Introduction to Authenticated Key Agreement

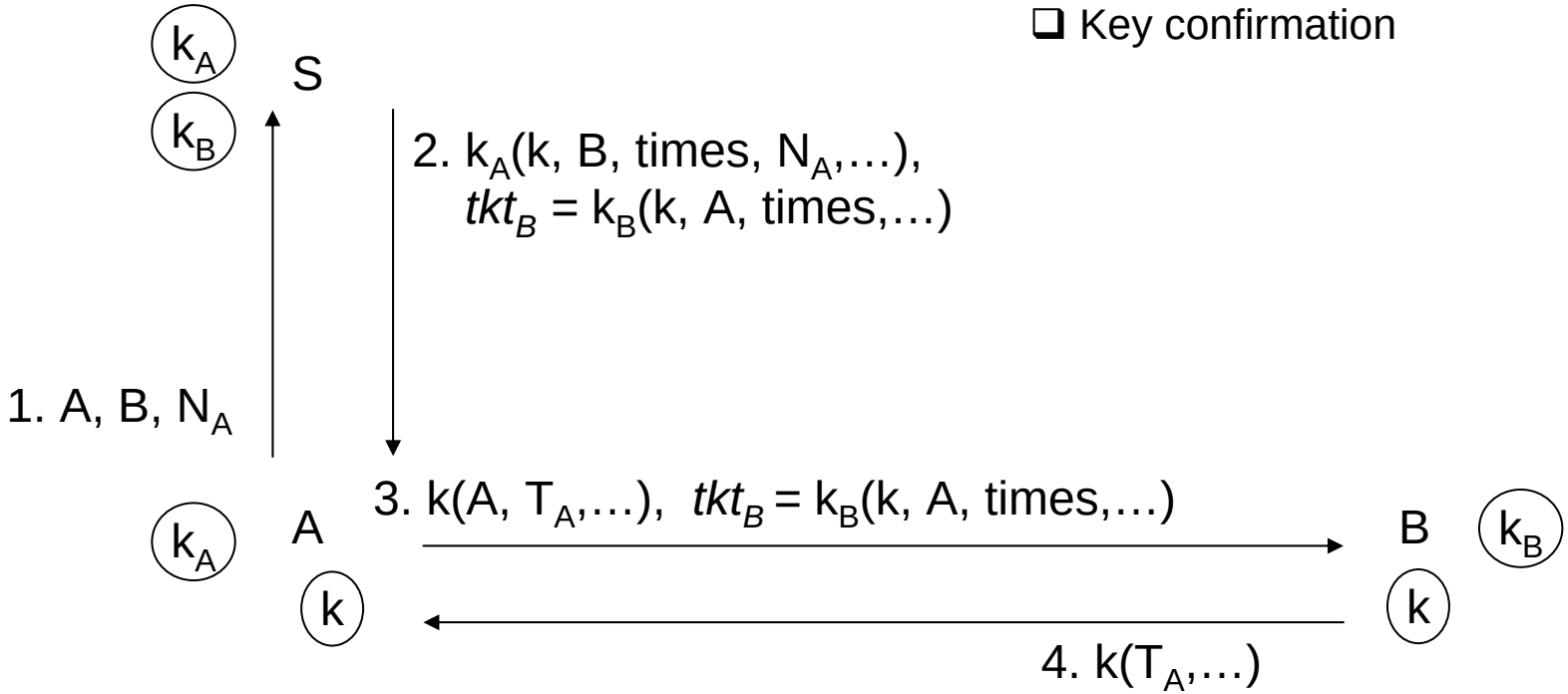
Kaisa Nyberg

29.9.2006

Textbook: W. Mao. Modern Cryptography T&P

Example: Kerberos

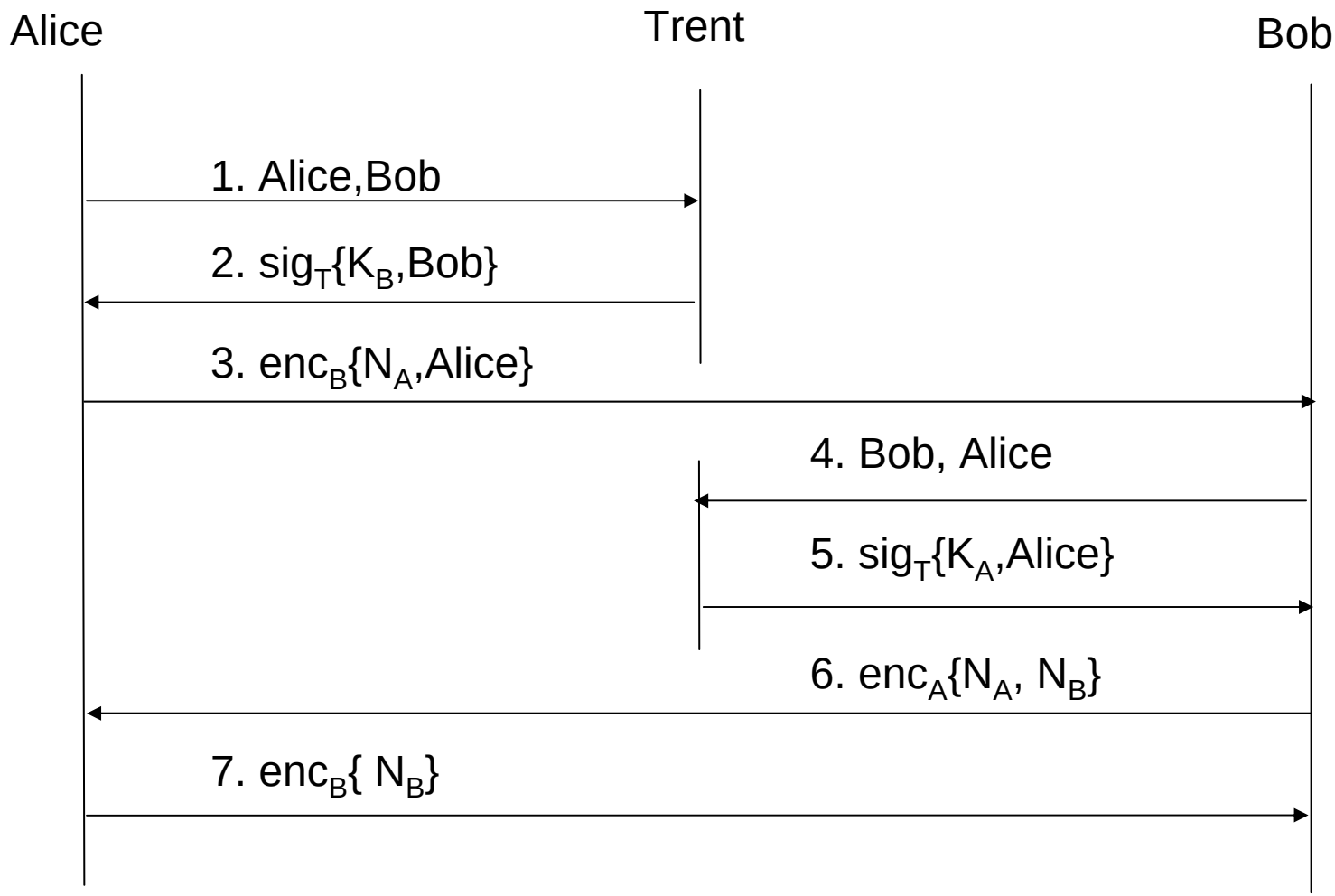
- Prior enrollment with server
  - Basis for authentication and key exclusivity
- Timestamps to ensure freshness
- Key transport
  - Key confirmation



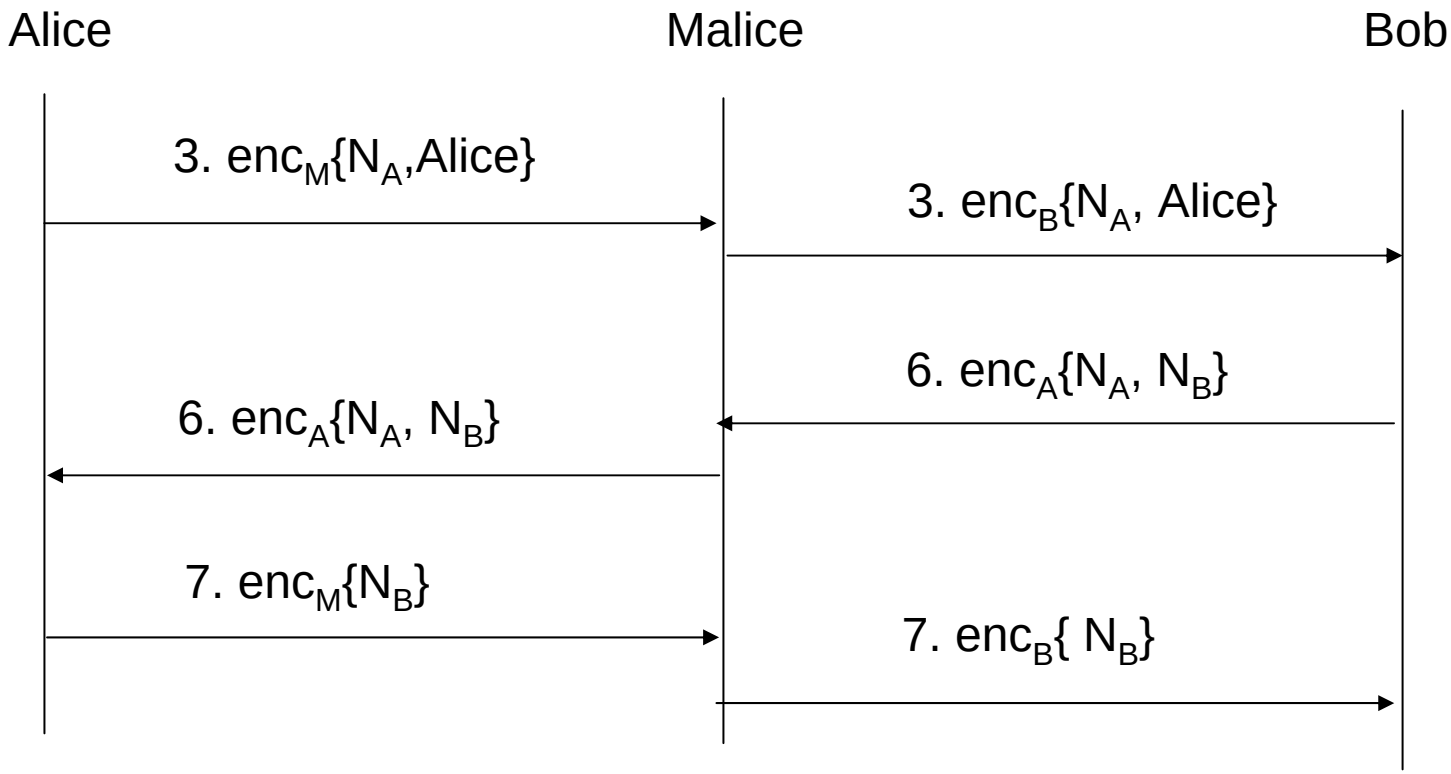
Needham-Schroeder protocol (1978)

- An earlier version of the Kerberos protocol (without time-stamps)
  - B had no guarantee of the freshness of the ticket tkt_B. If Malice knows some previous key used by A and B it can force B to use the key again by replaying the corresponding ticket.

Needham-Schroeder Public Key Protocol



Attack using two simultaneous protocol runs (Lowe 1995)



Alice thinks she is talking to Malice, while Bob thinks he is talking to Alice

Fix: In message 6, insert Bob's name

Lesson to learn

- If the identity of a principal is essential to the meaning of a message, it is prudent to mention the principal's name explicitely in the message.
- But after the fix, can we be sure that the protocol is secure?

Authentication Notions: Data authentication

- Data (data-origin, message) authentication involves
  - Communications, receiver and transmitter
  - identifying the source of the data
  - freshness of a message (non-replay)
- Successful validation by the receiver establishes
  - the identity of the message transmitter
  - liveness (at some point) of the message transmitter
  - integrity of the data subsequent to being transmitted

Authentication Notions:
Entity authentication

- Entity Authentication is
  - is a lively correspondence by a principal with a second principal
  - Aims to corroborate the identity of the second entity
  - Entity authentication is very rarely the only goal of the security protocol
  - Entity authentication may be performed by other than cryptographic means (e.g., manual authentication)

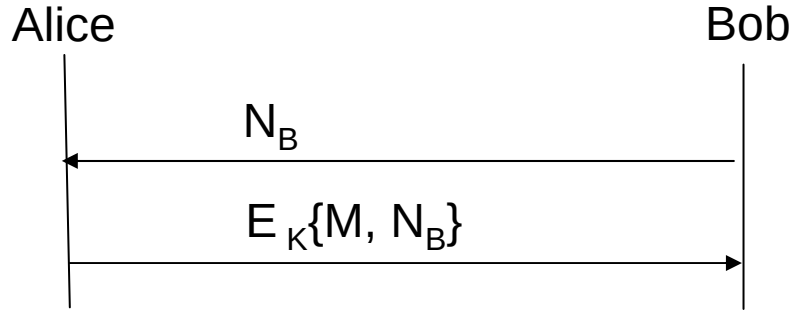
Authentication Notions: Authenticated Key Agreement

- Authenticated Key Agreement involves
  - Establishment of a cryptographic key between two entities
  - Data authentication of the established cryptographic key

- Security
  - Authentication protocol is flawed if a principal concludes a normal run of the protocol while the intended other principal would have a different conclusion.
  - A flaw in a protocol does not necessarily imply a flaw in the cryptographic algorithms used in the protocol.
  - Important to validate the suitability of a cryptographic algorithm for the protocol. Here theoretical models and security proofs are useful.

Basic Authentication Techniques: Message Freshness and Principal Liveness

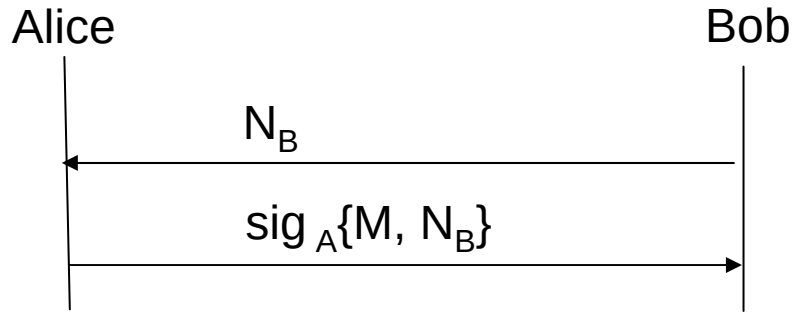
- Challenge-Response Mechanism
  - Alice and Bob share a key K of an encryption algorithm.
  - Alice has a message M, she wants to transmit to Bob.
  - Bob wants verify the freshness of M and liveness of Alice



  - It is necessary that the encryption algorithm offers data-integrity. If confidentiality is not needed then better to use a message authentication algorithm.

Challenge-Response Mechanism using digital signature

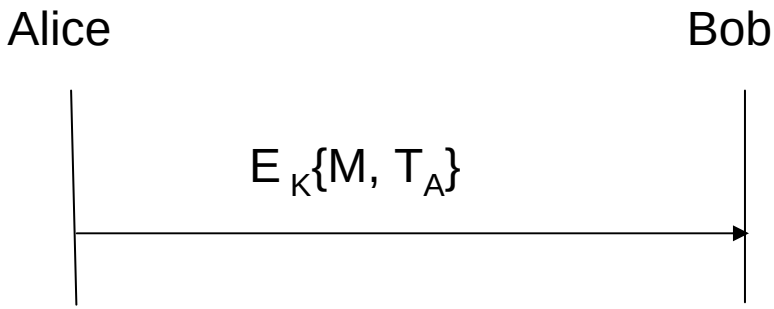
- Challenge-Response Mechanism
  - Alice uses digital signature mechanism, Bob has Alice's public key.
  - Alice has a message M, she wants to transmit to Bob.
  - Bob wants verify the freshness of M and liveness of Alice



  - Alice's free choice of M is important. Also, N_B shall never been taken to have some other meaning as the random challenge. Otherwise Bob can compute it as a hash of some contract beneficial to him.

Basic Authentication Techniques: Message Freshness and Principal Liveness

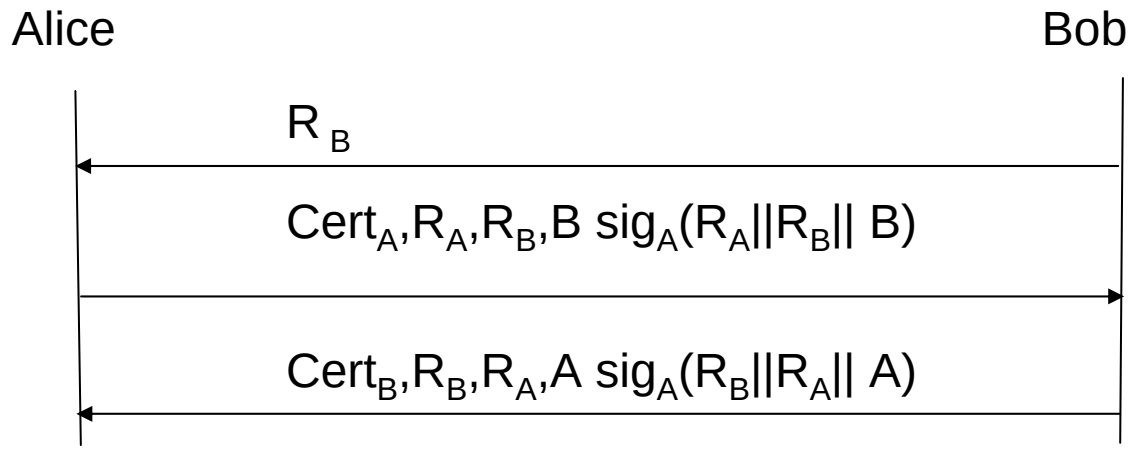
- Non-interactive variant
  - Alice and Bob share a key K of an encryption algorithm.
  - Timestamp mechanism



  - Requires synchronized clocks or counters
    - Disadvantages of counters: do not scale to a large number of principals, get easily out of synchronization, still widely used in wireless communication.

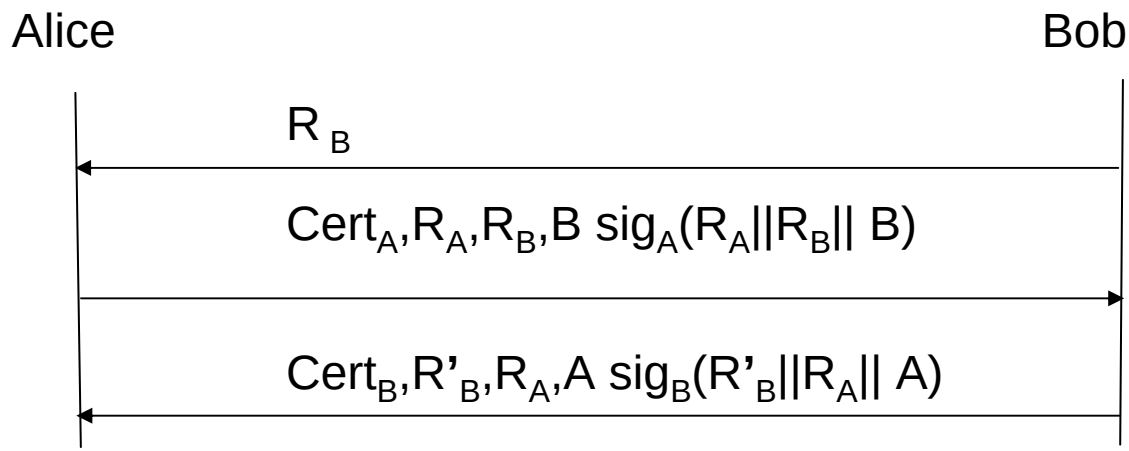
Mutual Authentication

- Mutual Challenge-Response using digital signatures (ISO 9798-3)
  - Premise: A has public key (signature) certificate $Cert_A$ and B has a public key certificate $Cert_B$



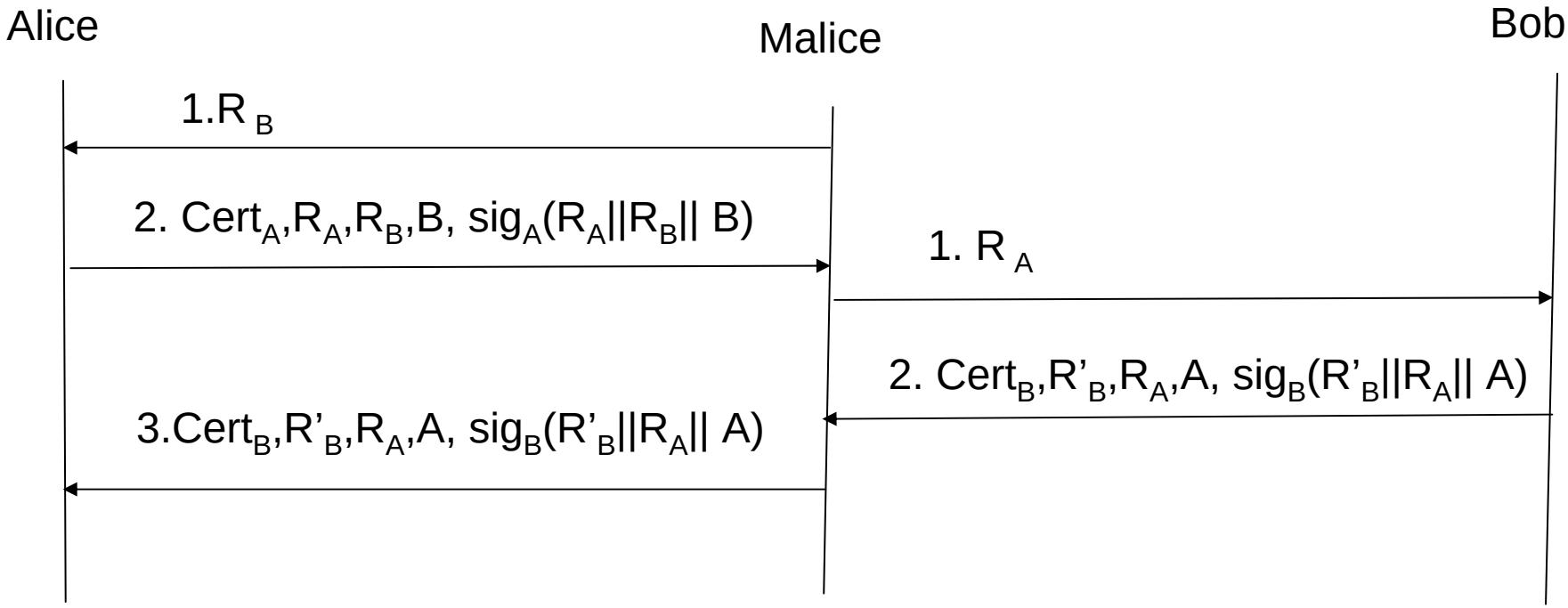
Mutual Authentication

- Original version of ISO 9798-3



  - The reasoning was that in this manner B can change the message he is signing

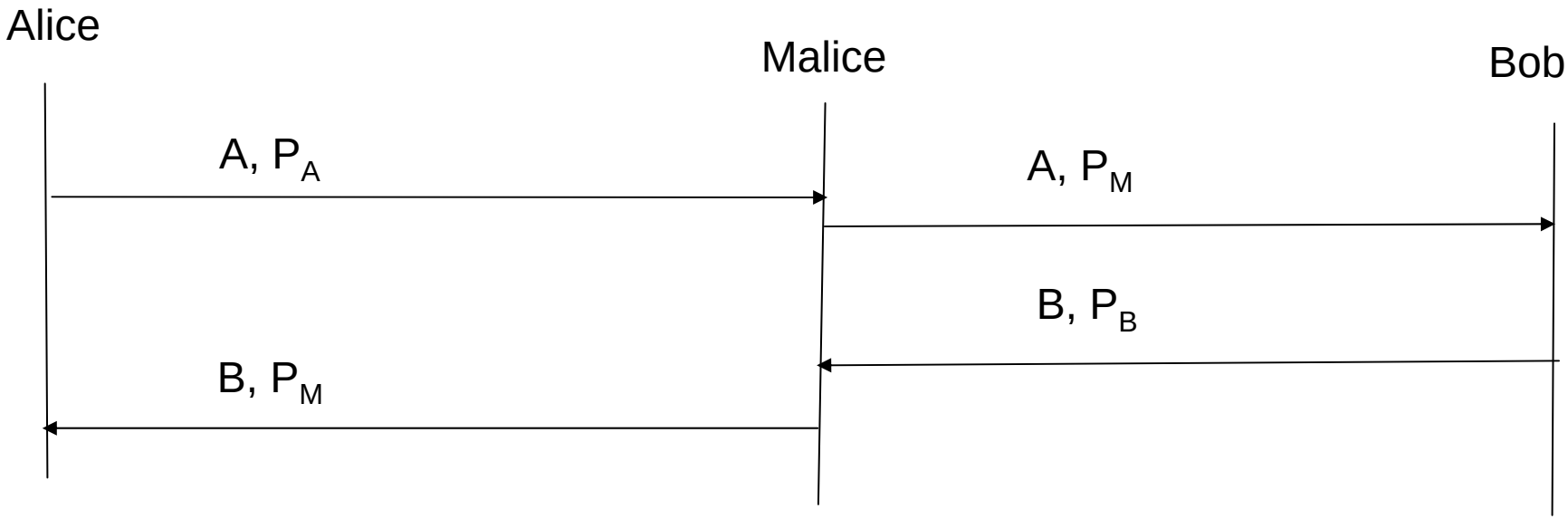
Mutual Authentication: The Canadian Attack (1992) (reflection)



- Alice thinks she is talking to Bob, while Bob is still waiting the protocol to complete.
- Malice uses Bob in a run of the protocol as an oracle to get a valid message for a different run of the protocol.

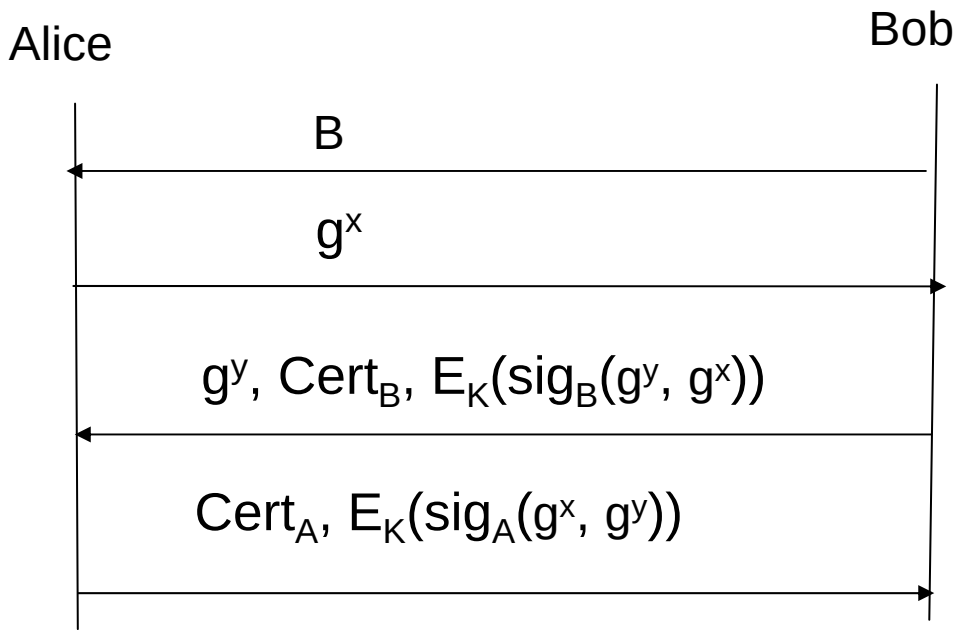
Authenticated Key Exchange based on Asymmetric Cryptography

- Public Key Cryptography does not solve the authentication problem: Malice(Man)-in-the-Middle (MitM)



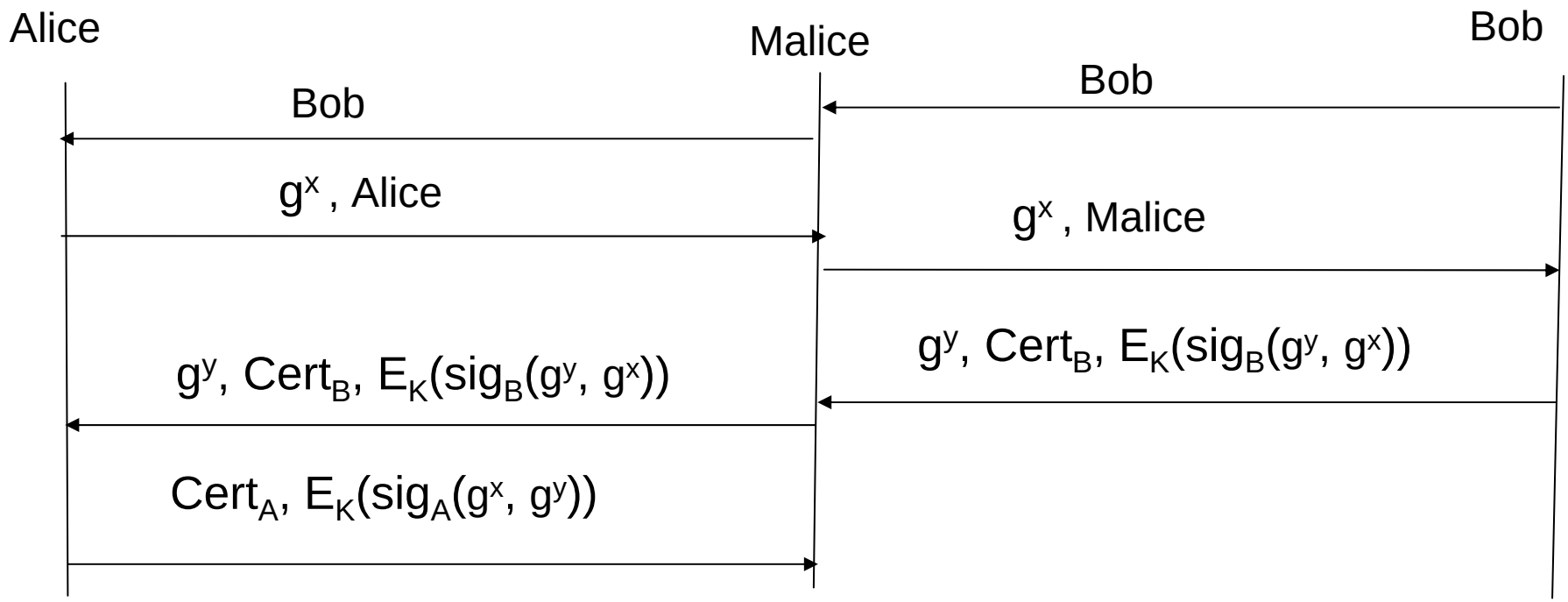
STS Protocol: Authenticated Diffie-Hellman

- Provides *perfect forward secrecy (PFS)*: compromise of long term private keys does not compromise past session keys
- *PFS* requires the use of public key cryptography
- *PFS* needed only if session keys are used to protect long term confidentiality



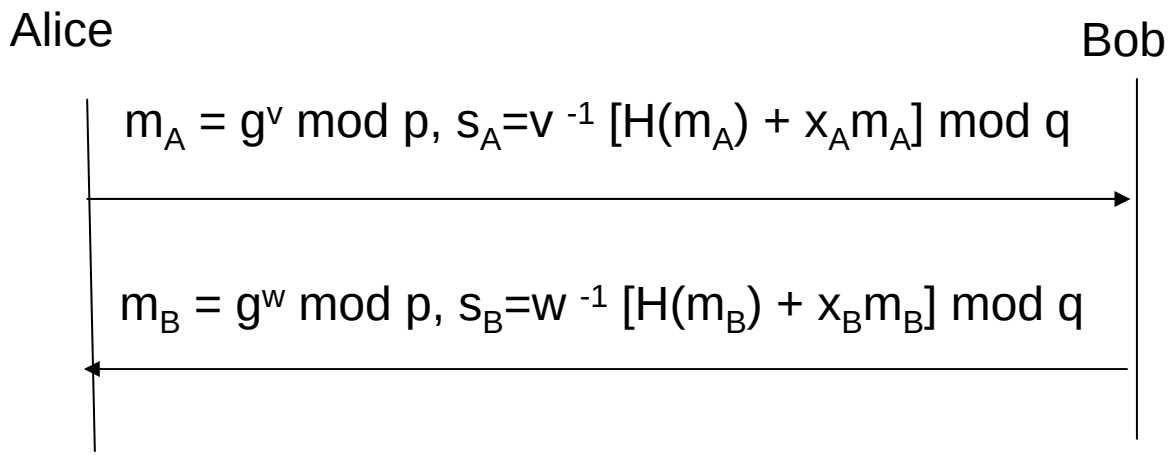
STS Protocol: Lowe's Attack

- Minor flaw: Authentication fails (why?), but Malice does not get to know the shared secret key $K = g^{xy}$



Integrating Diffie-Hellman KE and DLP-based signatures: Arazi (1993)

- Alice and Bob have DSS parameters g, p and q
- Alices signature key pair x_A, y_A, Bob's key pair x_B, y_B



- Shared key $K = g^{vw} \bmod p$ (Diffie-Hellman key)
- *PFS*

Flaw in Arazi's scheme
Nyberg-Rueppel 1994 [4]

$$vw = s_A^{-1} s_B^{-1} [H(m_A)H(m_B) + H(m_A)x_B m_B + H(m_B)x_A m_A + x_A x_B m_A m_B] \bmod q$$

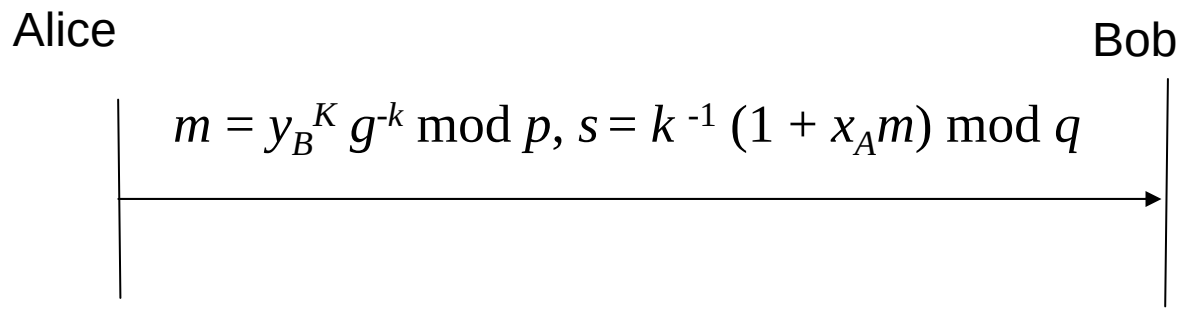
$$K^{s_A s_B} = g^{H(m_A)H(m_B)} y_B^{H(m_A)m_B} y_A^{H(m_B)m_A} (g^{x_A x_B})^{m_A m_B} \bmod p$$

- Given K (ephemeral secret) one can compute $g^{x_A x_B}$ and vice versa

- Arazi's scheme does not resist *known-key attack* (or independency of the session keys)
- Security against known key attack means that an agreed key will not be compromised even if agreed keys derived from the same long-term keying material in a subsequent run are compromised.

Combining ElGamal encryption and DLB-based signatures: NR (1996) [3]

- Alice and Bob have DSS parameters g, p and q
- Alices signature key pair x_A, y_A, Bob's key pair x_B, y_B



- Bob computes $g^k \bmod p$ as $g^{s^{-1}} \, y_A^{s^{-1} m} \bmod p$
- then $y_B^K = m \, g^{-k} \bmod p$ and finally g^K from y_B^K by raising it to power $(x_B)^{-1}$
- Shared key = $g^K \bmod p$

Desirable AKE Attributes

Law, Menezes, Qu, Solinas, Vanstone (1998) [2]

- *known-key security.* Each run of a key agreement protocol between two identities A and B should produce a unique secret key; such keys are *session keys.* A protocol should still achieve its goal in the face of an adversary who has learned some other session keys.
  - Example: Arazi's scheme does not provide known-key security
- *(perfect) forward secrecy*. If long-term private keys of one or more entities are compromised, the secrecy of previous session keys established by honest entities is not affected.
  - Example: STS protocol provides *PFS*
- *key-compromise impersonation*. Suppose A's long-term private key is disclosed. Clearly an adversary that knows this value can now impersonate A, since it is precisely this value that identifies A. However, it may be desirable that this loss does not enable an adversary to impersonate other entities to A.
  - Such a vulnerability occurs if A uses its own signatures to corroborate other entities' identities

Desirable AKE Attributes (cont'd)

- *unknown key-share*. Entity A cannot be coerced into sharing a key with entity B without A's knowledge, i.e., when A believes the key is shared with some entity C ≠ B, and B (correctly) believes that the key is shared with A.
  - Example: Assume that in Needham-Schroeder public key method Alice sends a D-H key share in message 3 and Bob sends his D-H keyshare in message 6. Assume Malice runs the attack described on page 5. Then the protocol ends in a situation where Alice believes she sahres the key with Malice, while in reality, she shares the key with Bob, and Bob correctly believes that he shares the key with Alice.
- *key control*. Neither entity should be able to force the session key to a pre-selected value.
  - Example, p.21, non-interactive AKE that does not provide key control to Bob

MQV and HMQV

- MQV Menzes-Qu-Vanstone (1995) revised 2003 [2]
- HMQV Hugo Krawczyk (2005) [1]
- Prerequisites: Alice and Bob have long-term DLP-based private keys *a* and *b* and public keys *A* and *B*. They run the basic Diffie-Hellman protocol and have ephemeral private keys *x* and *y* and public keys *X* and *Y (*but do not compute the shared secret as g^{xy}).
- They compute
  - (MQV) $d = 2^{\ell}+(X \bmod 2^{\ell})$ and $e = 2^{\ell}+(Y \bmod 2^{\ell})$, where $\ell = |q|/2$.
  - (HMQV) *d* and *e* as above, but: *X* replaced by *H*(*X*,Bob) and *Y* replaced by *H*(*Y*,Alice)
- A computes the key as $(YB^{e})^{x+da}$
- B computes the key as $(XA^{d})^{y+eb}$
- Elliptic curve variant exists

NR signatures and MQV

- Alice's NR-signature on message m is (r_A, s_A) where

  $$r_A = f(g^x \bmod p)\cdot m,\ s_A = (x + ar_A) \bmod q$$

  and f is an easily computable function.

- The shared key in MQV and HMQV is

  $$g^{s_A s_B} \bmod p$$

  where the signatures are taken for $m = 1$ and f is a suitable function.

- The strength of MQV compared to Arazi's scheme lies in the fact that the second parts of signatures are never sent and remain secret.

References

1. Hugo Krawczyk. HMQV: A High-Performance Secure Diffie-Hellman Protocol, Cryptology ePrint Archive: Report 2005/176
2. Laurie Law, Alfred Menezes, Minghua Qu, Jerry Solinas, Scott A. Vanstone, An Efficient Protocol for Authenticated Key Agreement. Des. Codes Cryptography 28(2): pp119–134 (2003)
3. KN and Rainer A. Rueppel. Message Recovery for Signature Schemes Based on the Discrete Logarithm Problem. Des. Codes Cryptography 7(1): pp61–81 (1996)
4. KN and Rainer A. Rueppel. Weaknesses in some recent key agreement protocols. Electronic Letters, 30, No 1 (1994) 26-27