

Cryptographic protocols

Myrto Arapinis
School of Informatics
University of Edinburgh

February 5, 2015

Context

Applications exchanging **sensitive data** over a **public network**:

- ▶ eBanking,
- ▶ eCommerce,
- ▶ eVoting,
- ▶ ePassports,
- ▶ Mobile phones,
- ▶ ...

Context

Applications exchanging **sensitive data** over a **public network**:

- ▶ eBanking,
- ▶ eCommerce,
- ▶ eVoting,
- ▶ ePassports,
- ▶ Mobile phones,
- ▶ ...

A malicious agent can:

- ▶ record, alter, delete, insert, redirect, reorder, and reuse past or current messages, and inject new messages
→ **the network is the attacker**
- ▶ control dishonest participants

More complex systems needed...

More complex systems needed...



$$\frac{e = E(K_E, \text{Transfer 100 € on Amazon's account})}{m = \text{MAC}(K_M, E(K_E, \text{Transfer 100 € on Amazon's account}))} \rightarrow$$



More complex systems needed...



$$\frac{e = E(K_E, \text{Transfer 100 € on Amazon's account})}{m = \text{MAC}(K_M, E(K_E, \text{Transfer 100 € on Amazon's account}))} \rightarrow$$



Replay attack



$$\xrightarrow{(e,m)}$$



$$\xrightarrow{(e,m)}$$

⋮

$$\xrightarrow{(e,m)}$$



... to achieve more complex properties

- ▶ **Confidentiality:** Some information should never be revealed to unauthorised entities.
- ▶ **Integrity:** Data should not be altered in an unauthorised manner since the time it was created, transmitted or stored by an authorised source.
- ▶ **Authentication:** Ability to know with certainty the identity of an communicating entity.
- ▶ **Anonymity:** The identity of the author of an action (e.g. sending a message) should not be revealed.
- ▶ **Unlinkability:** An attacker should not be able to deduce whether different services are delivered to the same user
- ▶ **Non-repudiation:** The author of an action should not be able to deny having triggered this action.
- ▶ ...

Cryptographic protocols

Cryptographic protocols

Programs relying on **cryptographic primitives** and whose goal is the establishment of “secure” communications.

Cryptographic protocols

Cryptographic protocols

Programs relying on **cryptographic primitives** and whose goal is the establishment of “**secure**” communications.

But!

Many exploitable errors are due not to design errors in the primitives, but to the way they are used, *i.e.* bad protocol design and buggy or not careful enough implementation

Numerous deployed protocols are flawed!!!

Needham-Schroeder protocol - G. Lowe, "An attack on the Needham-Schroeder public-key authentication protocol"

Kerberos protocol - I. Cervesato, A. D. Jaggard, A. Scedrov, J. Tsay, and C. Walstad, "Breaking and fixing public-key kerberos"

Single-Sign-On protocol - A. Armando, R. Carbone, L. Compagna, J. Cuellar, and M. L. Tobarra, "Formal analysis of SAML 2.0 web browser single sign-on: breaking the SAML-based single sign-on for google apps"

PKCS#11 API - M. Bortolozzo, M. Centenaro, R. Focardi, and G. Steel, "Attacking and fixing PKCS#11 security tokens"

BAC protocol - T. Chothia, and V. Smirnov, "A traceability attack against e-passports"

AKA protocol - M. Arapinis, L. Mancini, E. Ritter, and M. Ryan, "New privacy issues in mobile telephony: fix and verification"

Logical attacks

Many of these attacks do not even break the crypto primitives!!

Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA

Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA

A

|

B

|

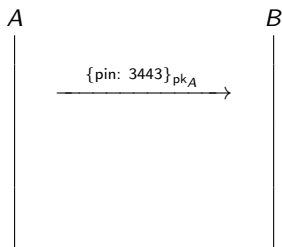
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA



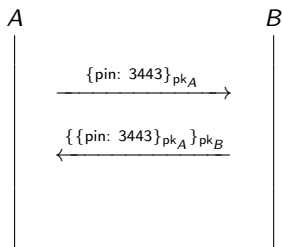
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA



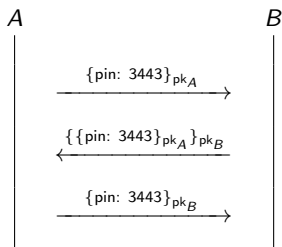
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA



since $\{\{\text{pin: 3443}\}_{pk_A}\}_{pk_B} = \{\{\text{pin: 3443}\}_{pk_B}\}_{pk_A}$ by commutativity

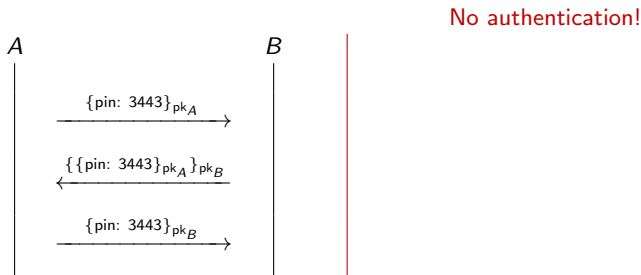
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA



since $\{\{\text{pin: 3443}\}_{pk_A}\}_{pk_B} = \{\{\text{pin: 3443}\}_{pk_B}\}_{pk_A}$ by commutativity

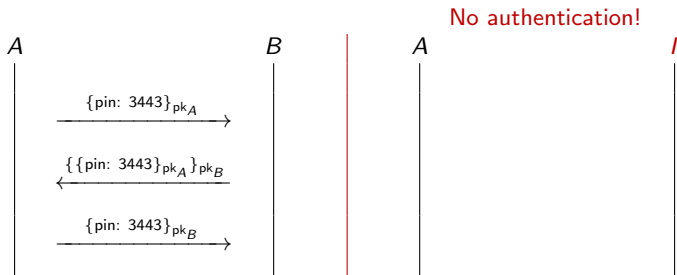
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA



since $\{\{\text{pin: 3443}\}_{pk_A}\}_{pk_B} = \{\{\text{pin: 3443}\}_{pk_B}\}_{pk_A}$ by commutativity

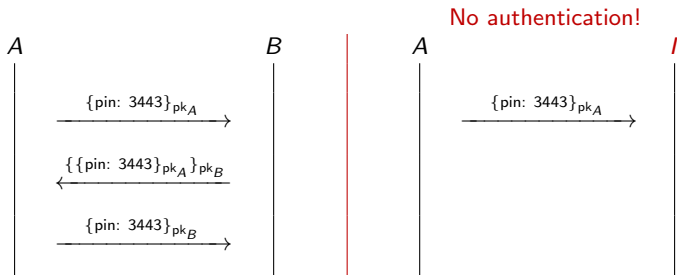
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA



since $\{\{\text{pin: 3443}\}_{pk_A}\}_{pk_B} = \{\{\text{pin: 3443}\}_{pk_B}\}_{pk_A}$ by commutativity

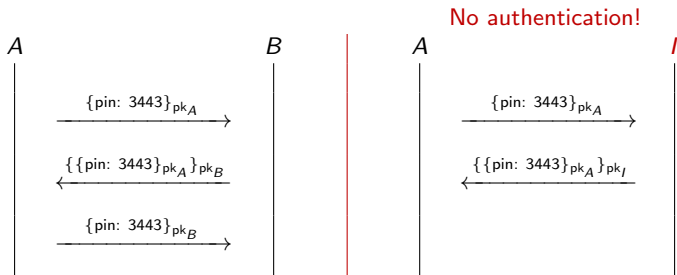
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA



since $\{\{\text{pin: 3443}\}_{pk_A}\}_{pk_B} = \{\{\text{pin: 3443}\}_{pk_B}\}_{pk_A}$ by commutativity

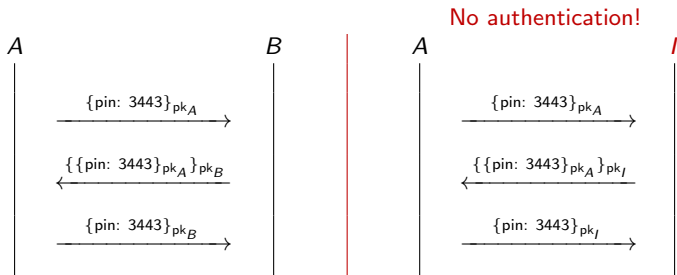
Example of a logical attack

Assume a commutative symmetric encryption scheme

$$\{\{m\}_{k_1}\}_{k_2} = \{\{m\}_{k_2}\}_{k_1}$$

where $\{m\}_k$ denotes the encryption of message m under the key k

Example: RSA



since $\{\{\text{pin: 3443}\}_{pk_A}\}_{pk_B} = \{\{\text{pin: 3443}\}_{pk_B}\}_{pk_A}$ by commutativity

Needham-Schroeder Public Key (NSPK)

NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

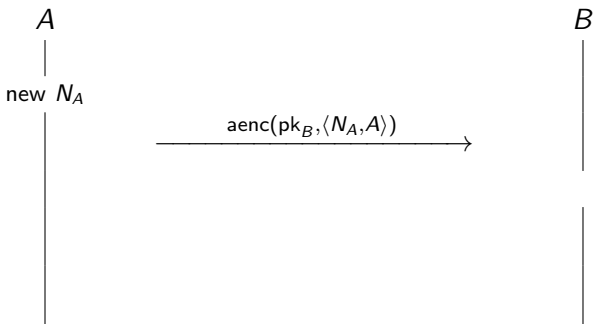
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

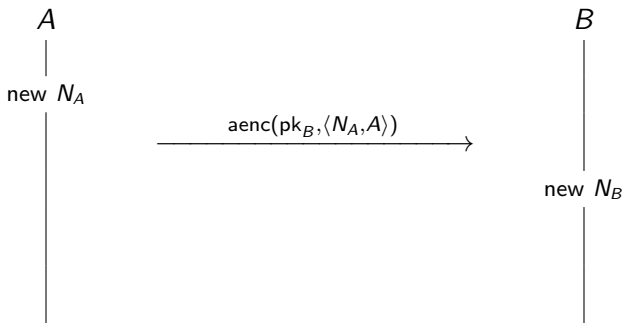
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

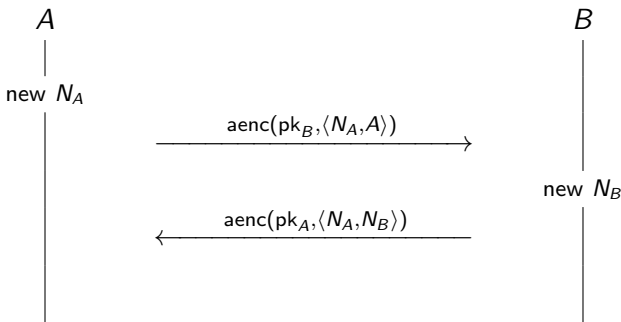
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

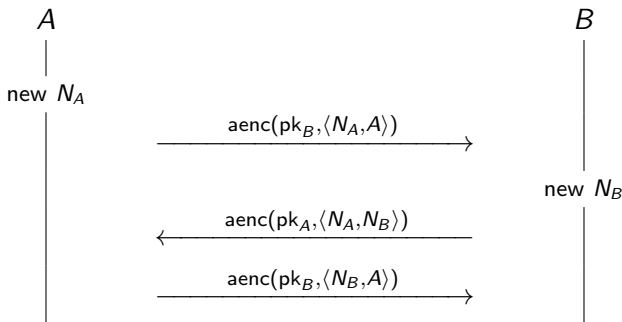
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

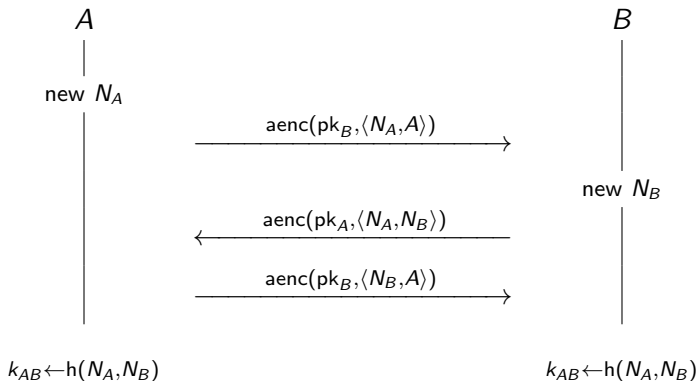
NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

Needham-Schroeder Public Key (NSPK)

NSPK: authentication and key agreement protocol



[N. Roger, M. Schroeder, Michael. "Using encryption for authentication in large networks of computers". Communications of the ACM (December 1978)]

NSPK: security requirements

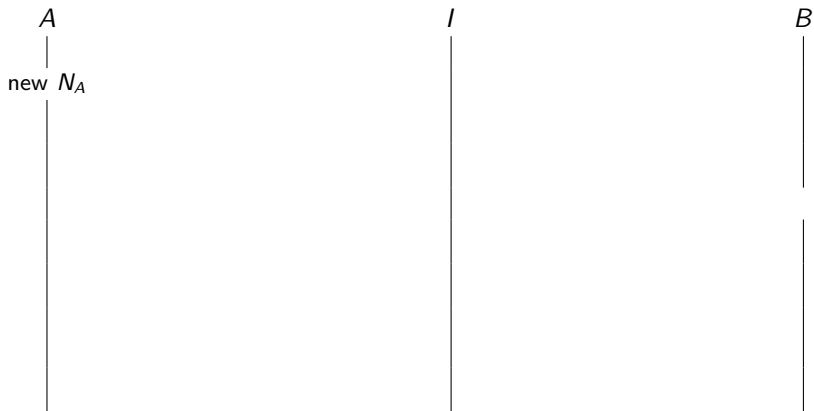
- ▶ **Authentication:** if Alice has completed the protocol, apparently with Bob, then Bob must also have completed the protocol with Alice.
- ▶ **Authentication:** If Bob has completed the protocol, apparently with Alice, then Alice must have completed the protocol with Bob.
- ▶ **Confidentiality:** Messages sent encrypted with the agreed key ($k \leftarrow h(N_A, N_B)$) remain secret.

NSPK: Lowe's attack on authentication

Attack found 17 years after the publication of the NS protocol!!

NSPK: Lowe's attack on authentication

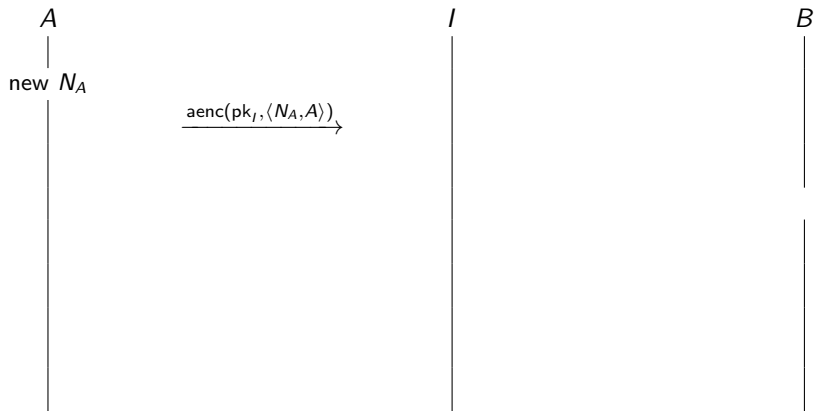
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

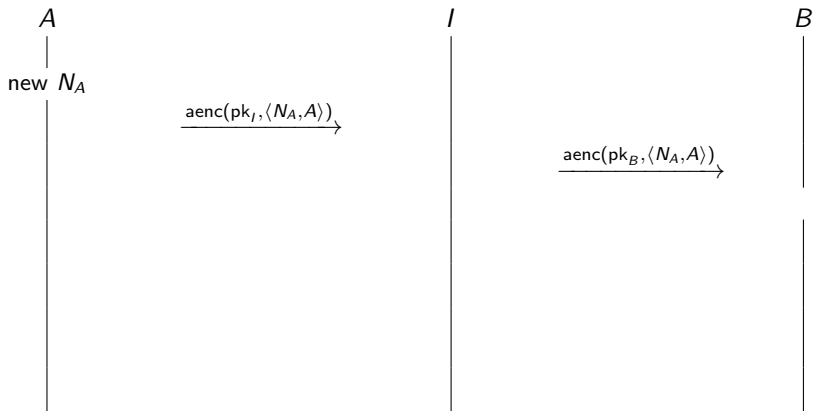
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

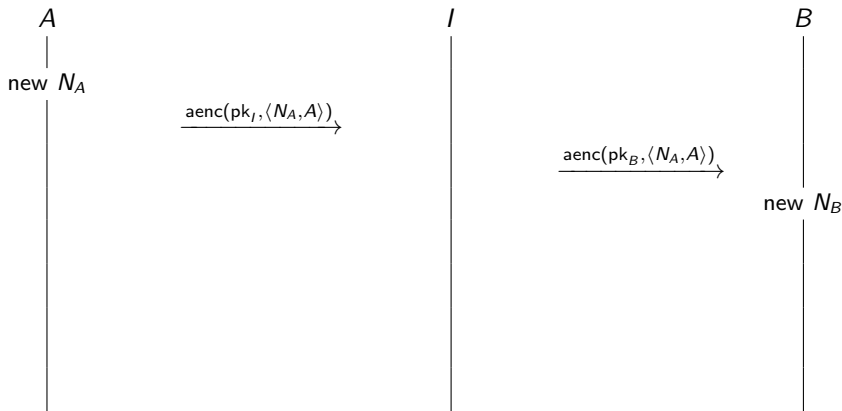
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

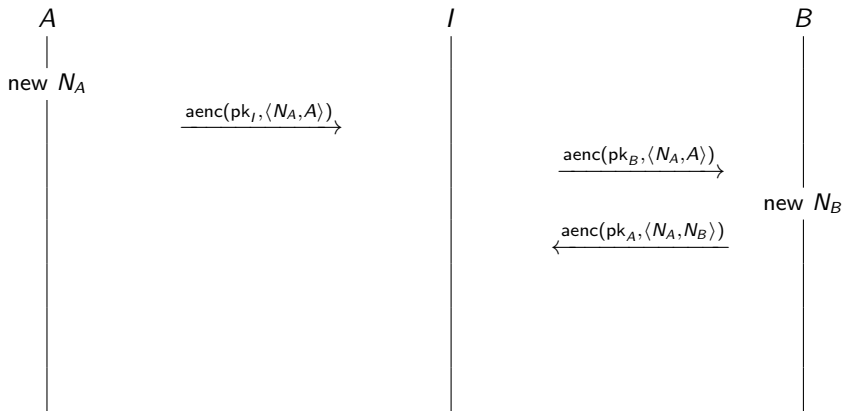
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

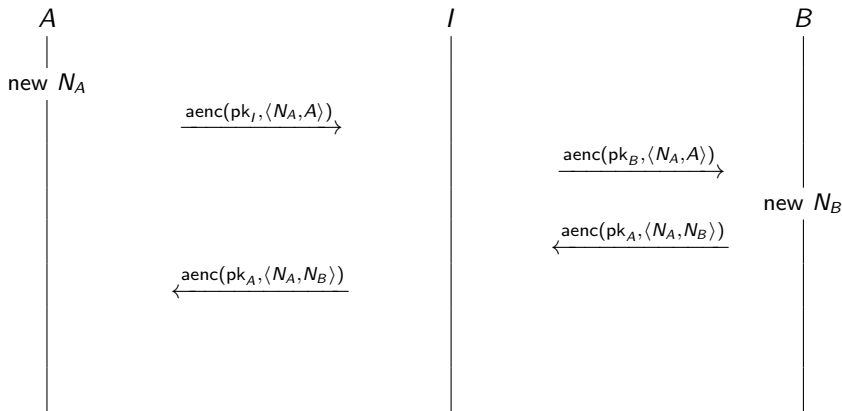
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

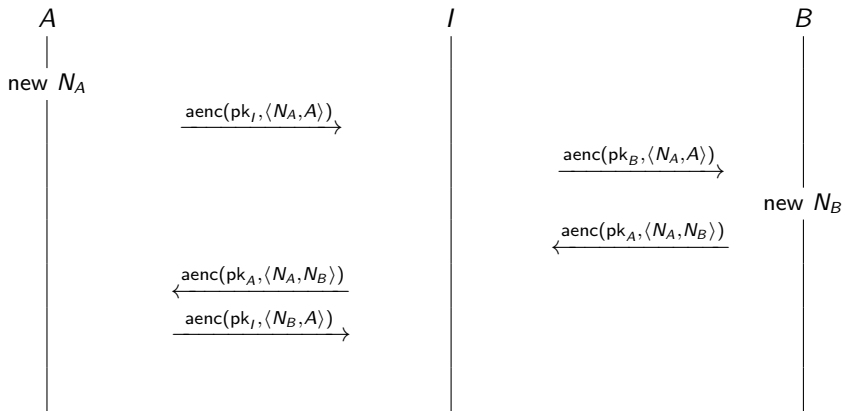
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

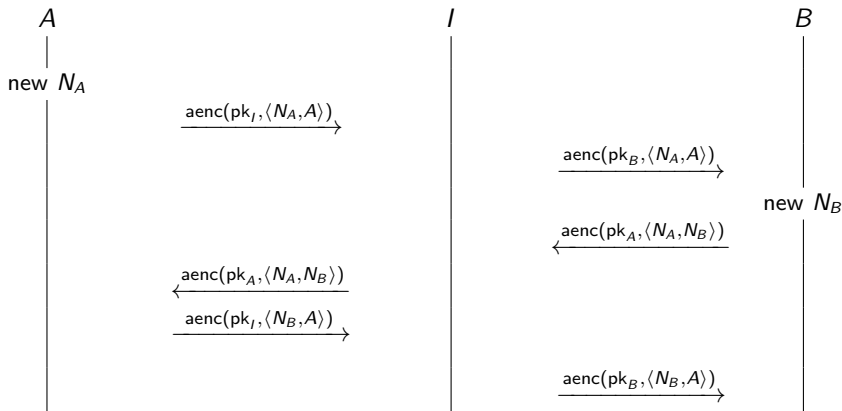
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's attack on authentication

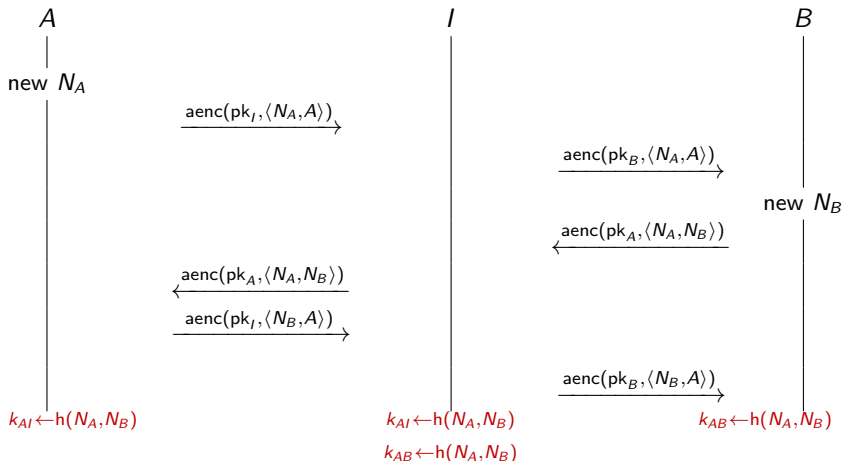
Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

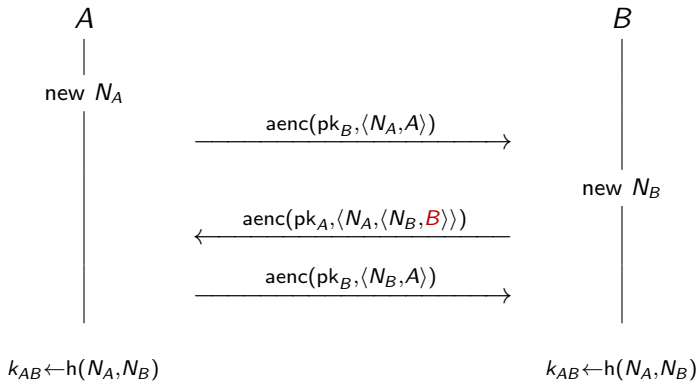
NSPK: Lowe's attack on authentication

Attack found 17 years after the publication of the NS protocol!!



[G. Lowe. "An attack on the Needham-Schroeder public key authentication protocol". Information Processing Letters (November 1995)]

NSPK: Lowe's fix



Public Key Kerberos PKINIT-26 (very abstract)

Goals: client authentication, key agreement, TGT delivery

- ▶ $\{m\}_k^s$: message m symmetrically encrypted under key k
- ▶ $\{m\}_k^a$: message m asymmetrically encrypted under key k
- ▶ $[m]_k$: message m digitally signed with key k
- ▶ t_C, t_K : timestamps
- ▶ $TGT = \{AK, C, t_K\}_{k_T}^s$

Public Key Kerberos PKINIT-26 (very abstract)

Goals: client authentication, key agreement, TGT delivery



- ▶ $\{m\}_k^s$: message m symmetrically encrypted under key k
- ▶ $\{m\}_k^a$: message m asymmetrically encrypted under key k
- ▶ $[m]_k$: message m digitally signed with key k
- ▶ t_C, t_K : timestamps
- ▶ $TGT = \{AK, C, t_K\}_{k_T}^s$

Public Key Kerberos PKINIT-26 (very abstract)

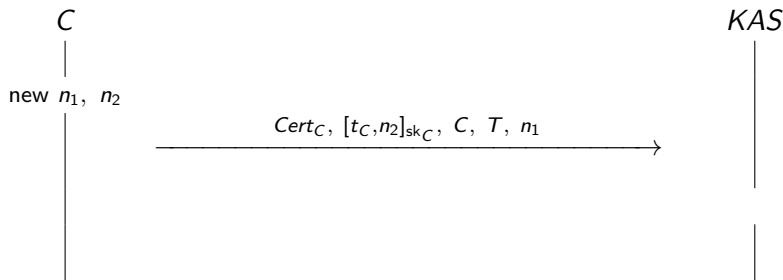
Goals: client authentication, key agreement, TGT delivery



- ▶ $\{m\}_k^s$: message m symmetrically encrypted under key k
- ▶ $\{m\}_k^a$: message m asymmetrically encrypted under key k
- ▶ $[m]_k$: message m digitally signed with key k
- ▶ t_C, t_K : timestamps
- ▶ $TGT = \{AK, C, t_K\}_{k_T}^s$

Public Key Kerberos PKINIT-26 (very abstract)

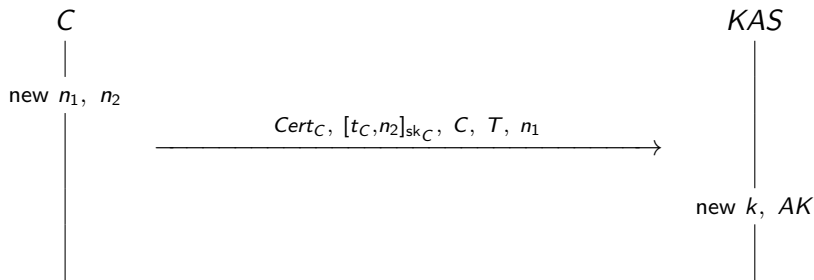
Goals: client authentication, key agreement, TGT delivery



- ▶ $\{m\}_k^s$: message m symmetrically encrypted under key k
- ▶ $\{m\}_k^a$: message m asymmetrically encrypted under key k
- ▶ $[m]_k$: message m digitally signed with key k
- ▶ t_C, t_K : timestamps
- ▶ $TGT = \{AK, C, t_K\}_{k_T}^s$

Public Key Kerberos PKINIT-26 (very abstract)

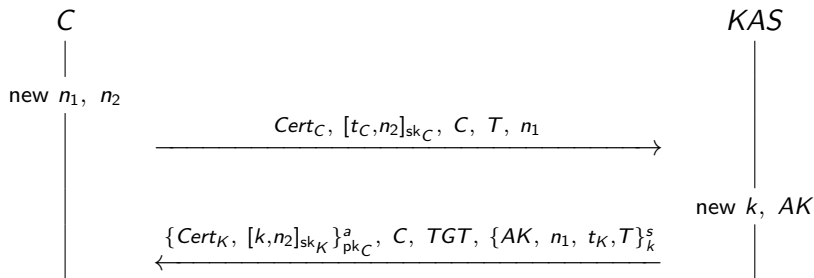
Goals: client authentication, key agreement, TGT delivery



- ▶ $\{m\}_k^s$: message m symmetrically encrypted under key k
- ▶ $\{m\}_k^a$: message m asymmetrically encrypted under key k
- ▶ $[m]_k$: message m digitally signed with key k
- ▶ t_C, t_K : timestamps
- ▶ $TGT = \{AK, C, t_K\}_{k_T}^s$

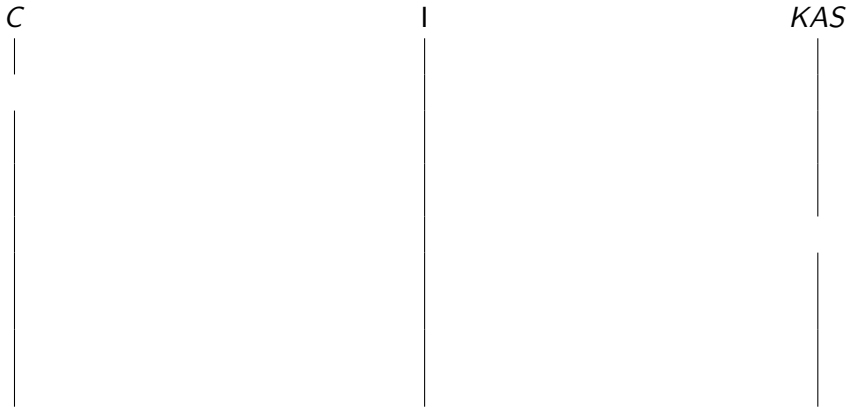
Public Key Kerberos PKINIT-26 (very abstract)

Goals: client authentication, key agreement, TGT delivery



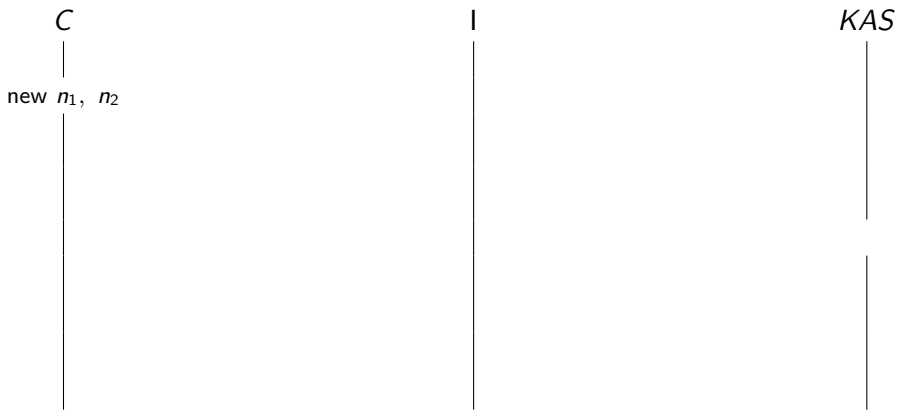
- ▶ $\{m\}_k^s$: message m symmetrically encrypted under key k
- ▶ $\{m\}_k^a$: message m asymmetrically encrypted under key k
- ▶ $[m]_k$: message m digitally signed with key k
- ▶ t_C, t_K : timestamps
- ▶ $TGT = \{AK, C, t_K\}_{k_T}^s$

PKINIT-26: attack



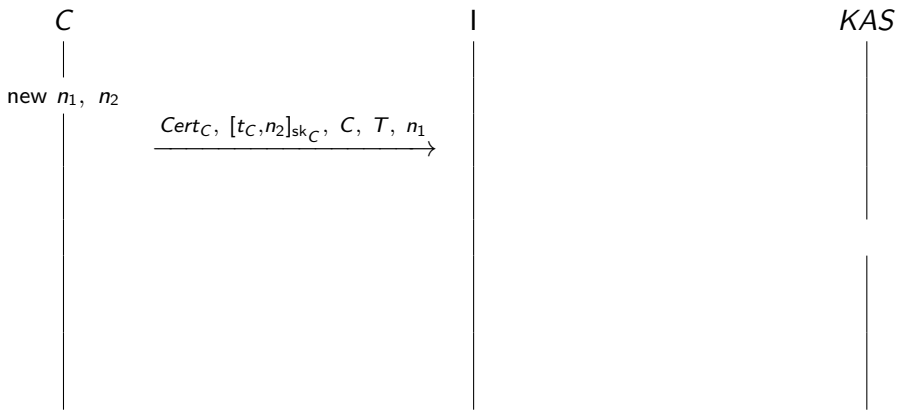
[I. Cervesato, A. D. Jaggard, A. Scedrov, J. Tsay, C. Walstad. "Breaking and Fixing Public-Key Kerberos". (ASIAN'06)]

PKINIT-26: attack



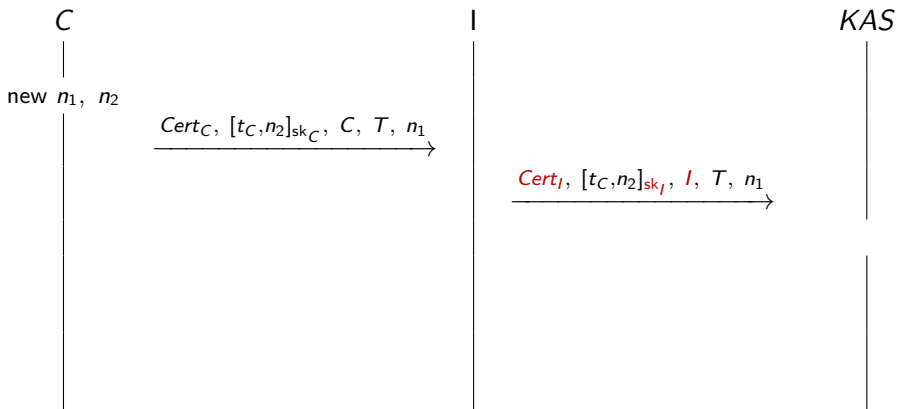
[I. Cervesato, A. D. Jaggard, A. Scedrov, J. Tsay, C. Walstad. "Breaking and Fixing Public-Key Kerberos". (ASIAN'06)]

PKINIT-26: attack



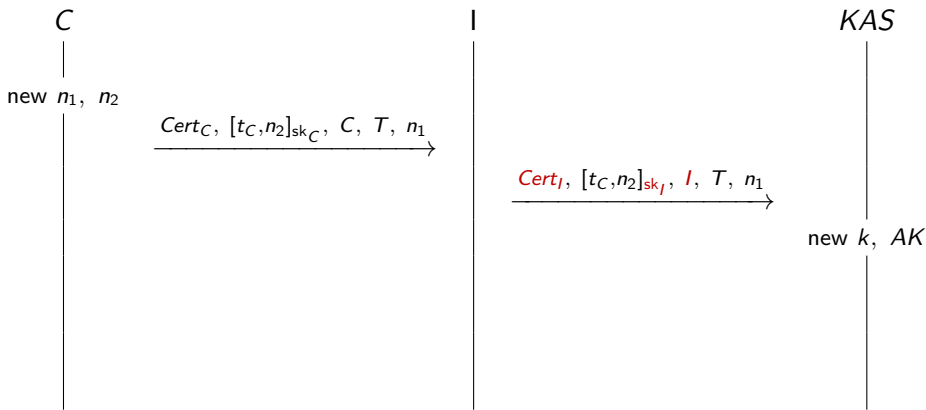
[I. Cervesato, A. D. Jaggard, A. Scedrov, J. Tsay, C. Walstad. "Breaking and Fixing Public-Key Kerberos". (ASIAN'06)]

PKINIT-26: attack



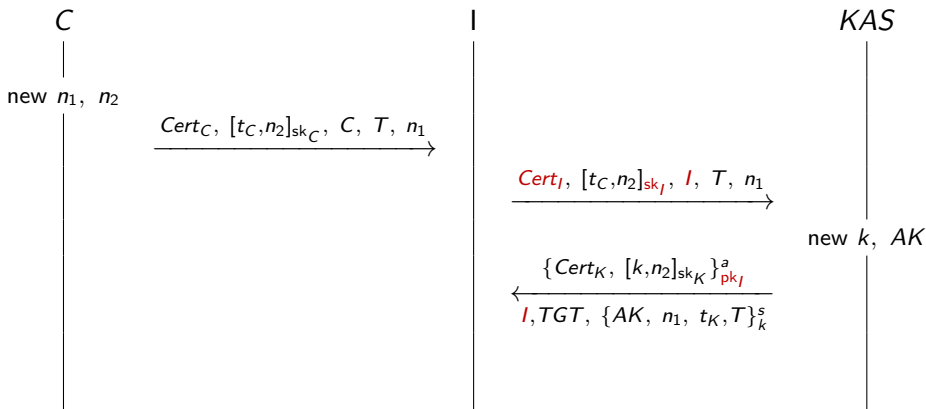
[I. Cervesato, A. D. Jaggard, A. Scedrov, J. Tsay, C. Walstad. "Breaking and Fixing Public-Key Kerberos". (ASIAN'06)]

PKINIT-26: attack



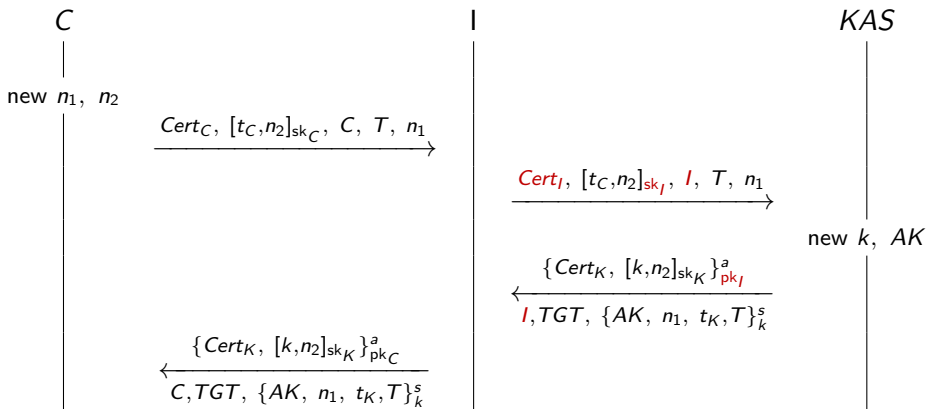
[I. Cervesato, A. D. Jaggard, A. Scedrov, J. Tsay, C. Walstad. "Breaking and Fixing Public-Key Kerberos". (ASIAN'06)]

PKINIT-26: attack



[I. Cervesato, A. D. Jaggard, A. Scedrov, J. Tsay, C. Walstad. "Breaking and Fixing Public-Key Kerberos". (ASIAN'06)]

PKINIT-26: attack



[I. Cervesato, A. D. Jaggard, A. Scedrov, J. Tsay, C. Walstad. "Breaking and Fixing Public-Key Kerberos". (ASIAN'06)]