Smart Phones

Presented by: Aleksandr Khasymski

Papers

- A User Study of Policy Creation in a Flexible Access-Control System
  - User study comparing ideal vs. Grey policies

- Seeing-Is-Believing: Using Camera Phones for Human-Verifiable Authentication
  - SiB – a system that uses 2D barcodes and camera phones for authentication

A User Study of Policy Creation in a Flexible Access-Control System

Lujo Bauer

Lorrie Faith Cranor

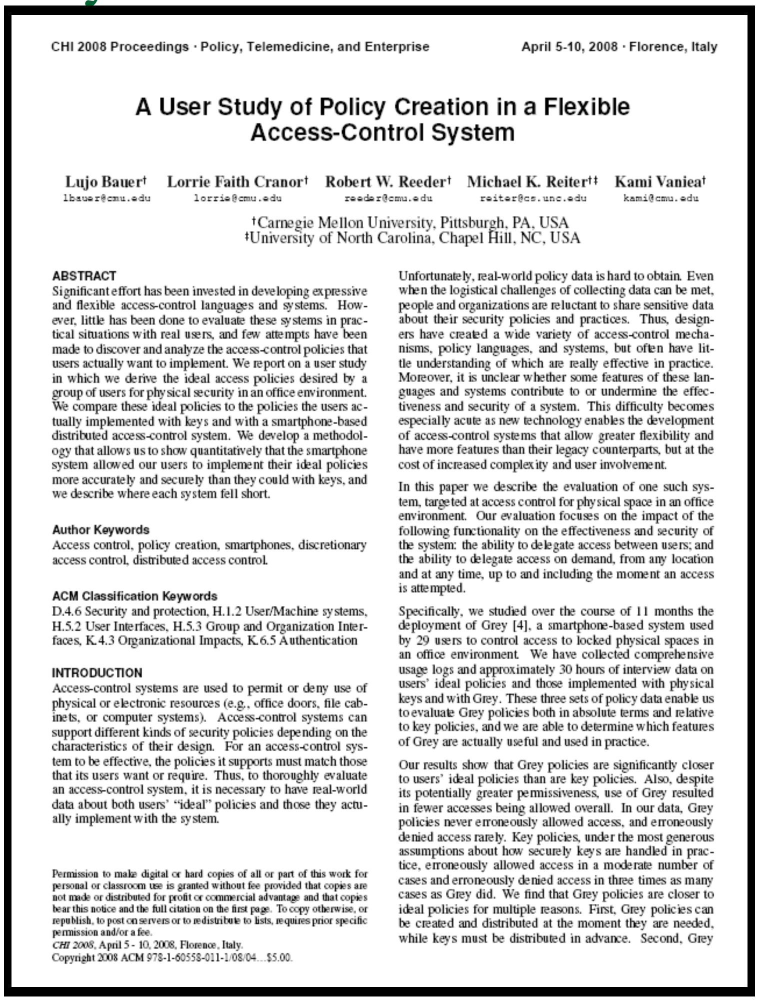
Robert W. Reeder

Michael K. Reiter

Kami Vaniea

Carnegie Mellon University, Pittsburgh, PA, USA

University of North Carolina, Chapel Hill, NC, USA



A User Study of Policy Creation in a Flexible Access-Control System

Lujo Bauer[†] lbauer@cmu.edu · Lorrie Faith Cranor[†] lorrie@cmu.edu · Robert W. Reeder[†] reeder@cmu.edu · Michael K. Reiter[†‡] reiter@cs.unc.edu · Kami Vaniea[†] kami@cmu.edu

[†]Carnegie Mellon University, Pittsburgh, PA, USA
[‡]University of North Carolina, Chapel Hill, NC, USA

ABSTRACT
Significant effort has been invested in developing expressive and flexible access-control languages and systems. However, little has been done to evaluate these systems in practical situations with real users, and few attempts have been made to discover and analyze the access-control policies that users actually want to implement. We report on a user study in which we derive the ideal access policies desired by a group of users for physical security in an office environment. We compare these ideal policies to the policies the users actually implemented with keys and with a smartphone-based distributed access-control system. We develop a methodology that allows us to show quantitatively that the smartphone system allowed our users to implement their ideal policies more accurately and securely than they could with keys, and we describe where each system fell short.

Author Keywords
Access control, policy creation, smartphones, discretionary access control, distributed access control.

ACM Classification Keywords
D.4.6 Security and protection, H.1.2 User/Machine systems, H.5.2 User Interfaces, H.5.3 Group and Organization Interfaces, K.4.3 Organizational Impacts, K.6.5 Authentication

INTRODUCTION
Access-control systems are used to permit or deny use of physical or electronic resources (e.g., office doors, file cabinets, or computer systems). Access-control systems can support different kinds of security policies depending on the characteristics of their design. For an access-control system to be effective, the policies it supports must match those that its users want or require. Thus, to thoroughly evaluate an access-control system, it is necessary to have real-world data about both users' "ideal" policies and those they actually implement with the system.

Unfortunately, real-world policy data is hard to obtain. Even when the logistical challenges of collecting data can be met, people and organizations are reluctant to share sensitive data about their security policies and practices. Thus, designers have created a wide variety of access-control mechanisms, policy languages, and systems, but often have little understanding of which are really effective in practice. Moreover, it is unclear whether some features of these languages and systems contribute to or undermine the effectiveness and security of a system. This difficulty becomes especially acute as new technology enables the development of access-control systems that allow greater flexibility and have more features than their legacy counterparts, but at the cost of increased complexity and user involvement.

In this paper we describe the evaluation of one such system, targeted at access control for physical space in an office environment. Our evaluation focuses on the impact of the following functionality on the effectiveness and security of the system: the ability to delegate access between users; and the ability to delegate access on demand, from any location and at any time, up to and including the moment an access is attempted.

Specifically, we studied over the course of 11 months the deployment of Grey [4], a smartphone-based system used by 29 users to control access to locked physical spaces in an office environment. We have collected comprehensive usage logs and approximately 30 hours of interview data on users' ideal policies and those implemented with physical keys and with Grey. These three sets of policy data enable us to evaluate Grey policies both in absolute terms and relative to key policies, and we are able to determine which features of Grey are actually useful and used in practice.

Our results show that Grey policies are significantly closer to users' ideal policies than are key policies. Also, despite its potentially greater permissiveness, use of Grey resulted in fewer accesses being allowed overall. In our data, Grey policies never erroneously allowed access, and erroneously denied access rarely. Key policies, under the most generous assumptions about how securely keys are handled in practice, erroneously allowed access in a moderate number of cases and erroneously denied access in three times as many cases as Grey did. We find that Grey policies are closer to ideal policies for multiple reasons. First, Grey policies can be created and distributed at the moment they are needed, while keys must be distributed in advance. Second, Grey

Outline

- Introduction/Motivation
- Grey
- Methodology
  - Ideal Policies
  - Physical Key Policies
    - Assumption about hidden keys
  - Grey Policies
- Discussion

Introduction

- A study of an access control system, Grey.
- Almost identical setup as in the user study from Tuesday.
  - Same building.
- Differences:
  - More users, 29.
  - Longer period, 11 Months.
  - Only access to physical resources studied.
  - Focus is on user policy - “ideal” vs. actual

Contributions

- Document a collection of ideal policy data.
- Develop a metric and methodology for **quantitatively comparing** the **accuracy** of implemented policies.
- Present a case study in which a smartphone-based discretionary access-control system **outperforms** keys in **overall security** and **effectiveness** of implementing users' desired policies, and identify the features that account for these improvements.

Grey

- Distribute access-control system.
- Uses off-the-shelf smart phones.
- Smart phones can communicate with computers imbedded in the doors to gain access.
- Owner of a resource can define proactive and reactive policies, e.g. give access on request or proactively grant access.

Methodology

- Environment
  - Office building.
- Users
  - Professors, Students, and Administrative staff.
- Procedure
  - Extensive data logs and user interviews.
  - Initial interview
    - Ideal policy
  - Regular interviews
    - Physical key and Grey policy.

Methodology cont.

- Analysis
  - Access-control policy defined per resource, with a rule for every resource user.
  - 9 resources, 27 users each.
  - Analyzed log data to determine all 244 rules in the Grey policies.
  - Obtained physical key policy from interviews.
  - Determined discrepancies between ideal and actual policies and recorded *false accepts*, and *false rejects*.

Ideal, physical key, and Grey policies

- Ideal policies constructed from interviews
- Physical key and Grey determined from actual practices.

Ideal Access Conditions
I1. True (can access anytime)
I2. Logged
I3. Owner notified
I4. Owner gives real-time approval
I5. Owner gives real-time approval and witness present
I6. Trusted person gives real-time approval and is present
I7. False (no access)

Physical Key Access Conditions
K1. True (has a key)
K2. Ask trusted person with key access
K3. Know location of hidden key
K4. Ask owner who contacts witness
K5. False (no access)

Grey Access Conditions
G1. True (has Grey access)
G2. Ask trusted person with Grey access
G3. Ask owner via Grey
G4. Ask owner who contacts witness
G5. False (no access)

Physical Key Policies

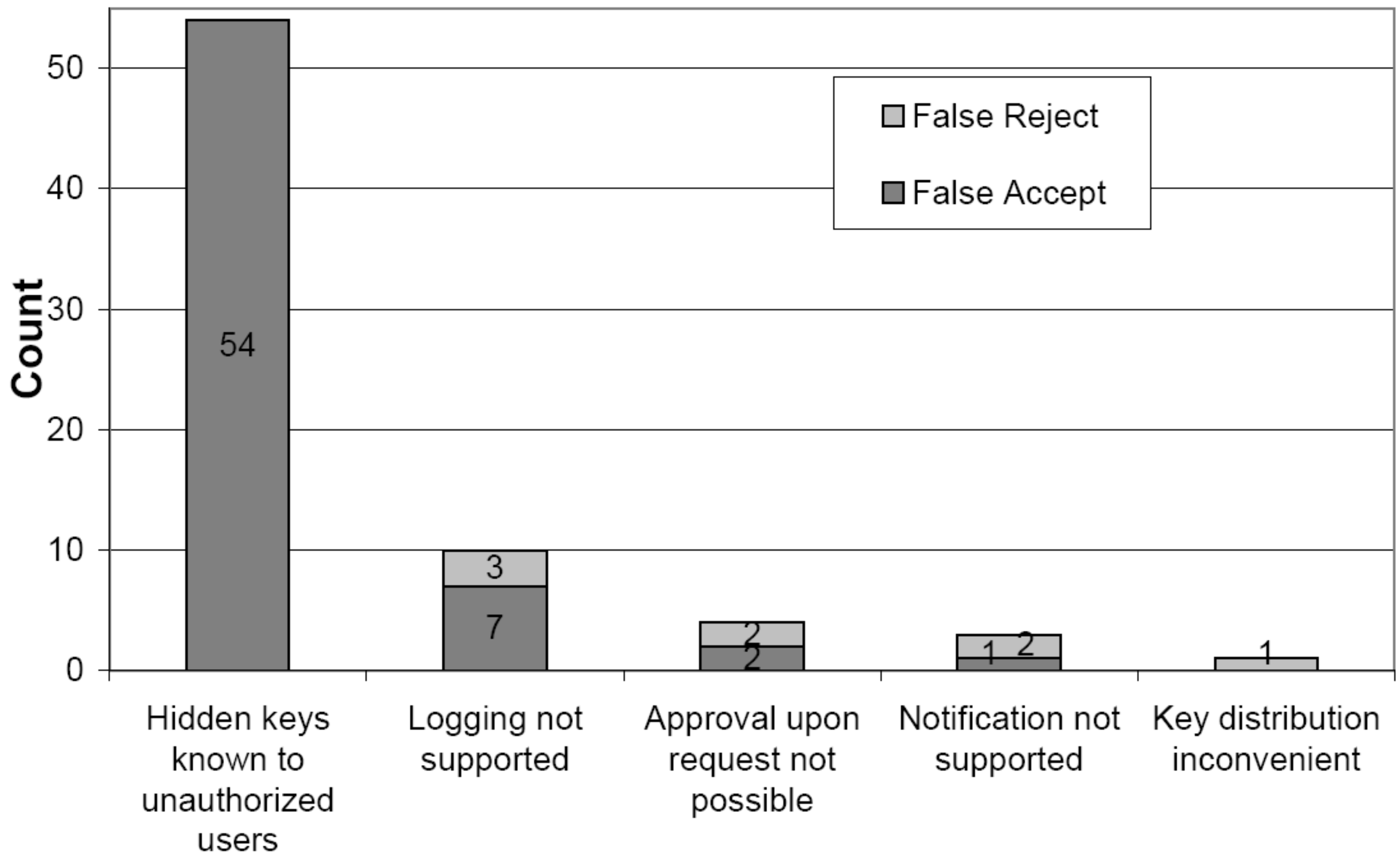
- Causes of discrepancies
  - Hidden keys were available to unauthorized users.
  - Logging (I2) was not supported.
  - Notification (I3) was not supported.
  - Approval upon request (I4) when the owner is not physically present at the resource was not possible.
  - Key distribution was inconvenient.

Hidden Key Assumption

- Optimistic assumption
  - Users will respect the key policy
- Moderate assumption
  - Users will use any hidden key located in a space to which they have access by the key policy, e.g. cubicle farm.
- Pessimistic assumption
  - Users will use any hidden key, e.g. hidden key in a professor's office.

Hidden keys assumption	*False accepts*	*False rejects*
Optimistic	7	12
Moderate	64	8
Pessimistic	169	3

Hidden Key Assumption cont.

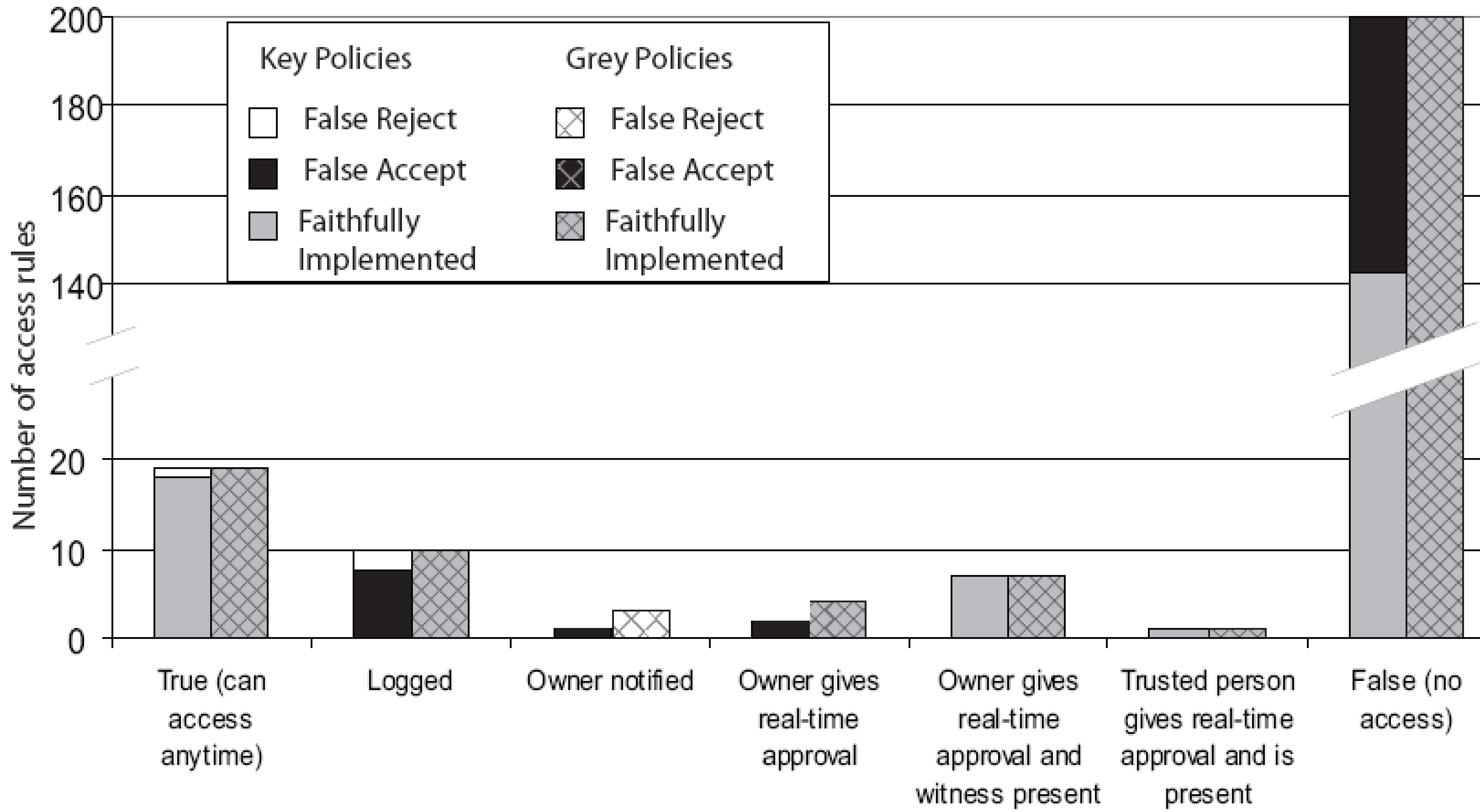


Counts of key policies' false accepts and rejects by cause, under the moderate assumption about knowledge of hidden keys.

Grey Policy

- Closely matched ideal
- Deferred delegation assumption
  - 10 false rejects
- No support for notification
  - 3 false rejects
  - Grey can easily be extended via a services like SMS

Results


Key Policies
False Reject
False Accept
Faithfully Implemented
Grey Policies
False Reject
False Accept
Faithfully Implemented
Number of access rules
200
180
160
140
20
10
0
True (can access anytime)
Logged
Owner notified
Owner gives real-time approval
Owner gives real-time approval and witness present
Trusted person gives real-time approval and is present
False (no access)

Discussion/Conclusion

- “Permissiveness”
  - Easy delegation does not cause excessive permissiveness.
  - Because people can easily manipulate policies the grey policies are more restrictive than the physical key ones, manly due to the hidden keys.
- Transitive delegation
  - Outside the study – some users wanted non-transitive delegation as well, e.g. for “trusted person”.
- Arbitrary grouping granularity
  - This feature of Grey was not explored due to small participant pool.
- Conclusion: Grey policy matches ideal more closely than physical key policy.

Class Discussion

- Is the study setup to succeed?
  - Not clear how ideal policies are derived.
  - How about temporal policies?
  - Non-transitive delegation.
- As the paper points out, the study evaluates the needs of the resource *owner*, which might be different from the ones of the recourse *user.*

Seeing-Is-Believing: Using Camera Phones for Human-Verifiable Authentication

Jonathan M. McCune

Adrian Perrig

Michael K. Reiter

Carnegie Mellon University

Seeing-Is-Believing:
Using Camera Phones for Human-Verifiable Authentication *

Jonathan M. McCune   Adrian Perrig   Michael K. Reiter
Carnegie Mellon University
{jonmccune, perrig, reiter}@cmu.edu

Abstract

Current mechanisms for authenticating communication between devices that share no prior context are inconvenient for ordinary users, without the assistance of a trusted authority. We present and analyze Seeing-Is-Believing, a system that utilizes 2D barcodes and camera-phones to implement a visual channel for authentication and demonstrative identification of devices. We apply this visual channel to several problems in computer security, including authenticated key exchange between devices that share no prior context, establishment of a trusted path for configuration of a TCG-compliant computing platform, and secure device configuration in the context of a smart home.

1. Introduction

Obtaining authenticated values from devices in ways that are easily understandable by non-expert users is currently an open problem. This is best exemplified by the problem a user faces when she wants to securely connect her wireless device to *that* other device (e.g., a network printer, an 802.11 base station, or another wireless device). In general, it is exceedingly difficult to determine which device is at the other end of a wireless connection without out-of-band knowledge. Balfanz et al. describe this as the problem of achieving *demonstrative identification* of communicating devices [4]. We approach this problem with the premise that, in many situations, a user can visually identify the desired device.

As camera-equipped mobile phones rapidly approach ubiquity, these devices become a naturally convenient platform for security applications that can be deployed quickly and easily to millions of users. Today's mobile phones increasingly feature Internet access and come equipped with cameras, high-quality displays, and short-range Bluetooth wireless radios. They are powerful enough to perform secure public key cryptographic operations in under one second.

We propose to use the camera on a mobile phone as a new *visual channel* to achieve demonstrative identification of communicating devices formerly unattainable in an intuitive way. We term this approach Seeing-Is-Believing (SiB). In SiB, one device uses its camera to take a snapshot of a barcode encoding cryptographic material identifying, e.g., the public key of another device. We term this a *visual channel*. Barcodes can be pre-configured and printed on labels attached to devices, or they can be generated on-demand and shown on a device's display.

We apply this visual channel to several problems in computer security. SiB can be used to bootstrap authenticated key exchange between devices that share no prior context, including such devices as mobile phones, wireless access points, and public printers. We use SiB to aid in the establishment of a trusted path for configuration of a TCG-compliant[1] computing platform, and to provide the user with assurance in the integrity of an application running on a TCG-compliant computing platform. We also use SiB to secure device configuration in the context of a smart home.

Outline We survey related work in Section 2 and provide an overview of SiB in Section 3. Section 4 presents the use of SiB for authenticated key exchange between mobile devices. Section 5 explains how to use SiB to achieve demonstrative identification of, and secure connection to, a particular wireless device, with establishment of a

*This research was supported in part by National Science Foundation grant number CNS-0433540, U.S. Army Research Office contract number DAAD19-02-1-0389, and by gifts from Bosch and 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, Bosch, Carnegie Mellon University, Intel, NSF, or the U.S. Government or any of its agencies.

[1]The Trusted Computing Group (TCG) is an organization that promotes open standards to strengthen computing platforms against software-based attacks [2, 3].

Outline

- Introduction/Motivation
- Related Work
- Seeing-Is-Believing (SiB)
  - Diffie-Hellman key exchange
  - Applications
- Implementation
- Security analysis
- Conclusion

Introduction

- How to tackle the problem of authenticating communication between devices?
- Researchers observe that in many cases users can visual identify the device.
- Solution:

  Exploit this secure “visual channel” using camera-equipped mobile phones as a way to “bootstrap” secure communication over an unsecure channel, such as Bluetooth.

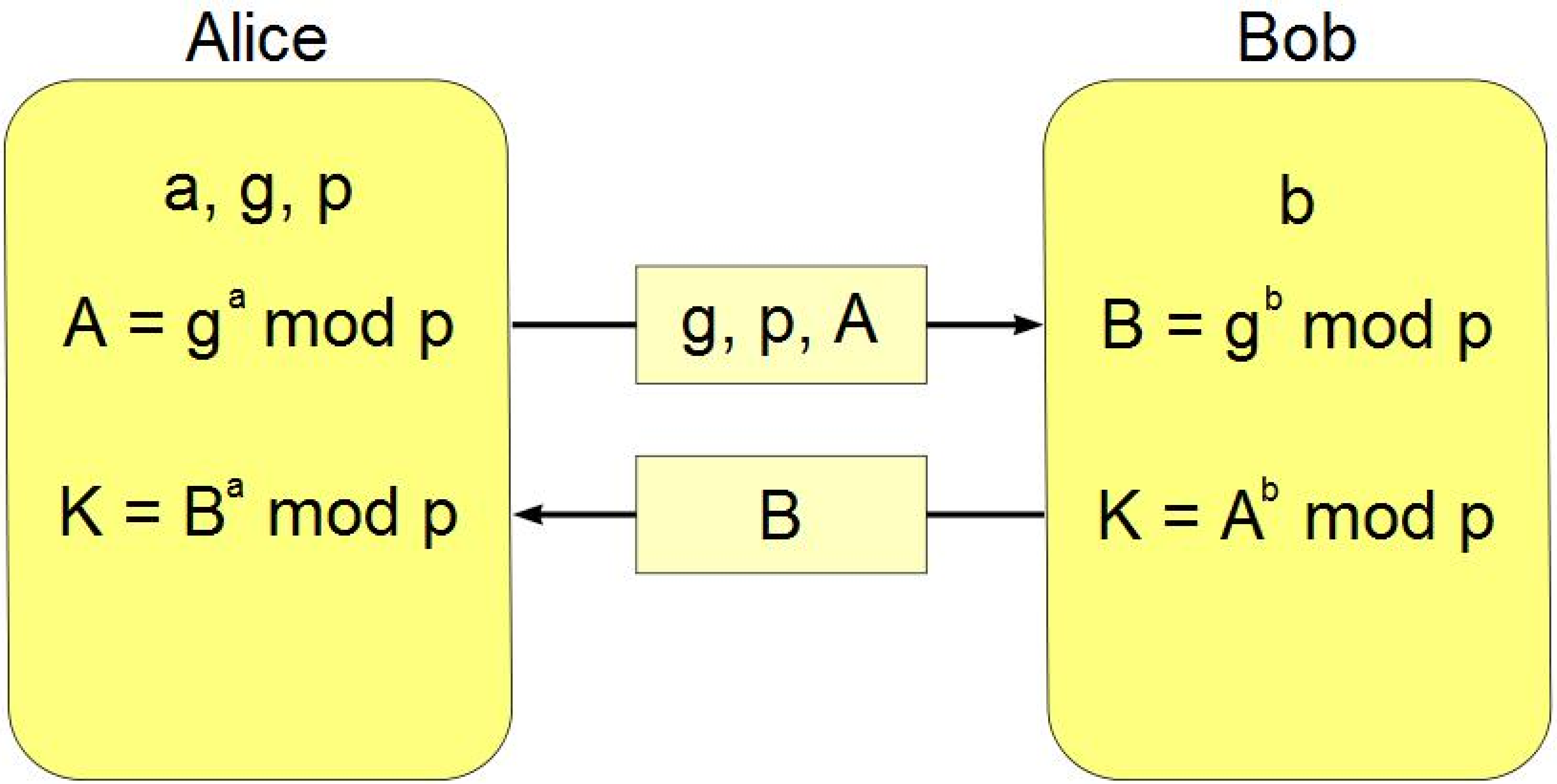
Related Work

- Diffie-Hellman key exchange is a classic mechanism for establishing a secure communication.
- Suffers from Man-in-the-middle attack (MITM)
- Solutions in related work:
  - Pre-established secret password
    - Not practical in devices with limited keyboards
  - Visual metaphors for keys
    - Requires users to manual inspect metaphors
  - Physical contact
    - Cumbersome
- Another solution: SiB!

Seeing-is-Believing (SiB)

- SiB solves the MITM attack.
- Also provides *demonstrative identification* – the user is sure that her device is communicating with *that* device.
- The requirement is that both devices have a camera and can display a 2D barcodes.

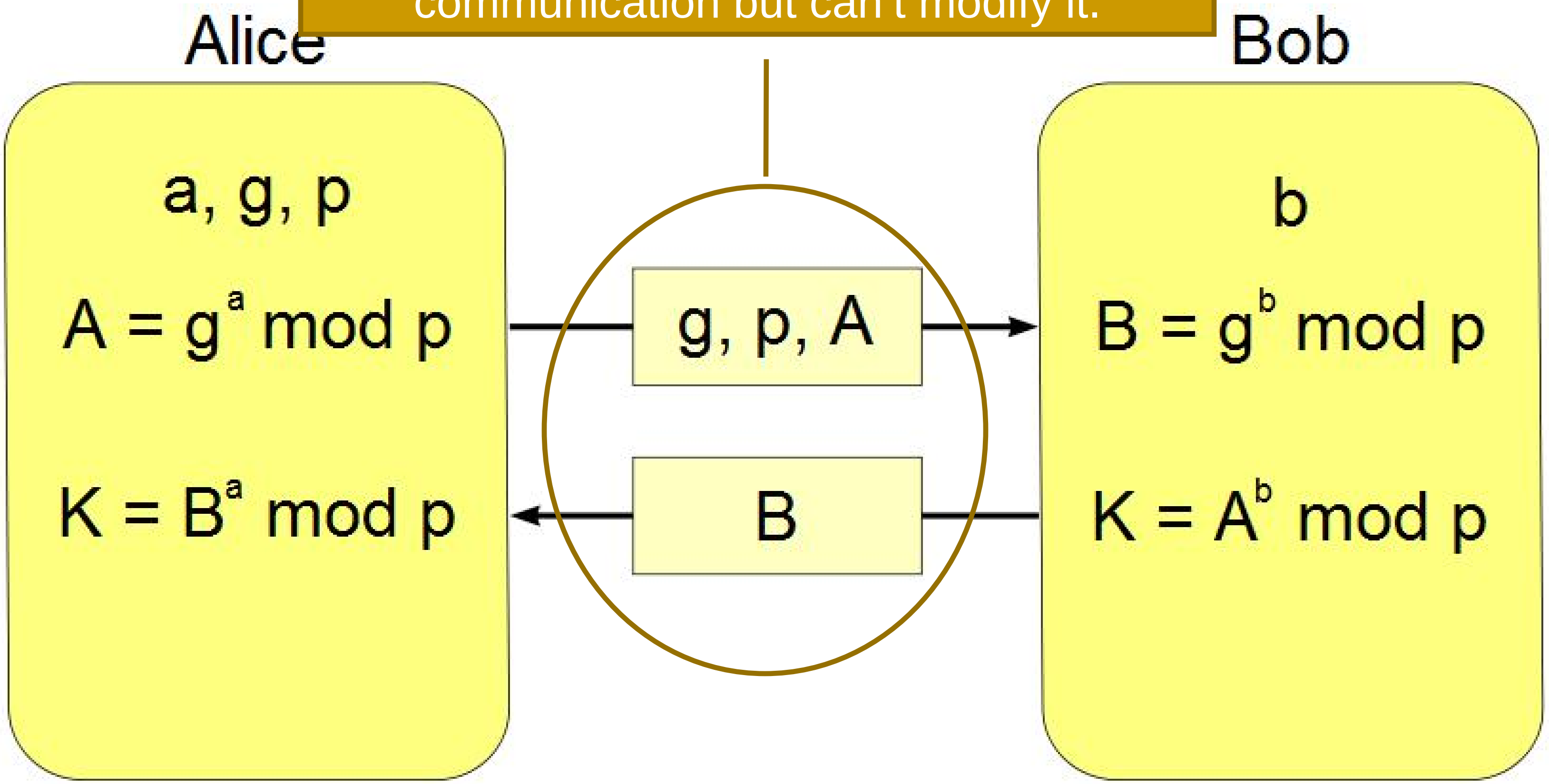
Diffie-Hellman key exchange



$$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$$

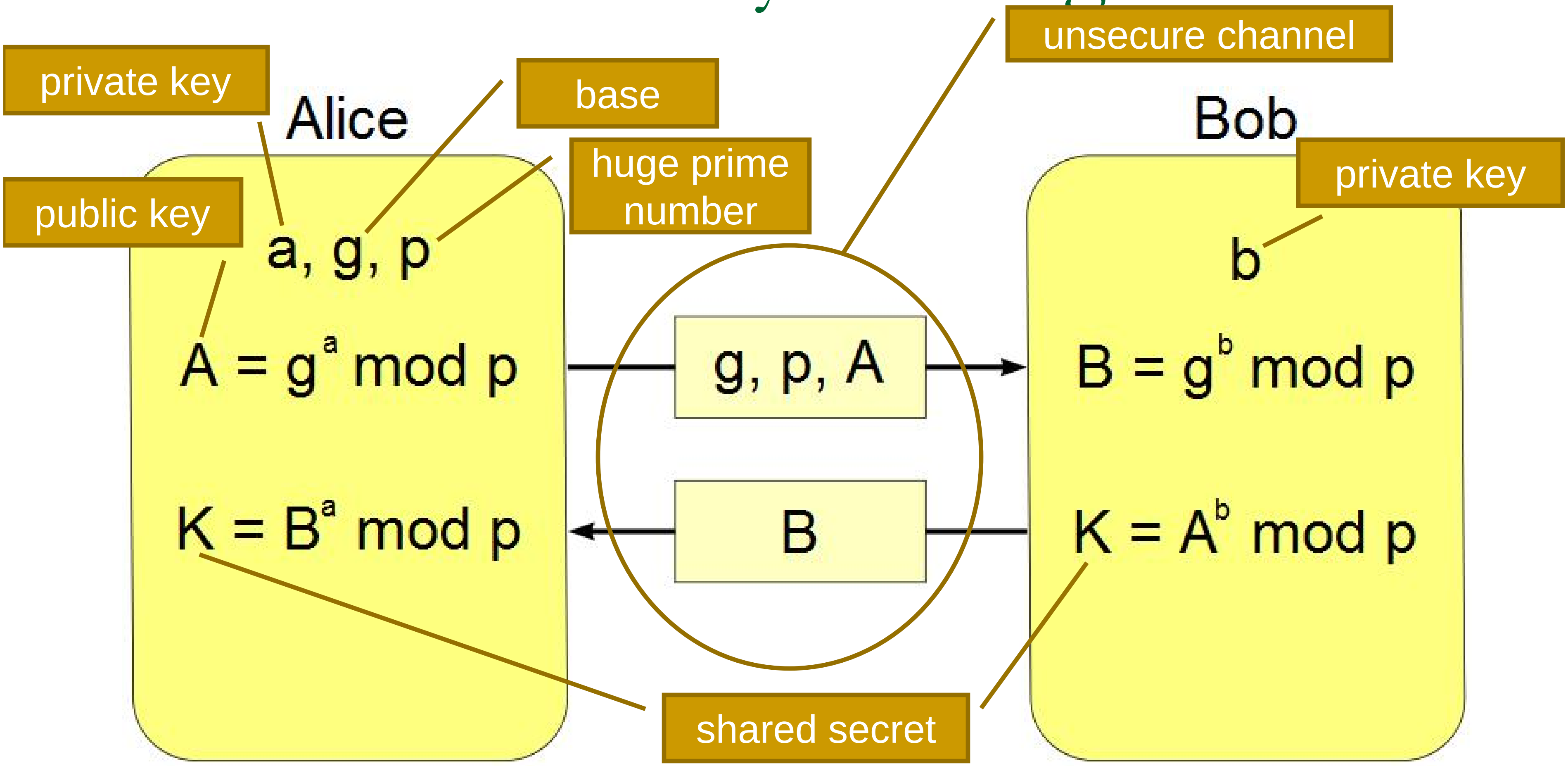
Diffie-Hellman key exchange

Eavesdropper Eve can intercept the communication but can't modify it.



$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$

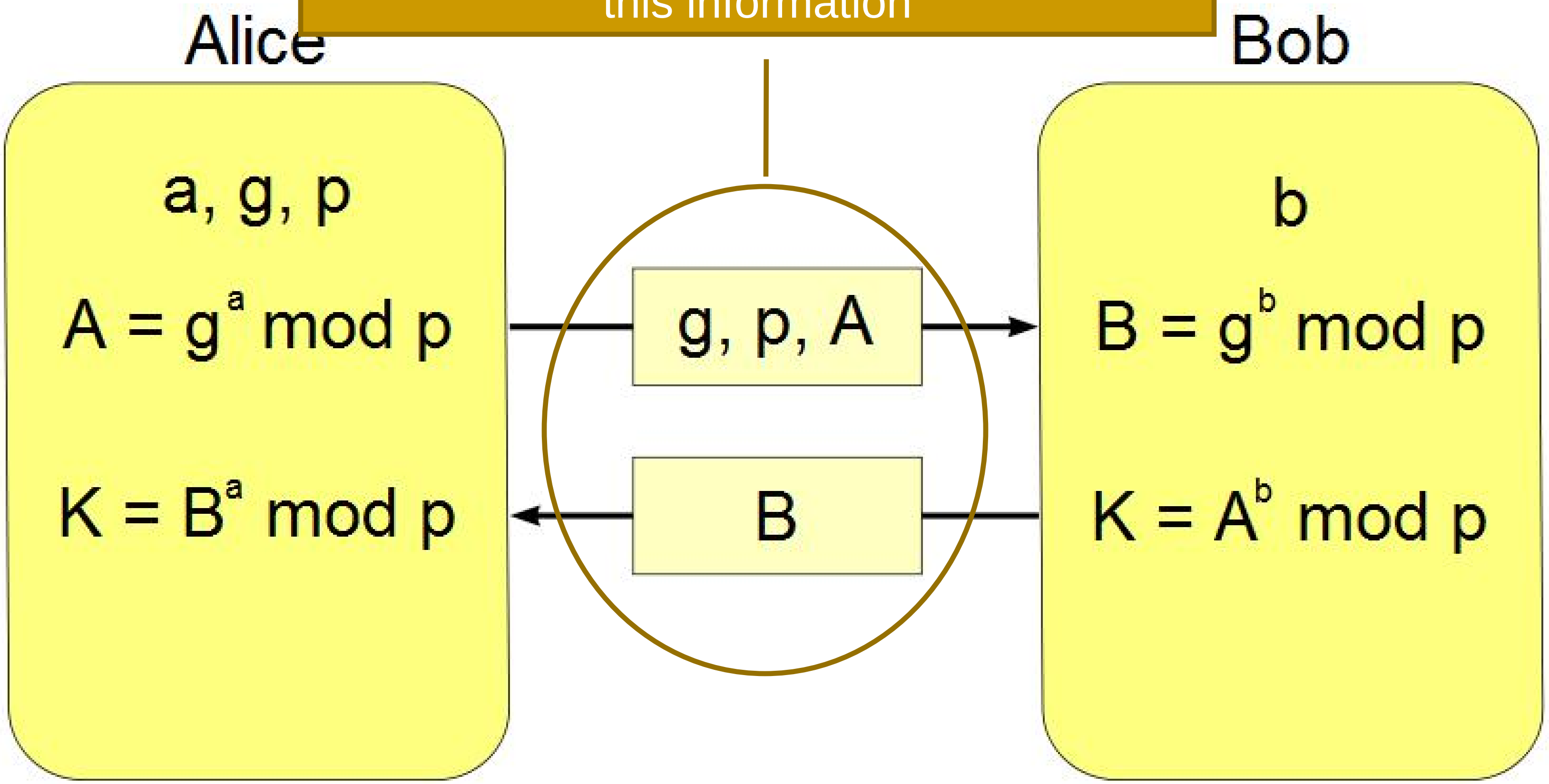
Diffie-Hellman key exchange



$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$

Diffie-Hellman key exchange

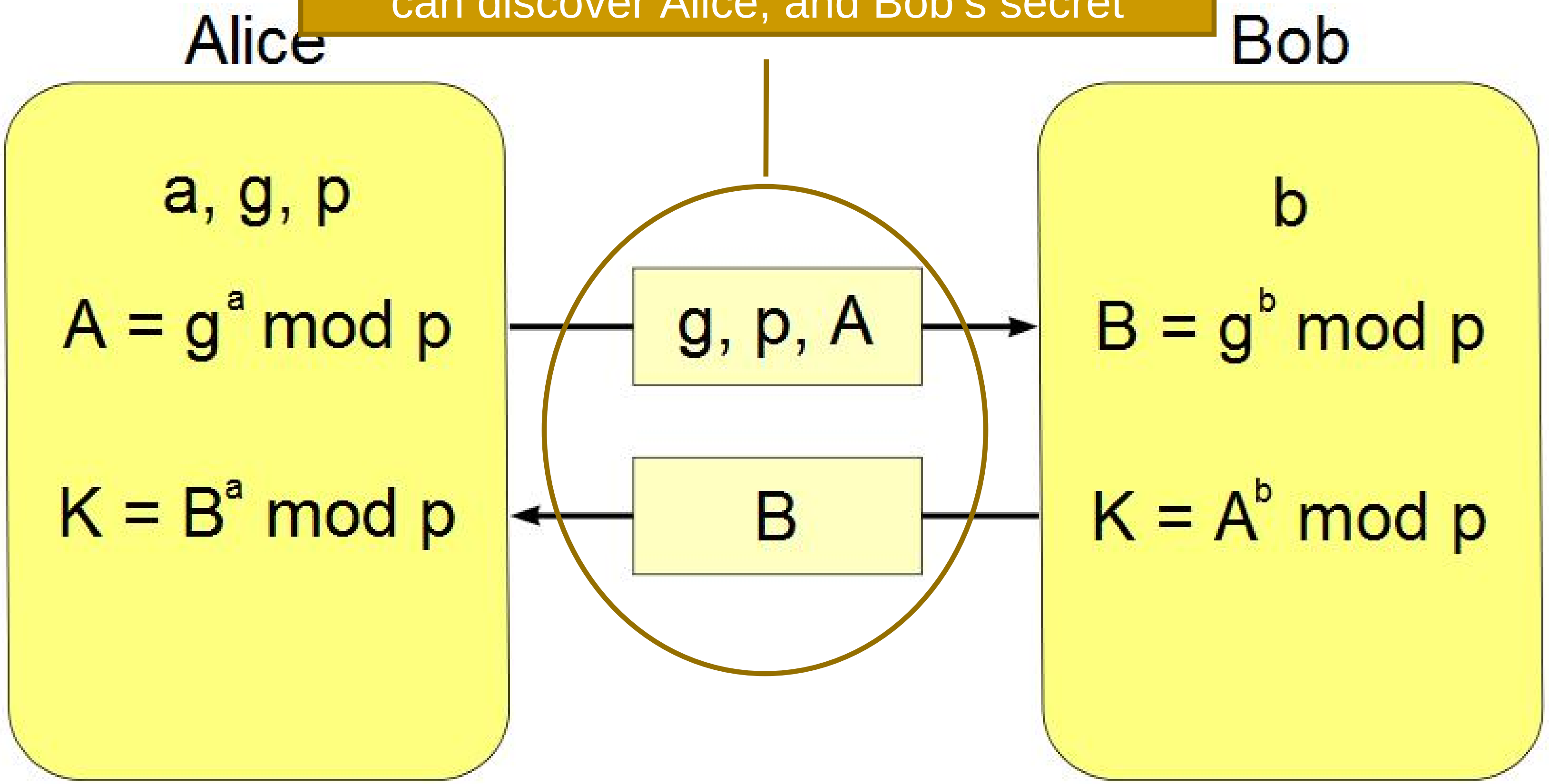
Eavesdropper Eve cannot construct K from this information



$$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$$

Diffie-Hellman key exchange

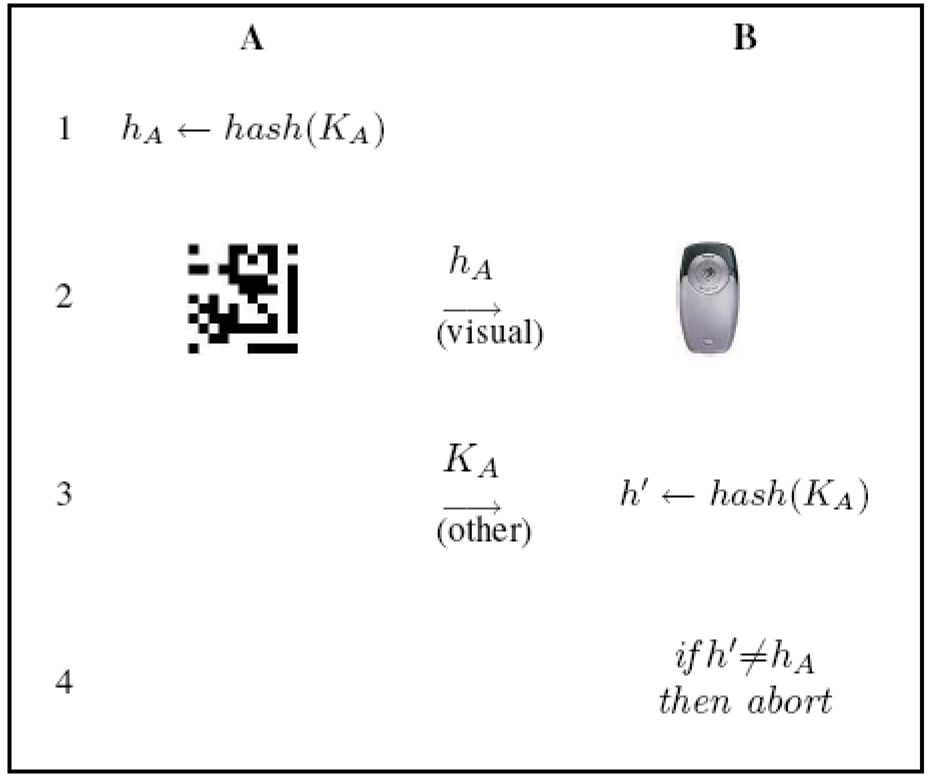
If Eve can tamper with the channel, she can discover Alice, and Bob's secret



$$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$$

Diffie-Hellman key exchange augmented with SiB

- Solution:
  - Compute a hash of the public key
  - Transform hash to 2D barcode
  - Transfer it over secure visual channel
  - Transfer public key over Bluetooth
  - Recompute and compare hashes



Requirements for SiB

- Authentication can be:
  - Unidirectional
  - Biderectional
- Presence
  - Authenticating device is certain of its proximity to the other device
    - Useful in a smart-home

X \ Y	CD	C	D	N
CD	✓	✓*	✓	✓*
C	✓	✓*	✓	✓*
D	presence	presence	×	×
N	×	×	×	×

Legend

✓	Strong authentication possible
✓*	Barcode label required on housing
presence	Confirm presence only
×	No authentication possible

Can device of type X authenticate device of type Y?

Applications of Unidirectional Authenitcation

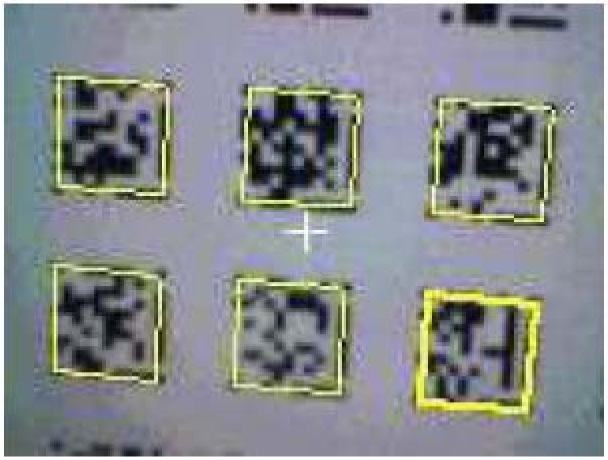
- Sticker
  - Wireless access point
  - Public network printer
- Uses with Trusted Platform Module (TPM) in TCG-compliant computing platform

Application with a TPM

- TPM configured by user or vendor with Owner Authorization Data (OAD), e.g. password
- "Spyware" can log keystrokes and other inputs on a computer.
  - It can capture the password while user enters it.
- Solution:
  - Hash code of the public key is affixed to the computer.
  - OAD is stored on the phone.
  - Transmitted only if TPM's public key is authenticated.

Application with Screen Ownership

- Platform Configuration Registers (PCRs)
  - Can be used to ascertain that particular software configuration is running.
- Solution:
- Initial configuration
  - Generate public/private key pair based on PCRs.
  - Generate barcodes based on the public key and capture them with the camera.
- Subsequent verification
  - Phone presents cryptographic challenge.
  - Application signs it with private key.
  - Only untampered application will display the correct barcodes.
- Requirements (for window manager)
  - Application is "always-on-top".
  - Other application cannot screen capture.



Presence

- Device with no camera can detect the “presence” of another device near it.
- The device displays a barcode.
- Only devices that can “see” the barcode can properly encode data and send it to the authenticating device.
- Useful in the context of a smart home.

Implementation Details

- Run on Nokia 6600 runnig Symbian OS.
- Barcode has Reed-Solomon bits to detect errors in recognition.
- SiB is able to process 2 or 3 barcode snapshots per second.
- Successfully read up to 5 barcodes from a single image for a sustainable rate of 10 to 15 barcodes per second.

Security Analysis

- Small barcodes can be susceptible to brute force attacks.
- Solution:
  - Use multiple barcodes to achieve useful data content of more than 80 bits – industry standard.
  - Use ephemeral Diffie-Hellman keys.
    - Very limited time for the hacker to discover key.

Security Analysis cont.

Channel	COTS	Resists MITM	Convenient
Ultrasound	○	○	●
Audible ("beeps")	○	◐	●
Radio	●	○	●
Physical Contact	○	●	●
Wired Link	●	●	○
Spoken Passwords	N/A	●	○
Written Passwords	N/A	●	○
Visual Hash Verif.	●	●	◐
Infrared	●	◐	◐
Seeing-Is-Believing	●	●	●

Figure 7. Characteristics of various channels proposed for authentication. We acknowledge that rating the convenience of a channel is subjective; however, we believe it is useful to compare various channels in this way. COTS indicates that the necessary hardware is already present in Commercial Off-The-Shelf products. Symbols: yes (●), partial (◐), no (○).

Conclusion

- SiB achieve human identifiable authentication between two devices
  - Protects against MITM attacks
  - Provides demonstrative authentication
- SiB can be used in
  - Bi-directional authentication
  - Unidirectional authentication

Class Discussion

- Is SiB practical in any case other than when both devices have cameras and displays?
- Both in the TPM case and other unidirectional authentications, SiB protects only against software-based attacks. Is that sufficient, for example, in the public printer case?
- Can bigger displays and better cameras for current cell phones be used to improve the system?