Secure Composition of Secure Protocols

Composing Protocols in a Secure Way

Giuseppe Persiano

Dipartimento di Informatica ed Appl.

Università di Salerno

Italy

`http://www.dia.unisa.it/~giuper`

Work supported by EU IP – Aeolus

AEOLUS

Algorithmic Principles for Building Efficient Overlay Computers

Security in AEOLUS

- **WP4.1: Trust Management**

  Policy specification

  Efficient Compliance Checking Algorithm

  Game Theoretic Techniques for Authorization

Security in AEOLUS

- **WP4.1: Trust Management**

  Policy specification

  Efficient Compliance Checking Algorithm

  Game Theoretic Techniques for Authorization

- **WP4.2: Privacy, indentity and anonymity**

  Anonymous communication and transactions

Security in AEOLUS

- **WP4.1: Trust Management**

  Policy specification

  Efficient Compliance Checking Algorithm

  Game Theoretic Techniques for Authorization

- **WP4.2: Privacy, indentity and anonymity**

  Anonymous communication and transactions

- **WP4.3: Secure distributed computation**

  Secure protocols in Global scenario:

  Concurrency and Non-Malleability

Security in AEOLUS

- **WP4.1: Trust Management**

  Policy specification

  Efficient Compliance Checking Algorithm

  Game Theoretic Techniques for Authorization

- **WP4.2: Privacy, indentity and anonymity**

  Anonymous communication and transactions

- **WP4.3: Secure distributed computation**

  Secure protocols in Global scenario:

Concurrency and Non-Malleability

A Simple Scenario

- Alice, Bob and Charles are competing in an auction.
- The auctioneer publishes an RSA key (N, e).
- Each bidder sends his offer by e-mail encrypted with the RSA key (N, e) of the auctioneer.

A Simple Scenario

- Alice, Bob and Charles are competing in an auction.
- The auctioneer publishes an RSA key (N, e).
- Each bidder sends his offer by e-mail encrypted with the RSA key (N, e) of the auctioneer.
- Alice sees Bob's encrypted bid B and computes her encrypted bid A as $A = B \cdot E(2)$.

A Simple Scenario

- Alice, Bob and Charles are competing in an auction.
- The auctioneer publishes an RSA key (N, e).
- Each bidder sends his offer by e-mail encrypted with the RSA key (N, e) of the auctioneer.
- Alice sees Bob's encrypted bid B and computes her encrypted bid A as $A = B \cdot E(2)$.

Bob will never win!!!

A Simple Problem

An RSA public key (N, e) is known to Alice and Bob.

Alice encrypts m by computing $C = E(m)$ and wants to convince Bob that she knows the cleartex m associated with ciphertex $C = E(m)$.

A Simple Protocol

[1. Alice]

pick r **at random;**

compute $H = E(r)$**;**

$a^0 \leftarrow r$; $a^1 \leftarrow r \cdot m$

send H **to Bob.**

A Simple Protocol

[1. Alice]

pick r **at random;**

compute $H = E(r)$**;**

$a^0 \leftarrow r$**;** $a^1 \leftarrow r \cdot m$

send H **to Bob.**

[2. Bob]

pick $b \leftarrow \{0, 1\}$ **at random;**

send b **to Alice.**

A Simple Protocol

[1. Alice]

pick r at random;

compute $H = E(r)$;

$a^0 \leftarrow r$; $a^1 \leftarrow r \cdot m$

send H to Bob.

[2. Bob]

pick $b \leftarrow \{0, 1\}$ at random;

send b to Alice.

[3. Alice]

send a^b to Bob.

A Simple Protocol

[1. Alice]

pick r **at random;**

compute $H = E(r)$**;**

$a^0 \leftarrow r$**;** $a^1 \leftarrow r \cdot m$

send H **to Bob.**

[2. Bob]

pick $b \leftarrow \{0, 1\}$ **at random;**

send b **to Alice.**

[3. Alice]

send a^b **to Bob.**

[4. Bob]

if $b = 0$ **verify** $E(a^0) = H$**;**

if $b = 1$ **verify** $E(a^1) = H \cdot C$**;**

A Simple Protocol

[1. Alice]

pick r at random;
compute $H = E(r)$;
$a^0 \leftarrow r$; $a^1 \leftarrow r \cdot m$
send H to Bob.

[2. Bob] $\xrightarrow{\mathbf{H}}$

pick $b \leftarrow \{0, 1\}$ at random; $\xleftarrow{\mathbf{b}}$
send b to Alice. $\xrightarrow{\mathbf{a}^b}$

[3. Alice]

send a^b to Bob.

[4. Bob]

if $b = 0$ verify $E(a^0) = H$;
if $b = 1$ verify $E(a^1) = H \cdot C$;

Is This A Solution?

Alice cannot cheat.

Bob does not learn anything about m.

Is This A Solution?

Alice cannot cheat.

Suppose Alice does **not** know m.

Then for each H Alice knows at most one of a^0 or a^1.

Alice is caught with probability $\geq 1/2$.

Bob does not learn anything about m.

The Simulation Paradigm [GMR]

For **each** possible strategy of Bob, there exists an efficient algorithm S (**simulator**) such that S on input C (**without** knowing m) produces, in expected polynomial time, the same view of Bob.

The Simulation Paradigm [GMR]

For each possible strategy of Bob, there exists an efficient algorithm S (simulator) such that S on input C (without knowing m) produces, in expected polynomial time, the same view of Bob.

[1.] pick $\tilde{b} \leftarrow \{0, 1\}$ at random;
pick r at random;
if $\tilde{b} = 0$ compute $H = E(r)$ and $a^0 = r$, $a^1 =?$;
if $\tilde{b} = 1$ compute $H = C^{-1} \cdot E(r)$ and $a^0 =?$, $a^1 = r$;
send H to Bob.

The Simulation Paradigm [GMR]

For each possible strategy of Bob, there exists an efficient algorithm S (simulator) such that S on input C (without knowing m) produces, in expected polynomial time, the same view of Bob.

[1.] pick $\tilde{b} \leftarrow \{0, 1\}$ at random;
pick r at random;
if $\tilde{b} = 0$ compute $H = E(r)$ and $a^0 = r$, $a^1 =?$;
if $\tilde{b} = 1$ compute $H = C^{-1} \cdot E(r)$ and $a^0 =?$, $a^1 = r$;
send H to Bob.

[2.] receive b from Bob;

The Simulation Paradigm [GMR]

For each possible strategy of Bob, there exists an efficient algorithm S (simulator) such that S on input C (without knowing m) produces, in expected polynomial time, the same view of Bob.

[1.] pick $\tilde{b} \leftarrow \{0,1\}$ at random;
pick r at random;
if $\tilde{b} = 0$ compute $H = E(r)$ and $a^0 = r$, $a^1 = ?$;
if $\tilde{b} = 1$ compute $H = C^{-1} \cdot E(r)$ and $a^0 = ?$, $a^1 = r$;
send H to Bob.

[2.] receive b from Bob;

[3.] if $\tilde{b} = b$ **Output:** (H, b, a^b) else **GOTO** 1;

Reducing Probability of Cheating

Solution: repeat $k(= 50)$ times sequentially.

$$\begin{array}{ccccc} \xrightarrow{\mathbf{H}_1} & \vdots & \vdots & \vdots & \xrightarrow{\mathbf{H}_k} \\ \xleftarrow{\mathbf{b}_1} & \vdots & \vdots & \vdots & \xleftarrow{\mathbf{b}_k} \\ \xrightarrow{\mathbf{A}^{b_1}} & \vdots & \vdots & \vdots & \xrightarrow{\mathbf{A}^{b_k}} \end{array}$$

Probability of cheating is at most 2^{-k}.

Good news: Bob's security is preserved.

Good news: Alice's security is preserved by sequential composition.

Bad news: Sequential composition uses $O(k)$ messages.

Parallel Composition

$$\xrightarrow{\mathbf{H_1,\cdots,H_k}}$$

$$\xleftarrow{\mathbf{b_1,\cdots,b_k}}$$

$$\xrightarrow{\mathbf{A_1^{b_1},\cdots,A_k^{b_k}}}$$

WOW: 3 **messages.**

Bad news: **it is not secure.**

Intuition: **to complete simulation,** S **has to guess** $\mathbf{b_1,\cdots,b_k}$ **correctly.**

Can be done in 4 rounds

Security is not preserved under parallel composition.

The Global Computing Scenario

In a Global Computing scenario:

- **Alice is interacting with n players (not just one).**
- **Alice is acting as prover and as a verifier.**
- **The communication is asynchronous.**
- **Messages from different sessions can interleave arbitrarily.**
- **No central coordination mechanism exists.**

A Global Computing Scenario

$$
\begin{array}{ccccccc}
\mathbf{V}_1 & & \mathbf{V}_2 & & \cdots & \mathbf{V}_n & \\
& \xrightarrow{\mathbf{H}^1} & & & & & \\
& & & \xrightarrow{\mathbf{H}^2} & & & \\
& & & & & & \xrightarrow{\mathbf{H}^n} \\
& & & & & & \xleftarrow{\mathbf{b}^n} \\
& & & & & & \xrightarrow{\mathbf{A}^n} \\
& & & \xleftarrow{\mathbf{b}^2} & & & \\
& & & \xrightarrow{\mathbf{A}^2} & & & \\
& \xleftarrow{\mathbf{b}^1} & & & & & \\
& \xrightarrow{\mathbf{A}^1} & & & & &
\end{array}
$$

Concurrent Composition - State of the Art

1. Essentially $O(\log n)$ rounds are sufficient. Canetti et al., 2001
2. Constant or quasi constant rounds are sufficient under various assumptions:
   (a) Quasi constant round (for single Alice). P and Visconti, 2005
   (b) 1 round if a common string is available to all
   (c) 4 rounds (optimal) if players have a (non-authenticated) keys in a public file. Di Crescenzo et al., 2004

Open Problem: Constant round with no assumption.

Other Side of Security

Suppose Alice does not know m

THEN

Alice is caught with probability $\geq 1/2$.

Implicit assumption: **Alice is executing only one session.**

Man In The Middle

Alice **Bob** **Charles**

$$\xrightarrow{\mathbf{C^A, H^A}}$$

$$\mathbf{C^B := C^A \cdot E(2)}$$
$$\mathbf{H^B := H^A \cdot E(r)}$$
$$\longrightarrow$$

Man In The Middle

Alice **Bob** **Charles**

$$\xrightarrow{\mathbf{C^A, H^A}}$$

$$\mathbf{C^B := C^A \cdot E(2)}$$
$$\mathbf{H^B := H^A \cdot E(r)}$$
$$\xrightarrow{\hspace{3em}}$$

$$\xleftarrow{\mathbf{0}}$$

$$\xleftarrow{\mathbf{0}}$$

$$\xrightarrow{\mathbf{a_A^0 : E(a_A^0) = H^A}}$$

$$\xrightarrow{\mathbf{a_B^0 := r \cdot a_A^0}}$$

Man In The Middle

Alice **Bob** **Charles**

$$\xrightarrow{\mathbf{C^A,H^A}}$$

$$\mathbf{C^B := C^A \cdot E(2)}$$
$$\mathbf{H^B := H^A \cdot E(r)}$$
$$\xrightarrow{\qquad\qquad}$$

$$\xleftarrow{\mathbf{1}}$$

$$\xleftarrow{\mathbf{1}}$$

$$\mathbf{a^1_A : E(a^1_A) = C^A \cdot H^A}$$
$$\xrightarrow{\qquad\qquad}$$

$$\mathbf{a^1_B := 2 \cdot r \cdot a^1_A}$$
$$\xrightarrow{\qquad\qquad}$$

Non-Malleable Protocols

State of the Art

1. $O(\log k)$ **rounds Dolev, Dwork and Naor 91**
2. **Constant round Barak 02, Pass and Rosen 05**

Concurrent Non-Malleable

1. **Impossible in the plain model Lindell 04**
2. **Constant round if the same random string is available to all parties [Di Crescenzo, De Santis Ostrovsky, P, Sahai 01, Canetti, Lindell, Ostrovsky, Sahai 02].**
3. **Constant round if players have (non-authenticated) public keys Ostrovsky, P, Visconti 06**