LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Bachelor Seminar "Logic for Security"
Winter Semester 2026/27

A Short Introduction to Security

Dr. Michael Kirsten

Chair of Theoretical Computer Science and Theorem Proving

(Partially based on slides by Bruno Blanchet and Stéphanie Delaune)

Version of October 21, 2026



Core Principles of Security



Source:
`https://destcert.com/resources/five-pillars-information-security/`

- CIA Triad is guiding model in information security
- **Confidentiality**: protecting information from unauthorized access
- **Integrity**: data are trustworthy, complete, and have not been altered or modified by unauthorized users
- **Availability**: data accessibility when needed

Cryptographic protocols



(1)





Cryptographic protocols

- small programs designed to secure communication (various security goals)
- use cryptographic primitives (e.g. encryption, hash function,







(1) by Fabio Lanari — Internet1.jpg by Rock1997 modified., CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=20995390

Security properties (1)

- **Secrecy**: May an intruder learn some secret message between two honest participants?
- Authentication: Is the agent Alice really talking to Bob?
- Fairness: Alice and Bob want to sign a contract. Alice initiates the protocol. May Bob obtain some advantage?
- Non-repudiation: Alice sends a message to Bob. Alice cannot later deny having sent this message. Bob cannot deny having received the message.
- ...

Security properties: E-voting (2)



Eligibility: only legitimate voters can vote, and only once

Fairness: no early results can be obtained which could influence the remaining voters

Individual verifiability:
a voter can verify that her vote was really counted

Universal verifiability:
the published outcome really is the sum of all the votes



Security properties: E-voting (3)

Privacy: the fact that a particular voter voted in a particular way is not revealed to anyone



Receipt-freeness: a voter cannot prove that she voted in a certain way (this is important to protect voters from coercion)

Coercion-resistance: same as receipt-freeness, but the coercer interacts with the voter during the protocol, (e.g. by preparing messages)

Cryptographic primitives

Cryptographic primitives

Algorithms that are frequently used to build computer security systems. These routines include, but are not limited to, encryption and signature functions.

Cryptographic primitives

Cryptographic primitives

Algorithms that are frequently used to build computer security systems. These routines include, but are not limited to, encryption and signature functions.

Symmetric encryption



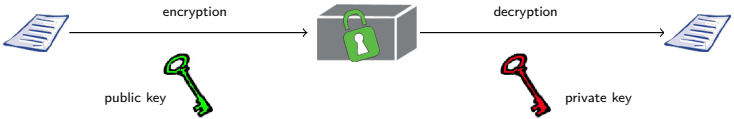
⟶ Examples: Caesar encryption, DES, AES, ...

Cryptographic primitives

Cryptographic primitives

Algorithms that are frequently used to build computer security systems. These routines include, but are not limited to, encryption and signature functions.

Asymmetric encryption

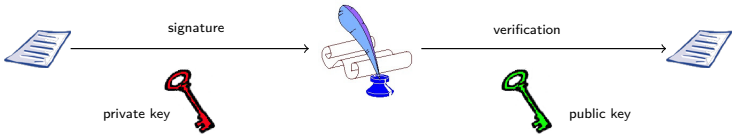


Cryptographic primitives

Cryptographic primitives

Algorithms that are frequently used to build computer security systems. These routines include, but are not limited to, encryption and signature functions.

Signature



Motivating Scenario: The “Dating Problem”

Calculating Mutual Interest with Playing Cards

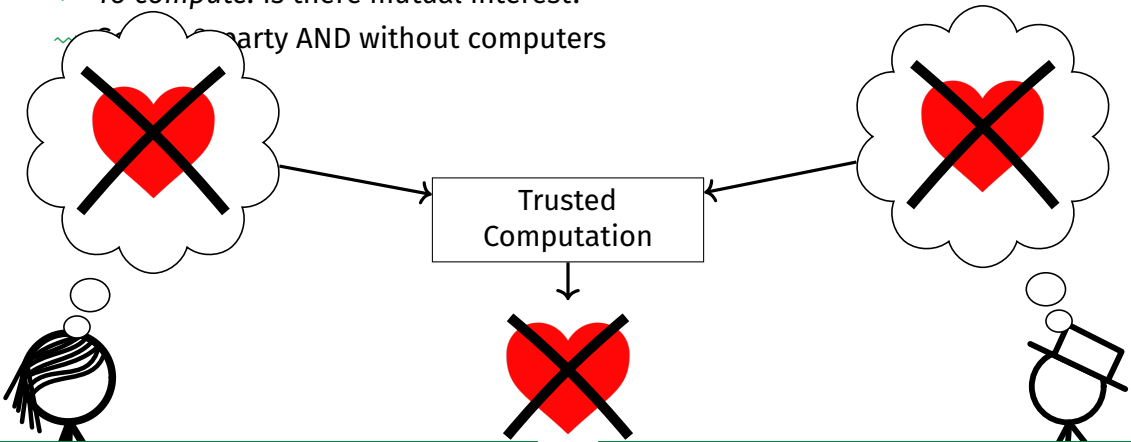
- *Secrets:* Do I fancy the other person?
- *To compute:* Is there mutual interest?

$\rightsquigarrow$ Secure 2-party AND without computers

Motivating Scenario: The “Dating Problem”

Calculating Mutual Interest with Playing Cards

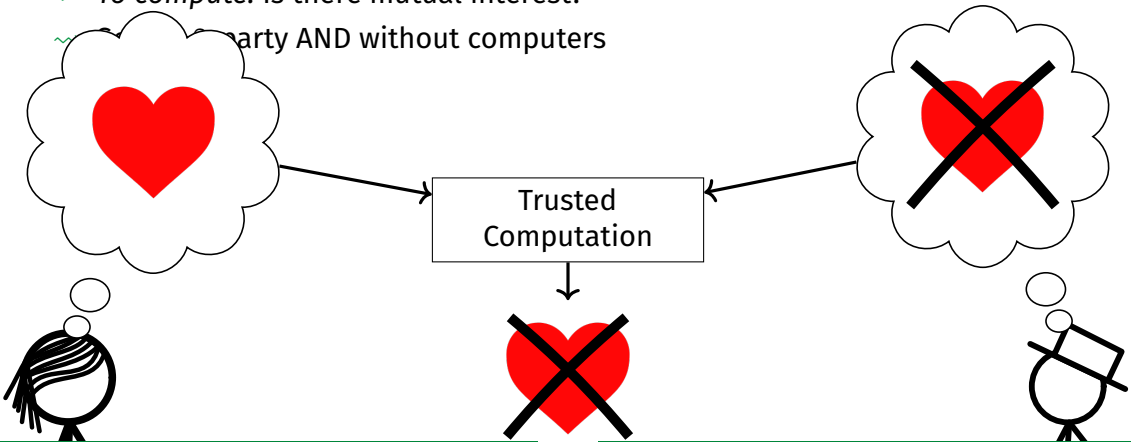
- *Secrets:* Do I fancy the other person?
- *To compute:* Is there mutual interest?
- [illegible]arty AND without computers



Motivating Scenario: The “Dating Problem”

Calculating Mutual Interest with Playing Cards

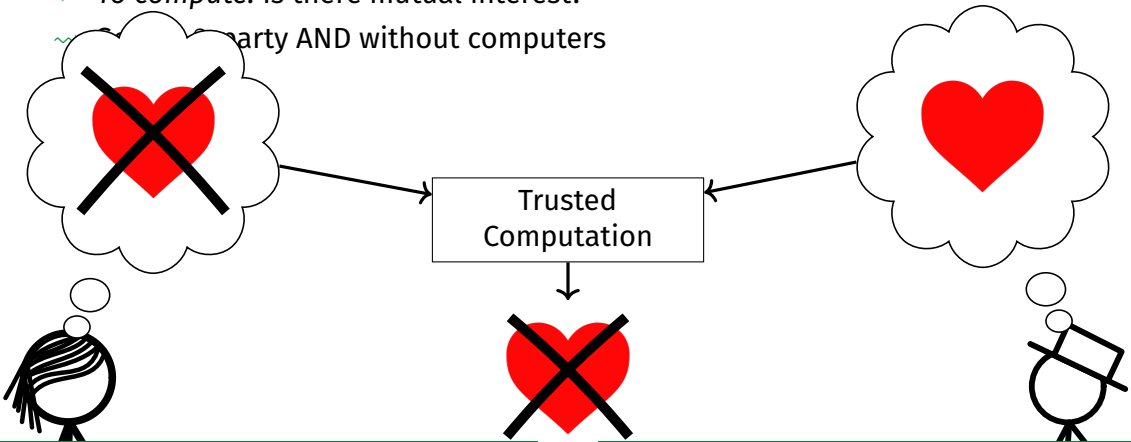
- *Secrets:* Do I fancy the other person?
- *To compute:* Is there mutual interest?
- [illegible]arty AND without computers



Motivating Scenario: The “Dating Problem”

Calculating Mutual Interest with Playing Cards

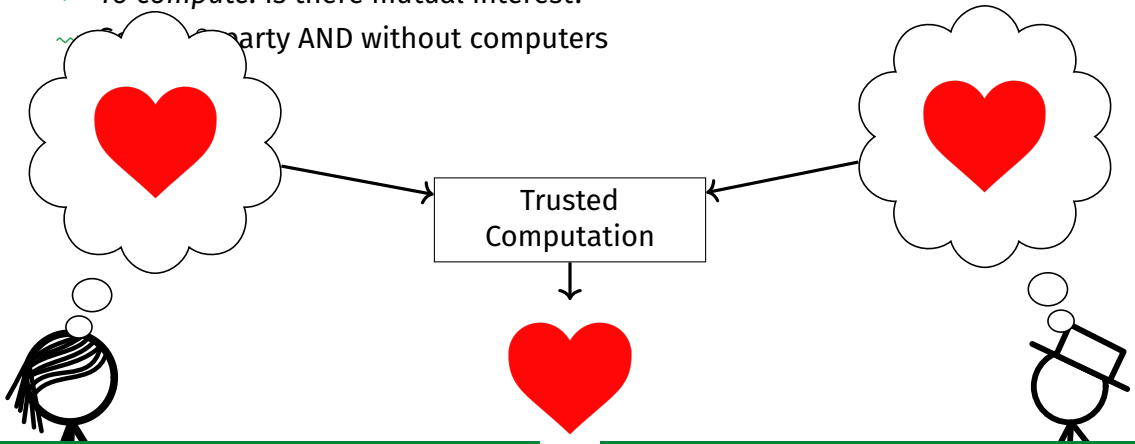
- *Secrets:* Do I fancy the other person?
- *To compute:* Is there mutual interest?
- [illegible]arty AND without computers



Motivating Scenario: The “Dating Problem”

Calculating Mutual Interest with Playing Cards

- *Secrets:* Do I fancy the other person?
- *To compute:* Is there mutual interest?
- [illegible]arty AND without computers



Cryptographic Multiparty Protocols

General Requirements

- **Ballot confidentiality:** Ballots kept secret
- **Ballot integrity:** Ballot casting/processing not manipulated

Cryptographic Multiparty Protocols

General Requirements

- **Ballot confidentiality:** Ballots kept secret
- **Ballot integrity:** Ballot casting/processing not manipulated

Application: Secure Multiparty Computation

- Multiple parties compute common function on their secret bits
- Intermediate states, output, and input are shared secrets
- External observer is convinced of correctness
- Computation is zero-knowledge and observer only learns output

Cryptographic Multiparty Protocols

General Requirements

- **Ballot confidentiality:** Ballots kept secret
- **Ballot integrity:** Ballot casting/processing not manipulated

Application: Secure Multiparty Computation

- Multiple parties compute common function on their secret bits
- Intermediate states, output, and input are shared secrets
- External observer is convinced of correctness
- Computation is zero-knowledge and observer only learns output

Simple Example

Card-based cryptography with two parties for “honest-but-curious” trust model

Card-Based Cryptographic Protocols

A Deck of Cards:

- Indistinguishable back sides: 
- Distinguishable ordered front sides, e.g., 1 2 3 …, or ♣ and ♡

Card-Based Cryptographic Protocols

A Deck of Cards:

- Indistinguishable back sides: 
- Distinguishable ordered front sides, e.g., 1 2 3 …, or ♣ and ♡

Simple Yes/No Vote: $\boxed{a}\,\boxed{b} \mathrel{\hat{=}} \begin{cases} \text{No} & \text{for } a < b \\ \text{Yes} & \text{for } a > b \end{cases}$

Card-Based Cryptographic Protocols

A Deck of Cards:

- Indistinguishable back sides: 
- Distinguishable ordered front sides, e.g., $\boxed{1}$ $\boxed{2}$ $\boxed{3}$ …, or $\boxed{\clubsuit}$ and $\boxed{\heartsuit}$

Simple Yes/No Vote: $\boxed{a}\boxed{b} \mathrel{\hat{=}} \begin{cases} \text{No} & \text{for } a < b \\ \text{Yes} & \text{for } a > b \end{cases}$

Three Actions: (1) *Turning* cards, (2) *Shuffling* (e.g., card cutting), and (3) *Announcing* output positions

Card-Based Cryptographic Protocols

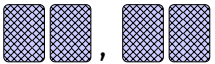
A Deck of Cards:

- Indistinguishable back sides: 
- Distinguishable ordered front sides, e.g., 1 2 3 …, or ♣ and ♡

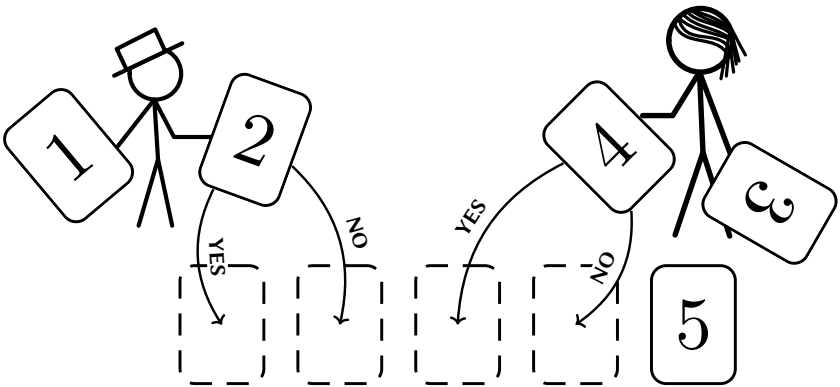
Simple Yes/No Vote: $\boxed{a}\,\boxed{b} \mathrel{\hat{=}} \begin{cases} \text{No} & \text{for } a < b \\ \text{Yes} & \text{for } a > b \end{cases}$

Three Actions: (1) *Turning* cards, (2) *Shuffling* (e.g., card cutting), and (3) *Announcing* output positions

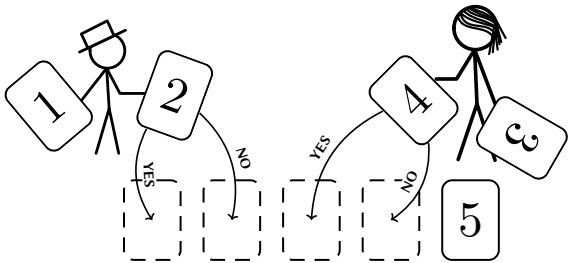
Computation of AND Function (with Two Voters):

- Input: two face-down votes (x, y): 
- Output: face-down result $(x \wedge y)$: 

Example: The “Five-Card Trick”



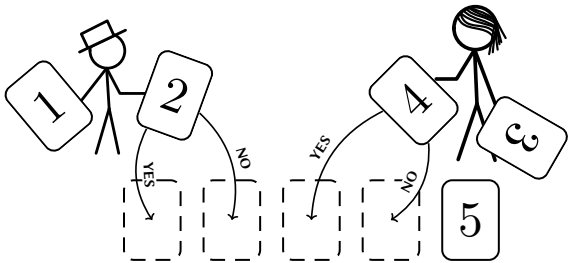
Example: The “Five-Card Trick”



Configurations:

YES/YES (>)	YES/NO (<)	NO/YES (<)	NO/NO (<)
2 1 4 3 5	2 1 3 4 5	1 2 4 3 5	1 2 3 4 5

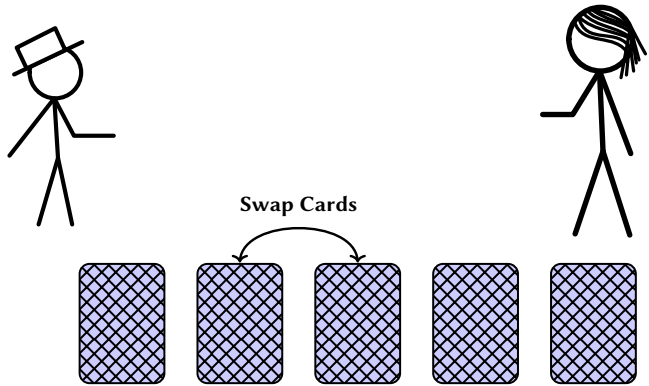
Example: The “Five-Card Trick”



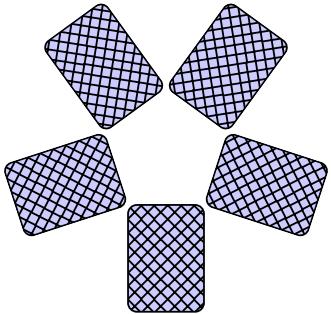
Configurations:

YES/YES (>)	YES/NO (<)	NO/YES (<)	NO/NO (<)
2 1 4 3 5	2 1 3 4 5	1 2 4 3 5	1 2 3 4 5
— —	— —	— —	— —

The “Five-Card Trick” by Niemi and Renvall (1999)



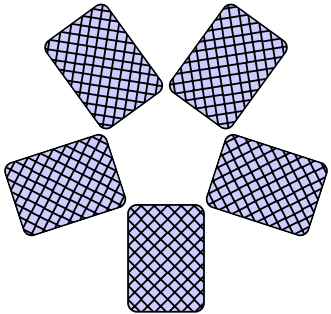
The “Five-Card Trick” by Niemi and Renvall (1999)



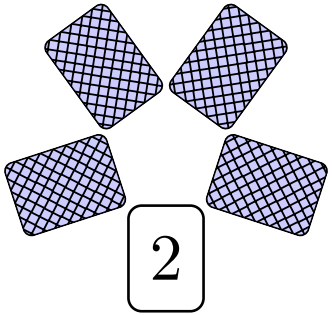
The “Five-Card Trick” by Niemi and Renvall (1999)



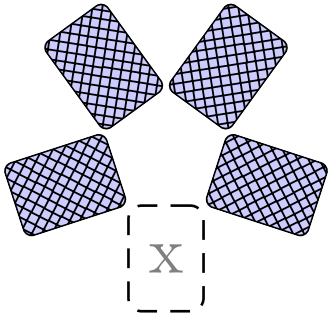
The “Five-Card Trick” by Niemi and Renvall (1999)



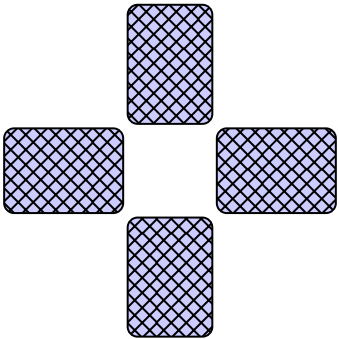
The “Five-Card Trick” by Niemi and Renvall (1999)



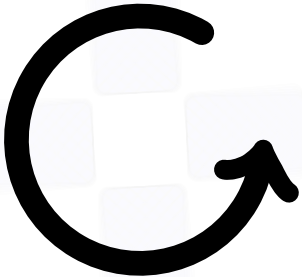
The “Five-Card Trick” by Niemi and Renvall (1999)



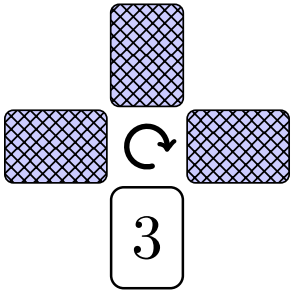
The “Five-Card Trick” by Niemi and Renvall (1999)



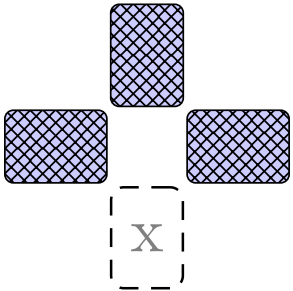
The “Five-Card Trick” by Niemi and Renvall (1999)



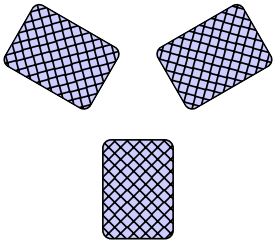
The “Five-Card Trick” by Niemi and Renvall (1999)



The “Five-Card Trick” by Niemi and Renvall (1999)



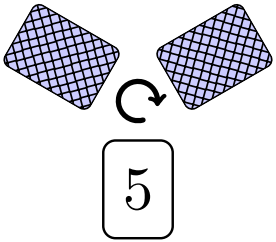
The “Five-Card Trick” by Niemi and Renvall (1999)



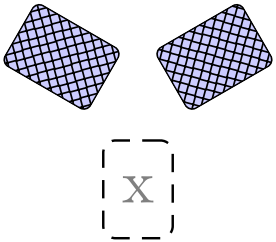
The “Five-Card Trick” by Niemi and Renvall (1999)



The “Five-Card Trick” by Niemi and Renvall (1999)



The “Five-Card Trick” by Niemi and Renvall (1999)



Five-Card AND Protocol by Niemi and Renvall (1999)

1 (perm, (2 3))
2 **repeat**
3 (shuffle, $\langle(1\ 2\ 3\ 4\ 5)\rangle$)
4 $v = (\text{turn}, \{1\})$
5 **until** v == 2 or v == 3
6 **repeat**
7 (shuffle, $\langle(2\ 3\ 4\ 5)\rangle$)
8 $v = (\text{turn}, \{2\})$
9 **until** v == 2 or v == 3
10 **repeat**
11 (shuffle, $\langle(3\ 4\ 5)\rangle$)
12 $v = (\text{turn}, \{3\})$
13 **until** v == 5
14 (result, 4, 5)

Security of cryptographic protocols

How cryptographic protocols can be attacked?

Breaking encryption



cryptanalysis

Logical attack



replay attacks

man-in-the-middle attacks

Symbolic model: example of attacks, replay attacks



transfer 100 euros into
the merchant's account



Symbolic model: example of attacks, replay attacks



transfer 100 euros into
the merchant's account





transfer 100 euros into
the merchant's account



Symbolic model: example of attacks, replay attacks



transfer 100 euros into
the merchant's account





transfer 100 euros into
the merchant's account



transfer 100 euros into
the merchant's account

⋮

transfer 100 euros into
the merchant's account



Symbolic model: example of attacks, replay attacks



transfer 100 euros into
the merchant's account





transfer 100 euros into
the merchant's account



transfer 100 euros into
the merchant's account

⋮

transfer 100 euros into
the merchant's account



Example: attack on the decoders (TV)
$\longrightarrow$ block the message that cancels the subscription

Needham-Schroeder (public-key) Protocol

Needham-Schroeder's Protocol (1978)



- $A \rightarrow B: \quad \{A, N_a\}_{\mathsf{pub}(B)}$
  $B \rightarrow A: \quad \{N_a, N_b\}_{\mathsf{pub}(A)}$
  $A \rightarrow B: \quad \{N_b\}_{\mathsf{pub}(B)}$



Needham-Schroeder's Protocol (1978)



$$
\begin{array}{lcll}
 & A & \rightarrow & B: \quad \{A, N_a\}_{\mathsf{pub}(B)} \\
\bullet & B & \rightarrow & A: \quad \{N_a, N_b\}_{\mathsf{pub}(A)} \\
 & A & \rightarrow & B: \quad \{N_b\}_{\mathsf{pub}(B)}
\end{array}
$$



Needham-Schroeder's Protocol (1978)



$A \rightarrow B : \{A, N_a\}_{\mathsf{pub}(B)}$

$B \rightarrow A : \{N_a, N_b\}_{\mathsf{pub}(A)}$

- $A \rightarrow B : \{N_b\}_{\mathsf{pub}(B)}$



Needham-Schroeder's Protocol (1978)



$$
\begin{array}{lcll}
A & \rightarrow & B: & \{A, N_a\}_{\mathsf{pub}(B)} \\
B & \rightarrow & A: & \{N_a, N_b\}_{\mathsf{pub}(A)} \\
A & \rightarrow & B: & \{N_b\}_{\mathsf{pub}(B)}
\end{array}
$$



Needham-Schroeder's Protocol (1978)



$$
\begin{array}{lcll}
A & \rightarrow & B: & \{A, N_a\}_{\mathsf{pub}(B)} \\
B & \rightarrow & A: & \{N_a, N_b\}_{\mathsf{pub}(A)} \\
A & \rightarrow & B: & \{N_b\}_{\mathsf{pub}(B)}
\end{array}
$$



Questions

- Is N_b secret between A and B ?
- When B receives $\{N_b\}_{\mathsf{pub}(B)}$, does this message really comes from A ?

Needham-Schroeder's Protocol (1978)



$$
\begin{array}{lcll}
A & \rightarrow & B: & \{A, N_a\}_{\mathsf{pub}(B)} \\
B & \rightarrow & A: & \{N_a, N_b\}_{\mathsf{pub}(A)} \\
A & \rightarrow & B: & \{N_b\}_{\mathsf{pub}(B)}
\end{array}
$$



Questions

- Is N_b secret between A and B ?
- When B receives $\{N_b\}_{\mathsf{pub}(B)}$, does this message really comes from A ?

Attack

An attack was found 17 years after its publication! [Lowe 96]

Example: Man in the middle attack



Agent A



Intruder I



Agent B

Attack

- involving 2 sessions in parallel,
- an honest agent has to initiate a session with I.

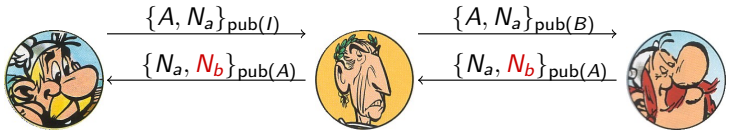
$$
\begin{aligned}
A \rightarrow B &: \{A, N_a\}_{pub(B)} \\
B \rightarrow A &: \{N_a, N_b\}_{pub(A)} \\
A \rightarrow B &: \{N_b\}_{pub(B)}
\end{aligned}
$$

Example: Man in the middle attack



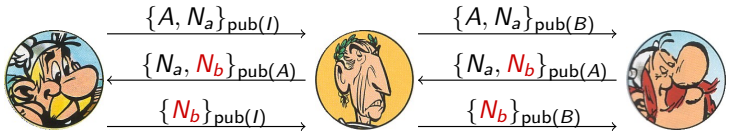
$$
\begin{array}{lll}
\mathsf{A} \to \mathsf{B} & : & \{A, N_a\}_{\mathsf{pub}(B)} \\
\mathsf{B} \to \mathsf{A} & : & \{N_a, N_b\}_{\mathsf{pub}(A)} \\
\mathsf{A} \to \mathsf{B} & : & \{N_b\}_{\mathsf{pub}(B)}
\end{array}
$$

Example: Man in the middle attack



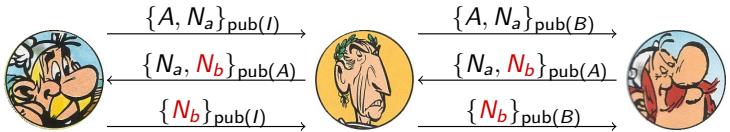
$$
\begin{array}{lll}
\mathsf{A} \to \mathsf{B} & : & \{A, N_a\}_{\mathrm{pub}(B)} \\
\mathsf{B} \to \mathsf{A} & : & \{N_a, N_b\}_{\mathrm{pub}(A)} \\
\mathsf{A} \to \mathsf{B} & : & \{N_b\}_{\mathrm{pub}(B)}
\end{array}
$$

Example: Man in the middle attack



$$
\begin{aligned}
\mathsf{A} \to \mathsf{B} &: \{A, N_a\}_{\mathrm{pub}(B)} \\
\mathsf{B} \to \mathsf{A} &: \{N_a, N_b\}_{\mathrm{pub}(A)} \\
\mathsf{A} \to \mathsf{B} &: \{N_b\}_{\mathrm{pub}(B)}
\end{aligned}
$$

Example: Man in the middle attack



Attack

- the intruder knows N_b,
- When B finishes his session (apparently with A), A has never talked with B.

$$\begin{aligned} A \to B &: \{A, N_a\}_{\mathrm{pub}(B)} \\ B \to A &: \{N_a, N_b\}_{\mathrm{pub}(A)} \\ A \to B &: \{N_b\}_{\mathrm{pub}(B)} \end{aligned}$$

TLS: An implementation attack



Le Monde.fr

TECHNOLOGIES | Jeux vidéo | Hits Playtime | Libertés numériques | Téléphonie mobile | Droit d'auteur | Rése

Faille Heartbleed : les sites pour lesquels il est conseillé de changer son mot de passe

Le Monde.fr | 11.04.2014 à 04h48 • Mis à jour le 14.04.2014 à 15h08 |

Par Michael Szadkowski

Abonnez-vous à partir de 1 € Réagir Classer Partager

Recommander Partager 14 010 personnes le recommandent.

Le changement de son mot de passe est conseillé pour tous les sites qui ont annoncé avoir fait une mise à jour de sécurité, après la découverte du bug qui affecte le protocole OpenSSL. | REUTERS/PAWEL KOPCZYNSKI

Deux jours après la révélation d'une faille de sécurité au sein du protocole OpenSSL, baptisée « Heartbleed », cette dernière est décrite par certains comme *« le pire cauchemar »* qui puisse arriver concernant la sécurité des échanges sur Internet.

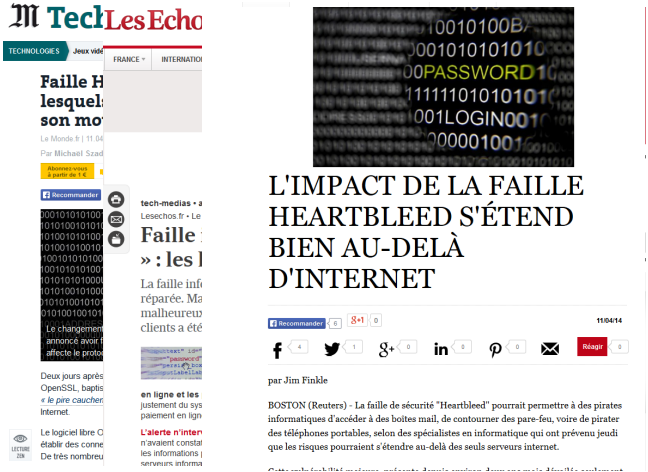
LECTURE 22h

Le logiciel libre OpenSSL est installé sur les serveurs de très nombreux sites pour établir des connexions chiffrées et sécurisées entre ce dernier et ses utilisateurs. De très nombreux sites Internet utilisent OpenSSL pour sécuriser leurs échanges.

TLS: An implementation attack



TLS: An implementation attack

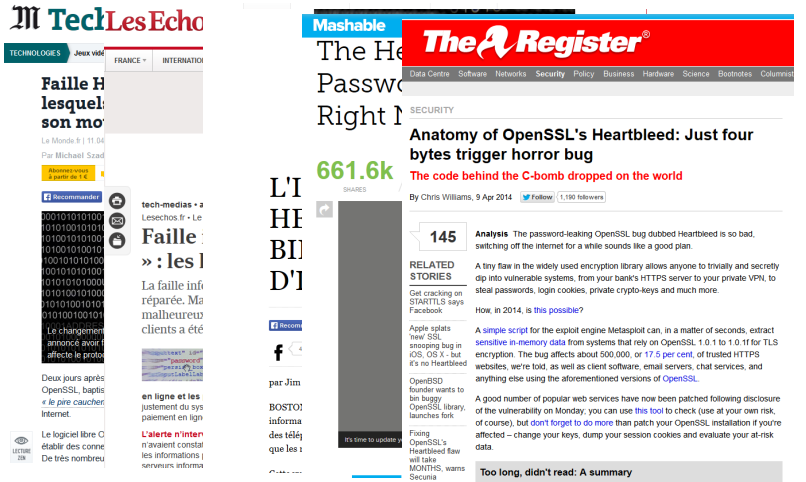


TLS: An implementation attack



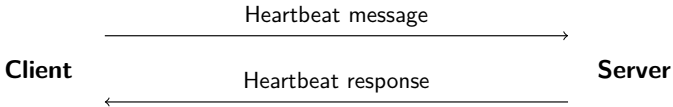


TLS: An implementation attack



Heartbleed explained (1): heartbeat

- SSL/TLS: protocol used for `https://` web pages.
- Heartbeat: an extension of TLS that just checks that a server is alive.
  - The client sends a packet to the server.
  - The server replies to the client with the same packet.



Heartbleed explained (2): heartbeat message in detail

SSLv3 record

Length of HeartbeatMessage

HeartbeatMessage

Type	Length of payload	Payload data
TLS1_HB_REQUEST		

Heartbleed explained (3): the attack

OpenSSL 1.0.1 before 1.0.1g does not check that the lengths are coherent!

Heartbeat attack message

SSLv3 record

Length of HeartbeatMessage
4 bytes

HeartbeatMessage

Type	Length of payload	Payload data
TLS1_HB_REQUEST	65535 bytes	1 byte

Heartbeat victim's response

SSLv3 record

Length of HeartbeatMessage
65538 bytes

HeartbeatMessage

Type	Length of payload	Payload data	
TLS1_HB_RESPONSE	65535 bytes	1 byte	65534 bytes leaked

Source: http://www.theregister.co.uk/2014/04/09/heartbleed_explained/