



- If OOB communication is 2-way, authentication takes place in steps 5a and 5b
- If OOB communication is one-way, one direction of authentication postponed to stage 2

Stage 1 Protocol for passkey entry

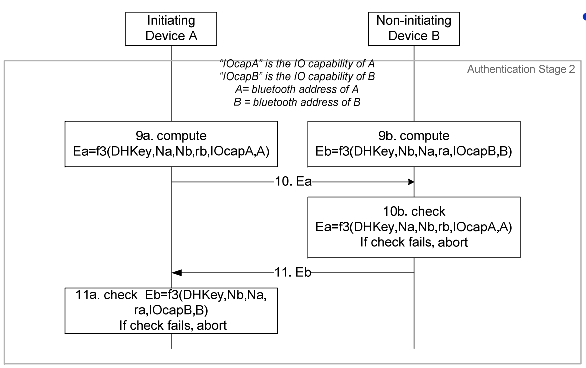
Bluetooth® v2.1 *Secure Simple Pairing*



- Main idea
  - In each round, each party demonstrates knowledge of 1-bit of secret passkey
  - Based on multi-round MANA III
  - 6 digit passkey, 20 rounds
- Active attacker has 2^{-19} chance of succeeding
  - 50% chance of getting each bit right
  - Not dependent on his computational resources
- Passkey must **not be reused**

Stage 2 Protocol

Bluetooth® v2.1 *Secure Simple Pairing*



- Primarily for key confirmation
- When OOB is 1-way, Ea (or Eb) serves as proof-of-knowledge of secret rb (or ra)

OOB Capability Mapping to Authentication Stage 1

Bluetooth® v2.1 *Secure Simple Pairing*

Device A / Device B	Has not received remote OOB authentication data	Has received remote OOB authentication table
Has not received remote OOB authentication data	Use the IO capability mapping table	Use OOB association with ra = 0, rb from OOB
Has received remote OOB authentication data	Use OOB association with ra from OOB, rb = 0	Use OOB association with ra from OOB, rb from OOB

Mapping I/O capabilities to association models

Initiator A / B Responder	DisplayOnly	DisplayYesNo	KeyboardOnly	NoInputNoOutput
DisplayOnly	Numeric Comparison with automatic confirmation on both devices.	Numeric Comparison with automatic confirmation on device B only.	Passkey Entry: Responder Display, Initiator Input.	Numeric Comparison with automatic confirmation on both devices.
DisplayYesNo	Numeric Comparison with automatic confirmation on device A only.	Numeric Comparison: Both Display, Both Confirm.	Passkey Entry: Responder Display, Initiator Input.	Numeric Comparison with automatic confirmation on device A only.
KeyboardOnly	Passkey Entry: Initiator Display, Responder Input.	Passkey Entry: Initiator Display, Responder Input.	Passkey Entry: Initiator and Responder Input	Numeric Comparison with automatic confirmation on both devices.
NoInputNoOutput	Numeric Comparison with automatic confirmation on both devices.	Numeric Comparison with automatic confirmation on device B only.	Numeric Comparison with automatic confirmation on both devices.	Numeric Comparison with automatic confirmation on both devices.

Authenticated

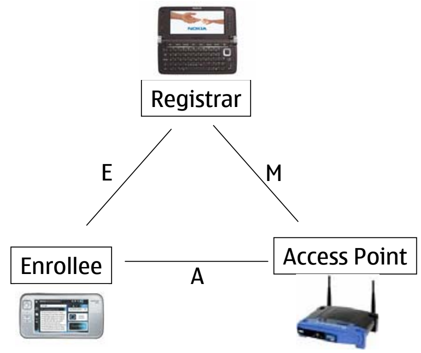
Cryptographic algorithms in Simple Pairing

- Key agreement uses elliptic curve Diffie-Hellman
  - FIPS P-192 curve
    - Security level thought to be comparable to 1024-bit RSA or 80-bit symmetric key algorithms
  - Reasons for choosing ECDH over DH in MODP groups
    - Message sizes are smaller
    - Time, memory use, and code footprint are comparable or better
    - Actual performance figures depends on platform. See http://www.cacr.math.uwaterloo.ca/conferences/2005/ecc2005/vanstone.pdf for some sample figures
- SHA256 is the building block for commitment and MAC functions
  - f1(), f2(), f3() are HMAC-SHA256 truncated to 128 bits (MSBs)
  - g() is SHA-256 truncated to 32 bits (LSB); encoded as 6 digits

Summary

- Bluetooth Simple Pairing is intended to improve usability and security
  - Easier device discovery
  - Strong security against passive eavesdroppers (EC DH key agreement)
  - Good enough (1-in-a-million success probability) security against active attackers
  - Part of Bluetooth 2.1 specification (July 2007)

WiFi Protected Setup (WPS)



- Registrar is the controller of the WiFi network
- Enrollee and Registrar perform key agreement
- Three types of authentication for key agreement
  - "Push Button": Unauthenticated
  - Device Password
  - Out-of-band: Flash drive or NFC
- Resulting key is used for
  - Transporting the actual WLAN key ("ConfigData" in next slides)
  - Long "device password" for future device management

WPS Registration Protocol

6.2. Registration Protocol Messages

Enrollee → Registrar: $M_1 =$ Version || N1 || Description || PK_E

Enrollee ← Registrar: $M_2 =$ Version || N1 || N2 || Description || PK_R [|| ConfigData] || $HMAC_{AuthKey}(M_1 \,||\, M_2^*)$

Enrollee → Registrar: $M_3 =$ Version || N2 || E-Hash1 || E-Hash2 || $HMAC_{AuthKey}(M_2 \,||\, M_3^*)$

Enrollee ← Registrar: $M_4 =$ Version || N1 || R-Hash1 || R-Hash2 || $ENC_{KeyWrapKey}$(R-S1) || $HMAC_{AuthKey}(M_3 \,||\, M_4^*)$

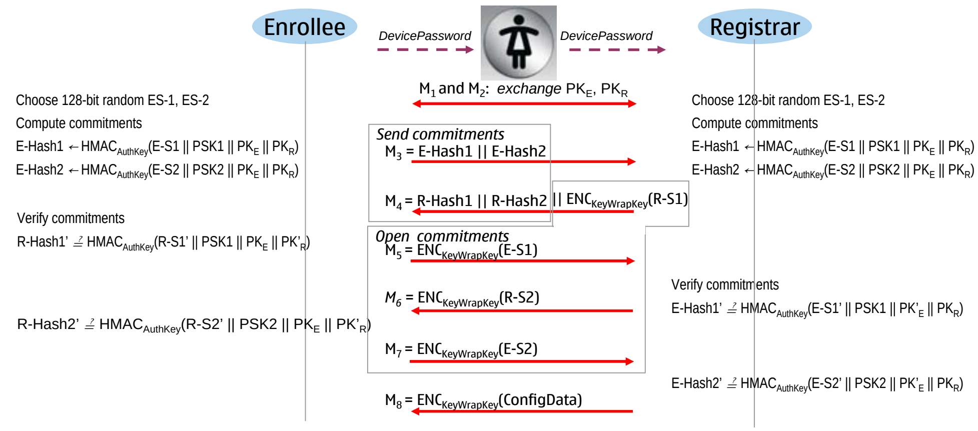
Enrollee → Registrar: $M_5 =$ Version || N2 || $ENC_{KeyWrapKey}$(E-S1) || $HMAC_{AuthKey}(M_4 \,||\, M_5^*)$

Enrollee ← Registrar: $M_6 =$ Version || N1 || $ENC_{KeyWrapKey}$(R-S2) || $HMAC_{AuthKey}(M_5 \,||\, M_6^*)$

Enrollee → Registrar: $M_7 =$ Version || N2|| $ENC_{KeyWrapKey}$(E-S2 [||ConfigData]) || $HMAC_{AuthKey}(M_6 \,||\, M_7^*)$

Enrollee ← Registrar: $M_8 =$ Version || N1 || [$ENC_{KeyWrapKey}$(ConfigData)] || $HMAC_{AuthKey}(M_7 \,||\, M_8^*)$

WPS Registration Protocol: the essentials

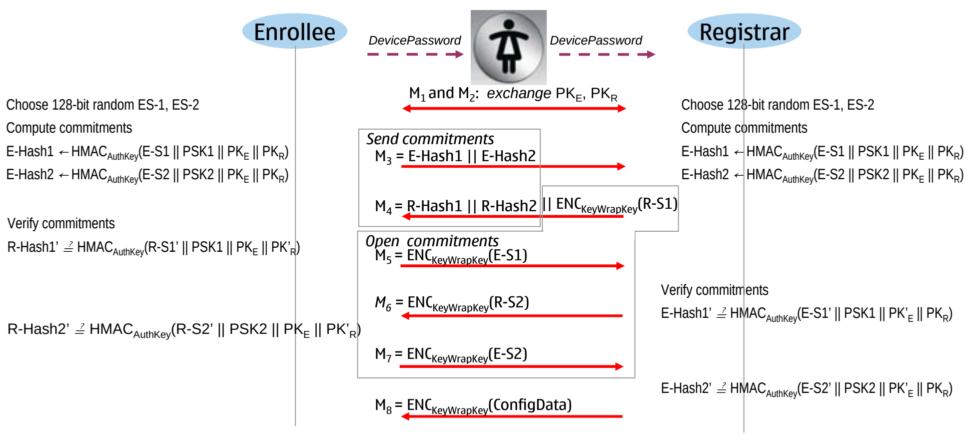


PSKi first 128 bits of HMAC-SHA-256$_{AuthKey}$(ith half of *DevicePassword*)

AuthKey and KeyWrapKey are derived from the Diffie-Hellman key

Based on multi-round MANA III (4- or 8-digit password, 2 rounds)

WPS Registration Protocol: the essentials



PSKi first 128 bits of HMAC-SHA-256$_{AuthKey}$(ith half of *DevicePassword*)

AuthKey and KeyWrapKey are derived from the Diffie-Hellman key

Based on multi-round MANA III (4- or 8-digit password, 2 rounds)

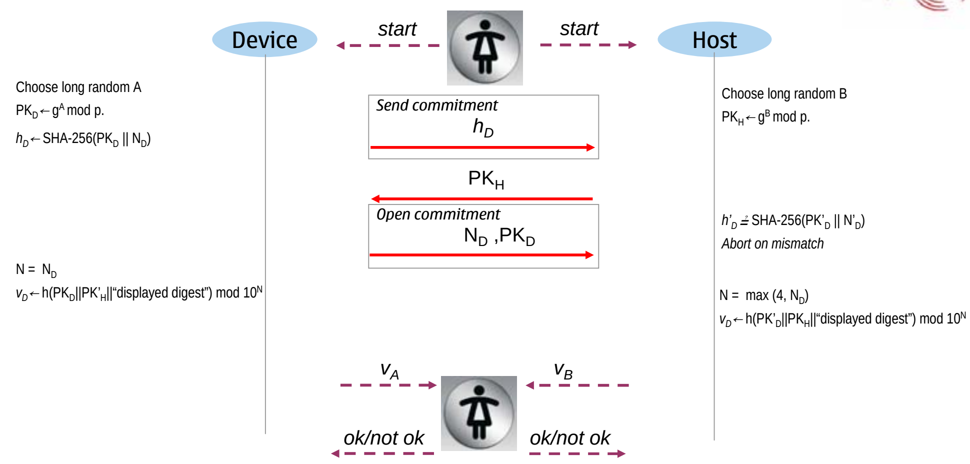
Cryptographic algorithms in WiFi Protected Setup

- Key agreement uses Diffie-Hellman
  - 1536-bit MODP group 5 from RFC 3526
- SHA-256 is used as the building block for key derivation, commitment and message authentication functions
  - Encryption keys are 128 bits
- AES in CBC mode is used for Key wrapping

Wireless USB Association Models

- Wireless USB connection between USB hosts and USB devices
- Two association models supported
  - Cable model
  - Numeric model

WUSB Numeric Association Model: the essentials

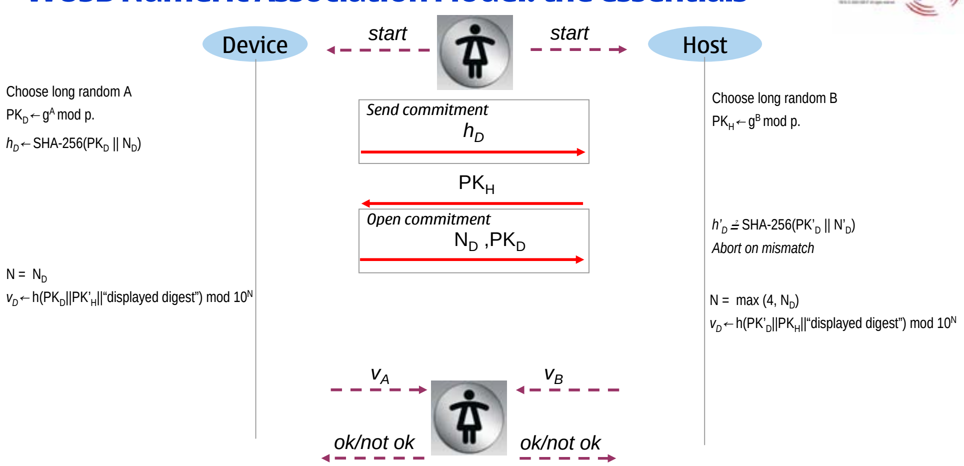


User approves acceptance if v_A and v_B match

h(): first 32-bits SHA-256() output mod 10^N

Similar to MANA IV with 2-4 digit checksums, but key pairs cannot be reused

WUSB Numeric Association Model: the essentials



User approves acceptance if v_A and v_B match

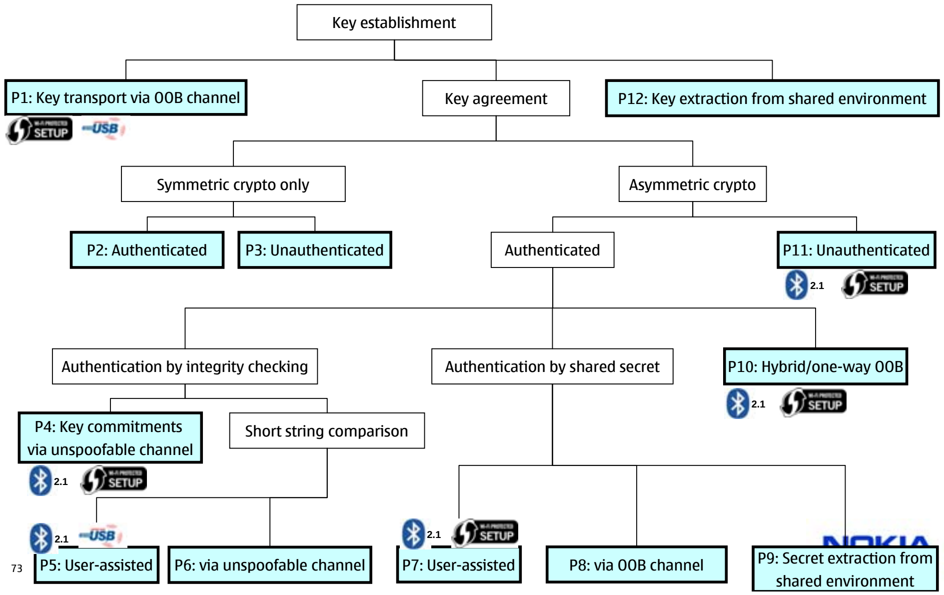
h(): first 32-bits SHA-256() output mod 10^N

Similar to MANA IV with 2-4 digit checksums, but key pairs cannot be reused

Cryptographic algorithms in WUSB Association Models

- Key agreement uses Diffie-Hellman
  - 3072-bit MODP group 15 from RFC 3526
- SHA-256 is used for commitments
  - Encryption keys are 128 bits
- AES in CBC mode is used for Key wrapping

Key establishment protocols for first connect



Comparison of security levels

Association Model	Offline attacks		Online active attacks			
	Protection	Work	Protection	Success Probability	Protection	Work*
Bluetooth Simple Pairing						
Numeric Comparison	DH P-192	2^{80}	6-digit checksum	2^{-20}	128b nonce	2^{128}
Passkey	DH P-192	2^{80}	6-digit passkey, 20 rounds	2^{-19}	128b nonce	2^{128}
"Just Works"	DH P-192	2^{80}	none	1		0
Out-of-band	DH P-192	2^{80}	OOB	-	128b nonce	2^{128}
WiFi Protected Setup						
Out-of-band	OOB + DH Gr. 5 - 1536	2^{90}	OOB	-	128b nonce + 64-bit key	2^{196}
In-band	DH Gr. 5 - 1536	2^{90}	8-digit passkey, 2 rounds	$2^{-13.2}$	128b nonce + 4-digit key	$2^{141.2}$
Push Button	DH Gr. 5 - 1536	2^{90}		1	-	0
Wireless USB Association Models						
Numeric	DH Gr. 15 - 3072	2^{128}	2- or 4-digit checksum	$2^{-6.6}$ or $2^{-13.2}$	256b nonce	2^{256}
Cable	OOB		OOB	-	-	-

* Average work needed to find the right pre-image (with probability 1)

Suomalainen, Valkonen, Asokan [ESAS 2007]

Towards analyzing usability

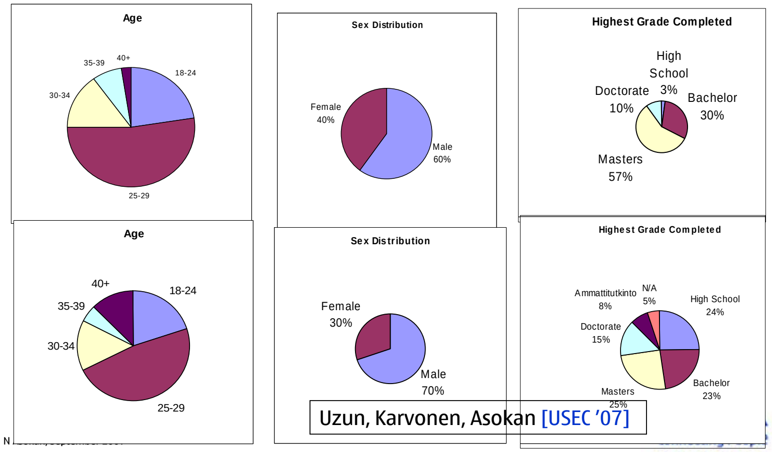
Comparative usability testing (preliminary)

- Comparing **short non-secret** check codes (P5)
  - Compare-and-Confirm, Select-and-Confirm, Copy-and-Confirm
- Using a **short secret** Passkey (P7)
  - Copy (a passkey from one device to another), Choose-and-enter (your passkey to bothe devices)
- Distinguish between "safe" and "fatal" user errors
  - Fatal errors lead to violation of a security objective
- Quantitative measurements and subjective feedback

Uzun, Karvonen, Asokan [USEC '07]

Who Tested the protocols

- Two groups of forty people



Uzun, Karvonen, Asokan [USEC '07]

Copy Passkey

- User copies passkey from one device to the other
  - 4- 8- and, 6-digit passkeys
  - No fatal error possibility
- Results
  - Users opinion: hard to use, professional, preferred
  - 6-digit passkey: takes around 15 seconds
  - 3% safe error rate

Uzun, Karvonen, Asokan [USEC '07]

Compare-and-Confirm (1/2)

- Each device shows a short code and the user is asked to compare the shown values.
- Round 1
  - Näive implementation: Yes/No question
  - Takes around 15 seconds.
  - **85% found it easiest** but only 10% found it professional ☺
  - **20% fatal error** rate: pressing yes without reading instructions!

Uzun, Karvonen, Asokan [USEC '07]

Compare-and-Confirm (2/2)

- Lessons from Round 1
  - "Safe default" [Saltzer and Schroeder]
  - Use of unfamiliar labels
- Round 2
  - Takes around 17 seconds
  - **Only 40% found it the easiest**
  - No fatal errors, 2.5% safe error rate

Uzun, Karvonen, Asokan [USEC '07]

User testing: observations and next steps

- User perception vs. reality
    - Ease-of-use, security
- "Too easy" is not always good?
- Use of unfamiliar labels vs. learning effects
- Fatal errors vs. safe errors
    - Reducing safe errors is important, too
- More controlled testing
- Testing in: familiar environments, repeated attempts, task-oriented
- Other interaction methods

NOKIA Connecting People

Outlook for the future

- Need to revisit Secure First Connect?
    - Unauthenticated key agreement may be the winner: cost and usability
    - But some scenarios would require authentication: input devices, medical devices?
    - "Wanted: inexpensive, intuitive, secure techniques for first connect"?
- Extending First Connect
    - Beyond security associations
        - How can users easily specify access control policies?
    - Group first connect

NOKIA Connecting People

Summary

- Secure first connect is currently difficult
- Standards are emerging but the jury is still out

- Need to balance security, usability *and* **cost**
- Usable security is more than just nice UIs
    - May call for new protocols, algorithms and system design

NOKIA Connecting People

Acknowledgements

Thanks to the folks who helped make some of the slides in this set,

- Kari Kostiainen, Nitesh Saxena, Ersin Uzun

to those whose provided valuable feedback,

- Silke Holtmanns, Seamus Moloney, Kaisa Nyberg, John Solis

and to those students and colleagues who collaborated in some of the research presented.

- Jan-Erik Ekberg, Philip Ginzboorg, Kristiina Karvonen, Kari Kostiainen, Seamus Moloney, Sven Laur, Kaisa Nyberg, Nitesh Saxena, Jani Suomalainen, Ersin Uzun, Jukka Valkonen

NOKIA Connecting People

Pointers to some references

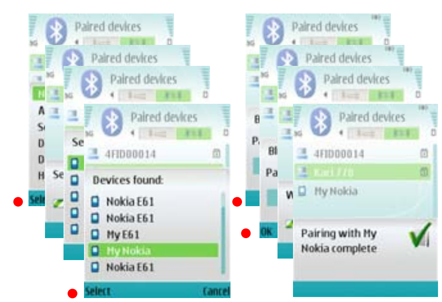
- MANA IV
  - [CANS 2006], LNCS 430,1 pp 90–107, http://dx.doi.org/10.1007/11935070_6
  - [IACR report 2005] http://eprint.iacr.org/2005/424
- Blinking lights (Saxena et al)
  - [IEEE S&P 2006] http://doi.ieeecomputersociety.org/10.1109/SP.2006.35
  - [IACR report 2006] http://eprint.iacr.org/2006/050
- Usability testing
  - [USEC 2007] http://www.usablesecurity.org/papers/uzun.pdf
  - [NRC report 2007] http://research.nokia.com/tr/NRC-TR-2007-002.pdf
- Comparative survey of First Connect standards
  - [ESAS 2007], LNCS 4572, pp 43-57 http://dx.doi.org/10.1007/978-3-540-73275-4_4
  - [NRC report 2007] http://research.nokia.com/tr/NRC-TR-2007-004.pdf
- [Larsson 2001] Jan-Ove Larsson. Higher layer key exchange techniques for Bluetooth security. Open Group Conference, Amsterdam October 24 , 2001.
- [PGPfone1996] http://web.mit.edu/network/pgpfone/manual/#PGP000057

First Connect Standards

- Bluetooth Secure Simple Pairing
  - Part of Bluetooth 2.1 specification: http://www.bluetooth.com/Bluetooth/Learn/Technology/Core_Specification_v21__EDR.htm
- WiFi Protected Setup
  - http://www.wi-fi.org/wifi-protected-setup/
  - Also see, Windows Connect Now-NET: http://www.microsoft.com/whdc/Rally/WCN-Netspec.mspx
- Wireless USB Association Models
  - http://www.usb.org/developers/wusb/

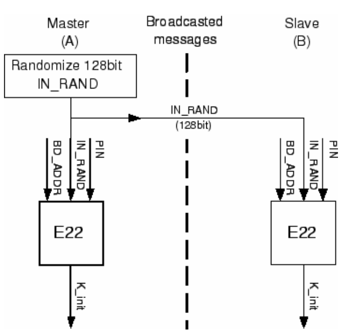
Additional background information

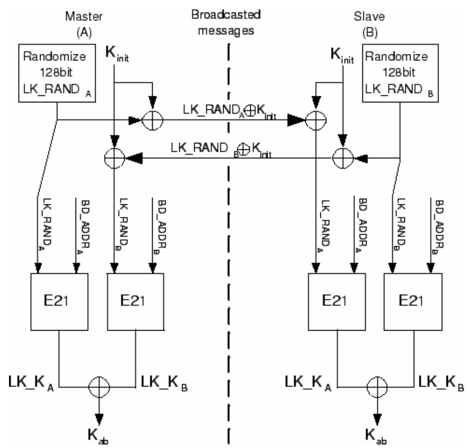
Bluetooth pairing today



Not easy
Not cheap
Not secure

Bluetooth pairing: Link key generation



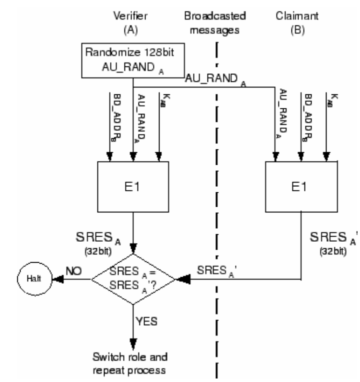


Step 1: Compute K_init

Step 2: Compute K_ab

Shaked and Wool, http://www.eng.tau.ac.il/~yash/shaked-wool-mobisys05/index.html

Bluetooth Mutual Authentication

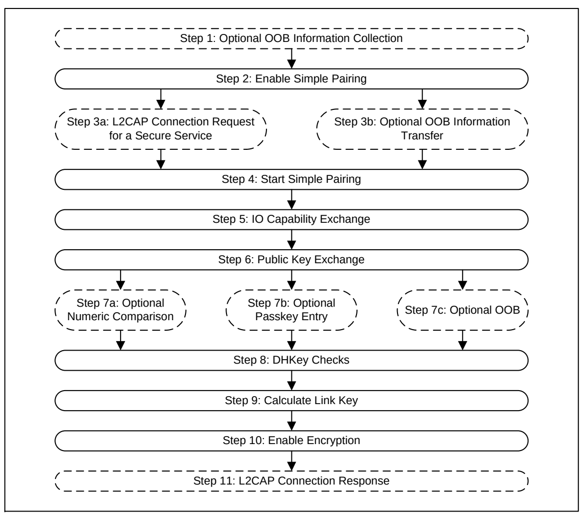


- All information except PIN is available to eavesdropper
- He can test candidate PINs against SRES'$_A$

Shaked and Wool, http://www.eng.tau.ac.il/~yash/shaked-wool-mobisys05/index.html

Secure Simple pairing flow diagram

Bluetooth v2.1 *Secure Simple Pairing*



WFA Protected Setup Registration protocol

Enrollee → Registrar: M_1 = Version || N1 || Description || PK_E

Enrollee ← Registrar: M_2 = Version || N1 || N2 || Description || PK_R [|| ConfigData] || $HMAC_{AuthKey}(M_1 \| M_2^*)$

Enrollee → Registrar: M_3 = Version || N2 || E-Hash1 || E-Hash2 || $HMAC_{AuthKey}(M_2 \| M_3^*)$

Enrollee ← Registrar: M_4 = Version || N1 || R-Hash1 || R-Hash2 || $ENC_{KeyWrapKey}$(R-S1) || $HMAC_{AuthKey}(M_3 \| M_4^*)$

Enrollee → Registrar: M_5 = Version || N2 || $ENC_{KeyWrapKey}$(E-S1) || $HMAC_{AuthKey}(M_4 \| M_5^*)$

Enrollee ← Registrar: M_6 = Version || N1 || $ENC_{KeyWrapKey}$(R-S2) || $HMAC_{AuthKey}(M_5 \| M_6^*)$

Enrollee → Registrar: M_7 = Version || N2 || $ENC_{KeyWrapKey}$(E-S2 [||ConfigData]) || $HMAC_{AuthKey}(M_6 \| M_7^*)$

Enrollee ← Registrar: M_8 = Version || N1 || [$ENC_{KeyWrapKey}$(ConfigData)] || $HMAC_{AuthKey}(M_7 \| M_8^*)$

AuthKey and KeyWrapKey are derived from the Diffie-Hellman key of PKE and PKR

PSK*i* = first 128 bits of $HMAC_{AuthKey}$(*i*th half of DevicePassword)

X-Hash*i* = $HMAC_{AuthKey}$(X-S*i* || PSK*i* || PKE || PKR)