Classical Protocols
Needham-Schroeder Public-Key Protocol

Design and Verification of Security Protocols and Security Ceremonies

Programa de Pós-Graduacão em Ciências da Computacão
Dr. Jean Everson Martina

August-November 2016

UNIVERSIDADE FEDERAL
DE SANTA CATARINA

Needham-Schroeder Public Key Protocol

- Many existing protocols are derived from the work of Needham and Schroeder (1978);

Needham-Schroeder Public Key Protocol

- Many existing protocols are derived from the work of Needham and Schroeder (1978);
- The handshake done by many public key cryptographic protocols are derived from NSPKP;

Needham-Schroeder Public Key Protocol

- Many existing protocols are derived from the work of Needham and Schroeder (1978);
- The handshake done by many public key cryptographic protocols are derived from NSPKP;
- NSPKP is a public-key authentication protocol designed to generate and propagate a session key which is used for subsequent symmetrically encrypted communication;

Needham-Schroeder Public Key Protocol

- Many existing protocols are derived from the work of Needham and Schroeder (1978);
- The handshake done by many public key cryptographic protocols are derived from NSPKP;
- NSPKP is a public-key authentication protocol designed to generate and propagate a session key which is used for subsequent symmetrically encrypted communication;
- There is no public key infrastructure in place, but the identities related top public keys are an assumption.

Needham-Schroeder Public Key Protocol

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$

Needham-Schroeder Public Key Protocol

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$

Needham-Schroeder Public Key Protocol

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
3. A $\rightarrow$ B: $\{|N_b|\}_{K_b}$

NSPKP Goals

- The goal of the protocol is to establish mutual authentication between two parties A and B in the presence of adversary;

NSPKP Goals

- The goal of the protocol is to establish mutual authentication between two parties A and B in the presence of adversary;
- A and B obtain a secret shared key though direct communication using public key cryptography;

NSPKP Goals

- The goal of the protocol is to establish mutual authentication between two parties A and B in the presence of adversary;
- A and B obtain a secret shared key though direct communication using public key cryptography;
- This adversary can intercept messages, delay messages, read and copy messages and generate messages;

NSPKP Goals

- The goal of the protocol is to establish mutual authentication between two parties A and B in the presence of adversary;
- A and B obtain a secret shared key though direct communication using public key cryptography;
- This adversary can intercept messages, delay messages, read and copy messages and generate messages;
- This adversary can not learn the privates keys of principals.

Freshness Concepts of NSPKP

- NSPKP uses nonces which are randomly generated values included in messages and used only once;

Freshness Concepts of NSPKP

- NSPKP uses nonces which are randomly generated values included in messages and used only once;
- If a nonce is generated and sent by one agent in one step and returned by another in a later step, the generator knows that the message is fresh and not a replay from an earlier exchange;

Freshness Concepts of NSPKP

- NSPKP uses nonces which are randomly generated values included in messages and used only once;
- If a nonce is generated and sent by one agent in one step and returned by another in a later step, the generator knows that the message is fresh and not a replay from an earlier exchange;
- Note that a nonce is not anchored in time. The only assumption is that it has not been used in any earlier interchange, with high probability because it is random and not used twice.

Needham-Schroeder Public Key Protocol - Questions

- What are the assumptions?

Needham-Schroeder Public Key Protocol - Questions

- What are the assumptions?
- What seems to be the goal?

Needham-Schroeder Public Key Protocol - Questions

- What are the assumptions?
- What seems to be the goal?
- What might the principals believe after each step?

Needham-Schroeder Public Key Protocol - Questions

- What are the assumptions?
- What seems to be the goal?
- What might the principals believe after each step?
- What is missing?

Needham-Schroeder Public Key Protocol - Questions

- What are the assumptions?
- What seems to be the goal?
- What might the principals believe after each step?
- What is missing?
- Is it secure?

Needham-Schroeder Public Key Protocol Description

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Alice send to Bob an encrypted nonce

Needham-Schroeder Public Key Protocol Description

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Alice send to Bob an encrypted nonce
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
   Bob returns to Alice her Nonce together with his own

Needham-Schroeder Public Key Protocol Description

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Alice send to Bob an encrypted nonce
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
   Bob returns to Alice her Nonce together with his own
3. A $\rightarrow$ B: $\{|N_b|\}_{K_b}$
   Alice returns to Bob his Nonce

Needham-Schroeder Public Key Protocol Assumptions

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Only Alice knows Na before message 1
   Only Bob can decrypt message 1

Needham-Schroeder Public Key Protocol Assumptions

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Only Alice knows Na before message 1
   Only Bob can decrypt message 1
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
   Only Bob knows Nb before message 2
   Bob knows Na because he can decrypt
   Ohly Alice can decrypt message 2

Needham-Schroeder Public Key Protocol Assumptions

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Only Alice knows Na before message 1
   Only Bob can decrypt message 1
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
   Only Bob knows Nb before message 2
   Bob knows Na because he can decrypt
   Ohly Alice can decrypt message 2
3. A $\rightarrow$ B: $\{|N_b|\}_{K_b}$
   Alice knows Nb because she can decrypt
   Only Bob can decrypt message 3

Needham-Schroeder Public Key Protocol Assumptions

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Only Alice knows Na before message 1
   Only Bob can decrypt message 1
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
   Only Bob knows Nb before message 2
   Bob knows Na because he can decrypt
   Ohly Alice can decrypt message 2
3. A $\rightarrow$ B: $\{|N_b|\}_{K_b}$
   Alice knows Nb because she can decrypt
   Only Bob can decrypt message 3
   Why do we need message 3?

Needham-Schroeder Public Key Protocol Interpretation

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Alice starts a session Na is the session control
   Alice's identity (A) is for Bob to know whom
   to encrypt message 2

Needham-Schroeder Public Key Protocol Interpretation

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Alice starts a session Na is the session control
   Alice's identity (A) is for Bob to know whom
   to encrypt message 2
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
   Bom sens Na back to keep the session
   Bob uses Nb to authenticate Alice on message 3
   When receiving message 2 Alice knows that only Bob could have created it because it contains Na

Needham-Schroeder Public Key Protocol Interpretation

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
   Alice starts a session Na is the session control
   Alice's identity (A) is for Bob to know whom
   to encrypt message 2
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
   Bom sens Na back to keep the session
   Bob uses Nb to authenticate Alice on message 3
   When receiving message 2 Alice knows that only Bob could
   have created it because it contains Na
3. A $\rightarrow$ B: $\{|N_b|\}_{K_b}$
   Alice already authenticated Bob. Now she wants to authenticat
   When receiving message 3 Bob knows that only Alice could
   have created it because it contains Nb

Needham-Schroeder Public Key Protocol Objectives

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
3. A $\rightarrow$ B: $\{|N_b|\}_{K_b}$

- The protocols authenticates Alice to Bob;

Needham-Schroeder Public Key Protocol Objectives

1. $A \rightarrow B: \{|N_a, A|\}_{K_b}$
2. $B \rightarrow A: \{|N_a, N_b|\}_{K_a}$
3. $A \rightarrow B: \{|N_b|\}_{K_b}$

- The protocols authenticates Alice to Bob;
- The protocols authenticate Bob to Alice;

Needham-Schroeder Public Key Protocol Objectives

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
3. A $\rightarrow$ B: $\{|N_b|\}_{K_b}$

- The protocols authenticates Alice to Bob;
- The protocols authenticate Bob to Alice;
- By the usage of fresh Nonces, we obtain the aliveness property in the protocols;

Needham-Schroeder Public Key Protocol Objectives

1. A $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2. B $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
3. A $\rightarrow$ B: $\{|N_b|\}_{K_b}$

- The protocols authenticates Alice to Bob;
- The protocols authenticate Bob to Alice;
- By the usage of fresh Nonces, we obtain the aliveness property in the protocols;
- But, Is it secure?

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:
  - Intercept anything in the network;

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:
  - Intercept anything in the network;
  - Block anything in the network;

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:
  - Intercept anything in the network;
  - Block anything in the network;
  - Replay Messages;

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:
  - Intercept anything in the network;
  - Block anything in the network;
  - Replay Messages;
  - Create new messages with what he learned with the traffic;

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:
  - Intercept anything in the network;
  - Block anything in the network;
  - Replay Messages;
  - Create new messages with what he learned with the traffic;
  - Behave as a normal agent;

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:
  - Intercept anything in the network;
  - Block anything in the network;
  - Replay Messages;
  - Create new messages with what he learned with the traffic;
  - Behave as a normal agent;
- He can not:

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:
  - Intercept anything in the network;
  - Block anything in the network;
  - Replay Messages;
  - Create new messages with what he learned with the traffic;
  - Behave as a normal agent;
- He can not:
  - Break cryptography;

Almighty Attack - Rules of the Game

- Charlie is a very powerful attacker.
- He can:
  - Intercept anything in the network;
  - Block anything in the network;
  - Replay Messages;
  - Create new messages with what he learned with the traffic;
  - Behave as a normal agent;
- He can not:
  - Break cryptography;
  - Guess random numbers.

Almighty Attack - Lowe's Attack

1. A $\rightarrow$ C: $\{|N_a, A|\}_{K_c}$

Almighty Attack - Lowe's Attack

1. A $\rightarrow$ C: $\{|N_a, A|\}_{K_c}$
1'. C(A) $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2'. B $\rightarrow$ C(A): $\{|N_a, N_b|\}_{K_a}$

Almighty Attack - Lowe's Attack

1. A $\rightarrow$ C: $\{|N_a, A|\}_{K_c}$
1'. C(A) $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2'. B $\rightarrow$ C(A): $\{|N_a, N_b|\}_{K_a}$
2. C $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$

Almighty Attack - Lowe's Attack

1. A $\rightarrow$ C: $\{|N_a, A|\}_{K_c}$
1'. C(A) $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2'. B $\rightarrow$ C(A): $\{|N_a, N_b|\}_{K_a}$
2. C $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
3. A $\rightarrow$ C: $\{|N_b|\}_{K_c}$

Almighty Attack - Lowe's Attack

1. A $\rightarrow$ C: $\{|N_a, A|\}_{K_c}$
1'. C(A) $\rightarrow$ B: $\{|N_a, A|\}_{K_b}$
2'. B $\rightarrow$ C(A): $\{|N_a, N_b|\}_{K_a}$
2. C $\rightarrow$ A: $\{|N_a, N_b|\}_{K_a}$
3. A $\rightarrow$ C: $\{|N_b|\}_{K_c}$
3'. C $\rightarrow$ B: $\{|N_b|\}_{K_b}$

- Bob believes to be talking to Alice , while he is talking to Charlie;

Almighty Attack - Lowe's Attack

1. $A \rightarrow C: \{|N_a, A|\}_{K_c}$
1'. $C(A) \rightarrow B: \{|N_a, A|\}_{K_b}$
2'. $B \rightarrow C(A): \{|N_a, N_b|\}_{K_a}$
2. $C \rightarrow A: \{|N_a, N_b|\}_{K_a}$
3. $A \rightarrow C: \{|N_b|\}_{K_c}$
3'. $C \rightarrow B: \{|N_b|\}_{K_b}$

- Bob believes to be talking to Alice , while he is talking to Charlie;
- Charlie uses Alice as an oracle to answers Bob's challenges;

Almighty Attack - Lowe's Attack

1. $A \rightarrow C: \{|N_a, A|\}_{K_c}$
1'. $C(A) \rightarrow B: \{|N_a, A|\}_{K_b}$
2'. $B \rightarrow C(A): \{|N_a, N_b|\}_{K_a}$
2. $C \rightarrow A: \{|N_a, N_b|\}_{K_a}$
3. $A \rightarrow C: \{|N_b|\}_{K_c}$
3'. $C \rightarrow B: \{|N_b|\}_{K_b}$

- Bob believes to be talking to Alice , while he is talking to Charlie;
- Charlie uses Alice as an oracle to answers Bob's challenges;
- Charlie can use Nb to prove to Bob he is Alice.

Lowe's Attack - Facts

- Gavin Lowe was a random Computer Theoretician at Oxford;

Lowe's Attack - Facts

- Gavin Lowe was a random Computer Theoretician at Oxford;
- The attack looks easy, but it took 15 years to be found;

Lowe's Attack - Facts

- Gavin Lowe was a random Computer Theoretician at Oxford;
- The attack looks easy, but it took 15 years to be found;
- The attack works because of a change on the threat model;

Lowe's Attack - Facts

- Gavin Lowe was a random Computer Theoretician at Oxford;
- The attack looks easy, but it took 15 years to be found;
- The attack works because of a change on the threat model;
- But his attack is important because it was only discovered with the help of a formal verification tool.

How Lowe did it?

- He used a tool called a model checker;

How Lowe did it?

- He used a tool called a model checker;
- This tool has able to saturate the protocol execution;

How Lowe did it?

- He used a tool called a model checker;
- This tool has able to saturate the protocol execution;
- The idea is to explore all the reachable states of the model;

How Lowe did it?

- He used a tool called a model checker;
- This tool has able to saturate the protocol execution;
- The idea is to explore all the reachable states of the model;
- This verification is considered bound to the amount of peers and parallel runs tested.

What we learned from this example?

- Some tests are too hard to be executed by hand;

What we learned from this example?

- Some tests are too hard to be executed by hand;
- Theoretical tool are a good breakthrough for any area;

What we learned from this example?

- Some tests are too hard to be executed by hand;
- Theoretical tool are a good breakthrough for any area;
- Event very simple and well studied protocols may contain hidden failures;

What we learned from this example?

- Some tests are too hard to be executed by hand;
- Theoretical tool are a good breakthrough for any area;
- Event very simple and well studied protocols may contain hidden failures;
- We learned to be diligent and somewhat paranoid on protocols and how they achieve their goals.

Discussion

- Which other protocols do you believe there are failures that were not detected yet?

Discussion

- Which other protocols do you believe there are failures that were not detected yet?
- How do we correct Lowe's attack on NSPKP?

Discussion

- Which other protocols do you believe there are failures that were not detected yet?
- How do we correct Lowe's attack on NSPKP?
- Is formally prove a protocol enough to claim it is secure?

Discussion

- Which other protocols do you believe there are failures that were not detected yet?
- How do we correct Lowe's attack on NSPKP?
- Is formally prove a protocol enough to claim it is secure?
- What if a user drop an assumption of the protocol? Is it still secure?

Discussion

- Which other protocols do you believe there are failures that were not detected yet?
- How do we correct Lowe's attack on NSPKP?
- Is formally prove a protocol enough to claim it is secure?
- What if a user drop an assumption of the protocol? Is it still secure?
- How secure is formally secure?

Questions????

UNIVERSIDADE FEDERAL
DE SANTA CATARINA

UNIVERSIDADE FEDERAL
DE SANTA CATARINA