Cryptographic Protocols

Carlos Borrego

Carlos.Borrego@uab.cat

Departament d'Enginyeria de la Informació i de les Comunicacions
Universitat Autònoma de Barcelona

Criptografia i Seguretat

Contingut

1. Procols
2. Zero knowledge proofs
3. Homomorphic encryption
4. Commitment Protocols
5. Blind Signatures
6. Ring Signatures

Shamir's three-step protocol

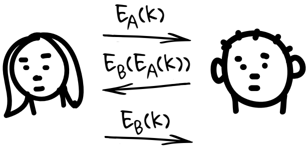
The protocol objective is to allow a secret communication between to parties **without any key exchange.**

The protocol assumes that the cryptosystem used **commutes**, that means the following property holds:

$$E_A(E_B(m)) = E_B(E_A(m))$$

Shamir's three-step protocol

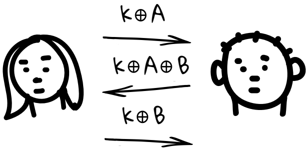
A wants to **send a key** (k) to be shared to B.



- A creates key k.
- A creates key A.
- A computes $E_A(k)$ and sends the result to B.
- B creates key B.
- B encrypts the received value $E_B(E_A(k))$ and sends the result to A.
- A decrypts the received value: $D_A(E_B(E_A(k))) = D_A(E_A(E_B(k))) = E_B(k)$ and sends the result to B.
- B computes $D_B(E_B(k))$ and obtains the key k.

Shamir's three-step protocol problem

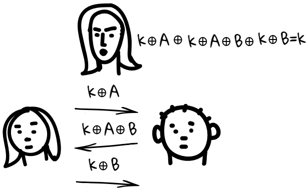
With $E_A(x) = x \oplus A$, then
$D_A(x) = x \oplus A$



- A creates key k.
- A creates key A.
- A computes $E_A(k)$ ($k \oplus A$) and sends the result to B.
- B creates key B.
- B encrypts the received $E_B(E_A(k))$ and sends it to A ($k \oplus A \oplus B$).
- A decrypts the received value:
  $D_A(E_B(E_A(k))) = D_A(E_A(E_B(k))) = E_B(k)$ and sends it to B ($k \oplus B$).
- B computes $D_B(E_B(k))$ and obtains the key k.

Shamir's three-step protocol problem

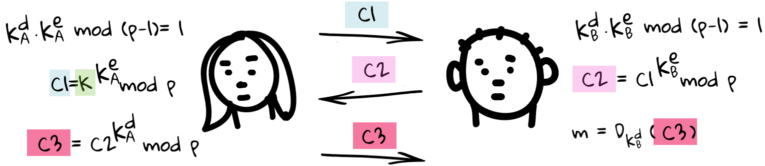
However, the protocol is totally non-secure in terms of secrecy!



An eavesdropper sees the following on the channel:
$k \oplus A$
$k \oplus A \oplus B$
$k \oplus B$
By simply x-oring the three inputs someone could **compute k**.

Shamir's three-step protocol solution

With $E_A(x) = x^A$:



Shamir's three-step protocol solution

■ **Exemple 7.1** **Exemple de protocol de tres passos de Shamir amb el criptosistema d'exponenciació.** En aquest exemple suposarem que els dos usuaris treballen amb el paràmetre $p = 131$. A més, l'usuari A disposarà de la clau de xifrat $k_A^e = 21$ i de la clau de desxifrat $k_A^d = (k_A^e)^{-1} \pmod{p-1} = 31$. D'altra banda, l'usuari B també tindrà el seu parell de claus. La de xifrat serà $k_B^e = 27$ i la de desxifrat $k_B^d = (k_B^e)^{-1} \pmod{p-1} = 53$.

Amb aquests paràmetres, l'usuari A vol enviar de forma secreta el missatge $m = 15$ a B i per fer-ho els passos del protocol seran els següents:

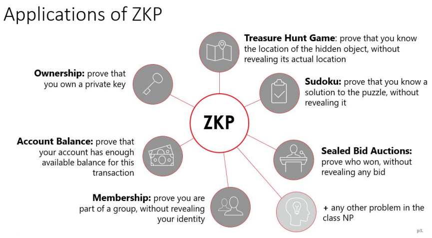
Pas	Alice		Bob
1.	$c_1 = 15^{21} \pmod{131} = 125$	$\xrightarrow{125}$	
2.		$\xleftarrow{27}$	$c_2 = (125)^{27} \bmod 131 = 27$
3.	$c_3 = (27)^{31} \bmod 131 = 129$	$\xrightarrow{129}$	
4.			$m = (129)^{53} \bmod 131 = 15$

Contingut

1. Procols
2. Zero knowledge proofs
3. Homomorphic encryption
4. Commitment Protocols
5. Blind Signatures
6. Ring Signatures

Zero knowledge proofs

A protocol involving a **prover** and a **verifier** that enables the prover to prove to a verifier without revealing any other information



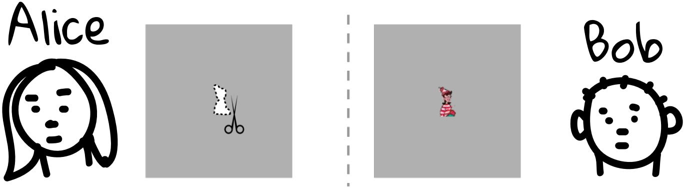
E.g., proving that a number n is of the form of the product of two prime number

Zero knowledge proofs properties

Completeness: If the statement is true, the honest verifier will be convinced of this fact by an honest prover.
Soundness: If the statement is false, no cheating prover can convince the honest verifier that it is true, except with some small probability
Zero knowledge: The proof does not leak any additional information

Zero knowledge proofs

Zero knowledge proofs



Completeness: As long as Alice finds Waldo, she's able to consistently use her proofs to show Waldo, in each game. Put simply, Alice's proof systems convince Bob that she found Waldo.
Soundness: Assuming Alice doesn't know Waldo's locations and presents random pieces of the scene to her proof systems then, her cardboard holes will display random images without Waldo. Put simply, Alice's proof systems are truthful and do not let her cheat.
Zero-Knowledge: As Alice proves to Bob that she has found Waldo, the only information revealed to Bob is that Alice has found Waldo. Waldo's location is never revealed. Put simply, Alice's proof systems prove her victory to Bob, without revealing her knowledge.

Zero knowledge proofs

How to Explain Zero-Knowledge Protocols to Your Children

QUISQUATER Jean-Jacques[1], *Myriam, Muriel, Michaël*
GUILLOU Louis[2], *Marie Annick, Gaïd, Anna, Gwenolé, Soazig*

in collaboration with *Tom BERSON*[3] for the English version

[1] Philips Research Laboratory, Avenue Van Becelaere, 2, B–1170 Brussels, Belgium.
[2] CCETT/EPT, BP 59, F–35512 Cesson Sévigné, France.
[3] Anagram Laboratories, P.O. Box 791, Palo Alto CA 94301, USA.

The Strange Cave of Ali Baba

◇ Know, oh my children, that very long ago, in the Eastern city of Baghdad, there lived an old man named Ali Baba. Every day Ali Baba would go to the bazaar to buy or sell things. This is a story which is partly about Ali Baba, and partly also about a cave, a strange cave whose secret and wonder exist to this day. But I get ahead of myself ...

One day in the Baghdad bazaar a thief grabbed a purse from Ali Baba who right away started to run after him. The thief fled into a cave whose entryway forked into two dark winding passages: one to the left and the other to the right (The Entry of the Cave).

Ali Baba did not see which passage the thief ran into. Ali Baba had to choose which way to go, and he decided to go to the left. The left-hand passage ended in a dead end. Ali Baba searched all the way from the fork to the dead end, but he did not find the thief. Ali Baba said to himself that the thief was perhaps in the other passage. So he searched the right-hand passage, which also came to a dead end. But again he did not find the thief. "This cave is pretty strange," said Ali Baba to himself, "Where has my thief gone?"

The following day another thief grabbed Ali Baba's basket and fled, as the first thief had fled, into the strange cave. Ali Baba pursued him, and again did not see which way the thief went. This time Ali Baba decided to search to the right. He went all the way to the end of the right-hand passage, but he did not find the thief. He said to himself that, like the first thief, the second thief had also been lucky in taking the passage Ali Baba did not choose to search. This had undoubtedly let the thief leave again and to blend quietly into the crowded bazaar.

The days went by, and every day brought its thief. Ali Baba always ran after the thief, but he never caught any of them. On the fortieth day a fortieth thief grabbed Ali Baba's turban and fled, as thirty-nine thieves had done before him, into the strange cave. Ali Baba yet again did not see which way the thief went. This time Ali Baba decided to search the left-hand passage, but again he did not find the thief at the end of the passage. Ali Baba was very puzzled.

He could have said to himself, as he had done before, that the fortieth thief had been as lucky as each of the other thirty-nine thieves. But this explanation was so

G. Brassard (Ed.): Advances in Cryptology - CRYPTO '89, LNCS 435, pp. 628-631, 1990.

LINK:
`http://pages.cs.wisc.edu/~mkowalcz/628.pdf`

Discrete Logarithm

Pas	Provador (P)		Verificador (V)
1.	Tria $r \in_R \mathbb{Z}_p \setminus \{0,1\}$		
	Calcula $c = g^r \pmod{p}$	$\xrightarrow{c}$	
2.		$\xleftarrow{b}$	Tria un bit aleatori $b \in_R \{0,1\}$
3.	Calcula $h = r + b \cdot x \pmod{p-1}$	$\xrightarrow{h}$	
4.			Verifica que
			$c \cdot y^b = g^h \pmod{p}$

Paper:

Chaum, D., Evertse, J.H. and Graaf, J.V.D., 1987, April. An improved protocol for demonstrating possession of discrete logarithms and some generalizations. In Workshop on the Theory and Application of of Cryptographic Techniques (pp. 127-141). Springer, Berlin, Heidelberg

Discrete Logarithm

Pas	Provador (P)		Verificador (V)
1.	Tria $r \in_R \mathbb{Z}_p \setminus \{0,1\}$		
	Calcula $c = g^r \pmod p$	$\xrightarrow{c}$	
2.		$\xleftarrow{b}$	Tria un bit aleatori $b \in_R \{0,1\}$
3.	Calcula $h = r + b \cdot x \pmod{p-1}$	$\xrightarrow{h}$	
4.			Verifica que $c \cdot y^b = g^h \pmod p$

If V always chooses 1 as *b*, P can generate r in step 1 as:

$$c' = \frac{g^r}{y} \pmod p$$

and in step 3, send r instead of $r + x$.

So, in step 4, the validation will be correct:

$$c' \cdot y = \frac{g^r}{y} \cdot y = g^r = g^h$$

Contingut

1. Procols
2. Zero knowledge proofs
3. **Homomorphic encryption**
4. Commitment Protocols
5. Blind Signatures
6. Ring Signatures

Homomorphic encryption

Protocol:

- Allows computations to be carried out on ciphertext
- Example: given E(a) and E(b), we can compute E(a+b), E(a·b) or E(a)·k
- There are several homomorphic crypto-systems. Example: Paillier

Paillier Cryptosystem

Properties:

- An homomorphic asymmetric algorithm for public key cryptography
- Additive homomorphic cryptosystem
- Homomorphic multiplication of plaintexts
- Encrypted numbers can be multiplied by a non encrypted scalar
- Does not include subtraction

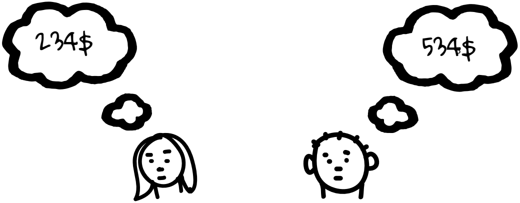
Paillier Cryptosystem

Recipe:

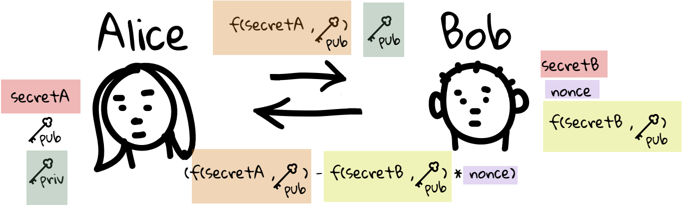
- Alice starts by selecting two random primes p and q and computes $n = pq$, $\lambda = \text{lcm}(p-1, q-1)$,$L(x) = (x-1)/n$
- Alice picks a random $g \in \mathbb{Z}^*_{n^2}$ such that $gcd((L(g^\lambda \bmod n^2)), n) = 1$, where $\lambda = \text{lcm}(p-1, q-1)$ and $L(x) = (x-1)/n$
- Alice's public key is $Pk_A : (n, g)$
- To encrypt a message m, Bob picks a random $r \in \mathbb{Z}^*_n$ and computes $c = E(m) = g^m \cdot r^n \bmod n^2$
- $E(a+b) = E(a) \cdot E(b) \bmod n^2 = g^{a+b} \cdot (r_1 \cdot r_2)^n \bmod n^2$
- To decrypt a ciphertext c, Alice computes $D(c) = L(c^\lambda \bmod n^2) = m$

Yao's Millionaires' problem

Yao's Millionaires' problem is a secure multi-party computation problem introduced in 1982 by computer scientist Andrew Yao. The problem discusses two millionaires, Alice and Bob, who are interested in knowing which of them is **richer** without **revealing** their actual wealth.



Yao's Millionaires' problem



Contingut

1. Procols
2. Zero knowledge proofs
3. Homomorphic encryption
4. Commitment Protocols
5. Blind Signatures
6. Ring Signatures

Commitment Example

La batalla d'insults d'espasa és una activitat que tot pirata ha de dominar. La intenció dels insults en la lluita amb espases és llançar un guàrdia contrari i permetre a un espadachín que pressioni el seu atacant. A tot el Carib, molts pirates fan servir insults estàndards.
Durant una baralla pot haver-hi un trencament natural en el joc d'espasa, on un pirata llançarà un insult com ara “Lluites com un llaurador”. L'adversari es veurà obligat a respondre amb una resposta enginyosa. Si la resposta és prou insultant, guanyaran la victòria a la batalla. Qui mantingui el domini podrà llançar el següent insult.

Commitment Example

El problema amb aquestes lluites és sempre qui comença a insultar. Els pirates són molt desconfiats i no es fien ni de les monedes ni de jocs tipus pedra, paper o tissora. Als pirates els encanten els protocols critptogràfics de compromís.

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Commitment Example

Contingut

1. Procols
2. Zero knowledge proofs
3. Homomorphic encryption
4. Commitment Protocols
5. Blind Signatures
6. Ring Signatures

Blind Signatures

Table 7.4: Protocol de signatura cega

Pas	Alice		Bob
1.	Tria $r \in_R \mathbb{Z}_n$ t.q $mcd(r,n) = 1$		
	Calcula $t = r^e \bmod n$		
	Tapat :		
	calcula $m' = m \cdot t \bmod n$	$\xrightarrow{m'}$	
2.			Signa el valor m' calculant:
		$\xleftarrow{s'}$	$s' = (m')^d \bmod n$
3.	Obté la signatura de m calculant		
	$s = \frac{s'}{r}$ (destapat)		

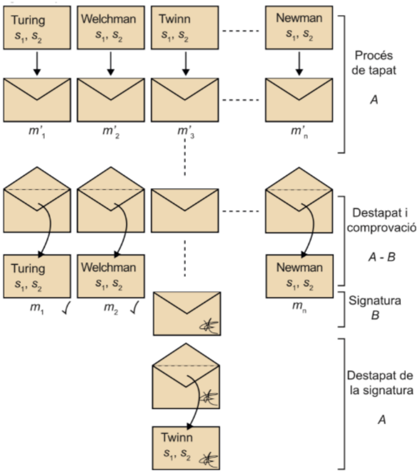
$$s = \frac{s'}{r} = \frac{(m')^d}{r} = \frac{(m \cdot t)^d}{r} = \frac{m^d \cdot t^d}{r} = \frac{m^d \cdot (r^e)^d}{r} = \frac{m^d \cdot r}{r} = m^d \pmod{n}$$

Blind Signatures

Pas	Alice		Bob
1.	Tria $25 \in_R \mathbb{Z}_{551}$ t.q. $mcd(15, 55) = 1$		
	Calcula $t = 25^{19} = 310 \bmod 551$		
	Tapat :		
	calcula $m' = 15 \cdot 310 = 242 \bmod 551$	$\xrightarrow{m'=242}$	
2.			Signa el valor $m' = 242$ calculant:
		$\xleftarrow{s'=14}$	$s' = 242^{451} = 14 \bmod 551$
3.	Obté la signatura de m calculant		
	$s = \frac{14}{25} = 14 \cdot 529 = 243 \pmod{551}$		

Fixeu-vos que el valor $s = 243$ és efectivament la signatura del missatge original $m = 15$ ja que $s = 15^{451} = 243 \bmod 551$

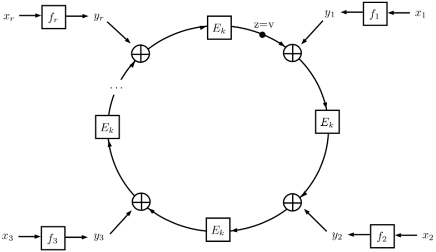
Blind Signatures



Contingut

1. Procols
2. Zero knowledge proofs
3. Homomorphic encryption
4. Commitment Protocols
5. Blind Signatures
6. Ring Signatures

Ring Signatures



Ring Signatures

Combination function:

$$C_{k,v}(y_1, y_2, \cdots, y_r) = E_k(y_r \oplus E_k(y_{r-1} \oplus E_k(y_{r-2} \oplus E_k(\cdots \oplus E_k(y_1 \oplus v) \cdots)))) = z$$

It is forced that the output z of the combination function must be equal to the initialization value v, that is:

$$C_{k,v}(y_1, y_2, \cdots, y_r) = v$$

Ring Signatures

1. El signant calcula els següent valors:
   - $k = h(m)$
   - b tal que $2^b > n_i$ per a $1 \leq i \leq r$
   - $v \in_R \{0,1\}^b$
   - $x_i \in_R \{0,1\}^b$ per a $1 \leq i \leq r$, per $i \neq s$
   - $y_i = f_i(x_i)$ per a $1 \leq i \leq r$, per $i \neq s$
2. Troba el valor y_s que soluciona l'equació: $C_{k,v}(y_1, y_2, \cdots, y_r) = v$
3. Calcula: $x_s = f_s^{-1}(y_s)$

Per tant, el valor de la signatura del missatge m serà

$$\sigma = \{PK_1, \cdots, PK_r, v, x_1, \cdots, x_r\}$$

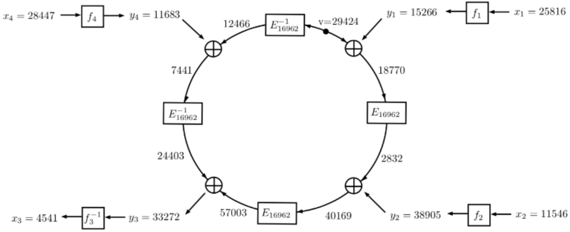
Ring Signatures

Suposarem també que el conjunt d'usuaris $\mathscr{U}$ amb claus públiques RSA conegudes que formaran part de l'anell serà $\mathscr{R} = \{u_1, u_2, u_3, u_4\}$ i les claus de cada un:
$PK_1 = (28907, 18541)$
$PK_2 = (41917, 22491)$
$PK_3 = (39407, 26077)$
$PK_4 = (32743, 17539)$
Per a aquest exemple, suposarem que l'usuari que fa la signatura és $s = 3$. La seva corresponent clau privada és $SK_3 = (39407, 27013)$.

El procés de signatura tindrà els següents passos:

1. Calcula:
   - $k = h(16962) = 16962$
   - $b = 16$: ja que $2^{16} = 65536 > n_i$ per a $1 \leq i \leq 4$
   - $v = 29424 \in_R \{0,1\}^{16}$
   - $x = \{25816, 11546, \emptyset, 28447\}$ amb $x_i \in_R \{0,1\}^{16}$
   - $y_1 = f_1(25816) = 25816^{18541} \pmod{28907} = 15266$
   - $y_2 = f_2(11546) = 11546^{22491} \pmod{41917} = 38905$
   - $y_4 = f_4(28447) = 28447^{17539} \pmod{32743} = 11683$
2. Troba el valor y_3 que soluciona l'equació: $C_{16962,29424}(15266, 38905, y_3, 11683) = 29424$ $y_3 = 33272$
3. Calcula: $x_3 = f_3^{-1}(33272) = 33272^{27013} \pmod{39407} = 4541$

Ring Signatures



Ring Signatures

$$\sigma = \{PK_1, PK_2, PK_3, PK_4, v, x_1, x_2, x_3, x_4\} =$$
$$= \{(28907, 18541), (41917, 22491), (39407, 26077), (32743, 17539), 29424, 25816, 11546, 4541, 28447\}$$

El verificador, per verificar la signatura realitzarà els següents passos:

1. Calcula:
   - $y_1 = f_1(25816) = 25816^{18541} \pmod{28907} = 15266$
   - $y_2 = f_2(11546) = 11546^{22491} \pmod{41917} = 38905$
   - $y_3 = f_3(4541) = 4541^{26077} \pmod{39407} = 33272$
   - $y_4 = f_4(28447) = 28447^{17539} \pmod{32743} = 11683$
   - $k = h(16962) = 16962$
2. Verifica:

$$C_{16962,29424}(15266, 38905, 33272, 11683) = 29424$$

Cryptographic Protocols

Carlos Borrego

Carlos.Borrego@uab.cat

Departament d'Enginyeria de la Informació i de les Comunicacions
Universitat Autònoma de Barcelona

Criptografia i Seguretat