

# Proofs of knowledge for non-monotone discrete log formulae and applications

---



Emmanuel Bresson  
Jacques Stern

École normale supérieure

1

## Contents

- Motivation: group signatures
  - ◆ Definition
  - ◆ Example
- Discrete-log based predicates
  - ◆ Definition
  - ◆ Composition
- Our contribution: Non-monotone formulae
  - ◆ How to prove “NOT”
- Applications to group signatures
  - ◆ Member revocation
  - ◆ Multi-signer mechanism
- Conclusions

2

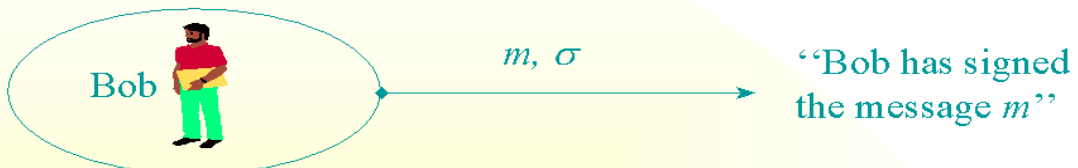
# Group signatures (1)

- Group-oriented cryptography
  - ◆ A publicly known group of players
  - ◆ A group manager and a judge
- Authentication without identification
  - ◆ Members sign-up with the manager
  - ◆ Members sign on behalf of the group
  - ◆ Anonymity within the group is granted
  - ◆ Can be lifted by the judge

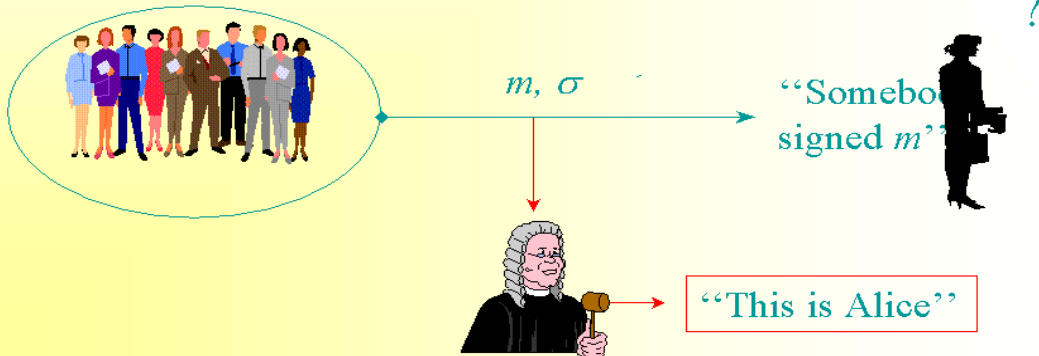
3

# Group signatures (2)

## •Ordinary signatures



## •Group signatures



4

# Blind issuing

- Interactive protocol:



- At the end:

Bob knows solution to hard problem  $x, y : C(x, y)$

Manager does not know  $x, y$  but receives commitment  $Y$  of  $y$

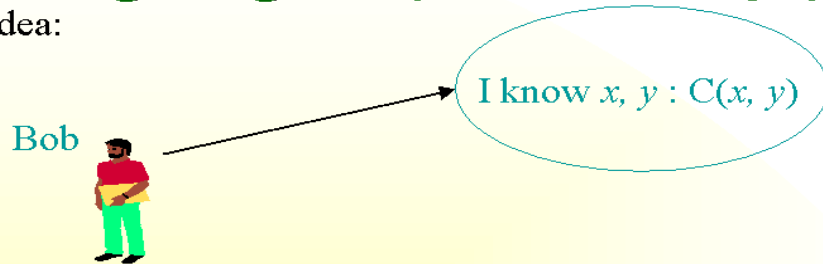
Typically  $Ux^{e1}-Vy^{e2}=1 \bmod n$

Typically  $Y = g^y$ , where  $g$  has order  $n$  in group  $G$

5

# Signing $m$ (first attempt)

- Basic idea:



I write a non-interactive ZK proof of knowledge of  $x, y$  depending on message  $m$ . This is my signature of  $m$

- Grants anonymity
- Not enough for lifting anonymity by Judge

6

# Camenisch et al. 's schemes

- Group signature:

Bob



I know  $x, y : C(x, y)$   
I encrypt  $Y$  for Judge

I write a non-interactive ZK proof of knowledge of  $x, y$  depending on message  $m$ . I prove at the same time that I have correctly encrypted  $Y = g^y$  for Judge. This is my signature of  $m$

- Uses El Gamal encryption

$$(A, B) = (g^r, h_j^r Y)$$

- Need to prove elaborate statements on discrete logs  
[CS97, CM98, ACJT00]

7

# Elaborate statements

- Adding extra variables:

Bob



I know  $x, y, r$   
 $Ux^3 - Vy^5 = 1 \pmod n$   
 $(A, B) = (g^r, h_j^r Y)$   
 $Y = g^y$

I know  $x, y, r, z, u$   
 $h^z = k^u g$   
 $A = g^r \quad B = h_j^r g^y$   
 $z = x^3 \quad u = y^5 \pmod n$

$h = g^U \quad k = g^V$   
 $z = x^3 \pmod n, \quad u = y^5 \pmod n$   
 $(A, B) = (g^r, h_j^r Y)$

- More variable:

$v = x^2 \pmod n$   
 $g^x = m_1$   
 $g^y = m_2$

I know  $x, y, r, z, u, v, w$   
 $h^z = k^u g$   
 $A = g^r \quad B = h_j^r g^y$   
 $g^x = m_1, \quad m_1^x = m_2, \quad m_2^x = g^z$   
 $u = y^5 \pmod n$

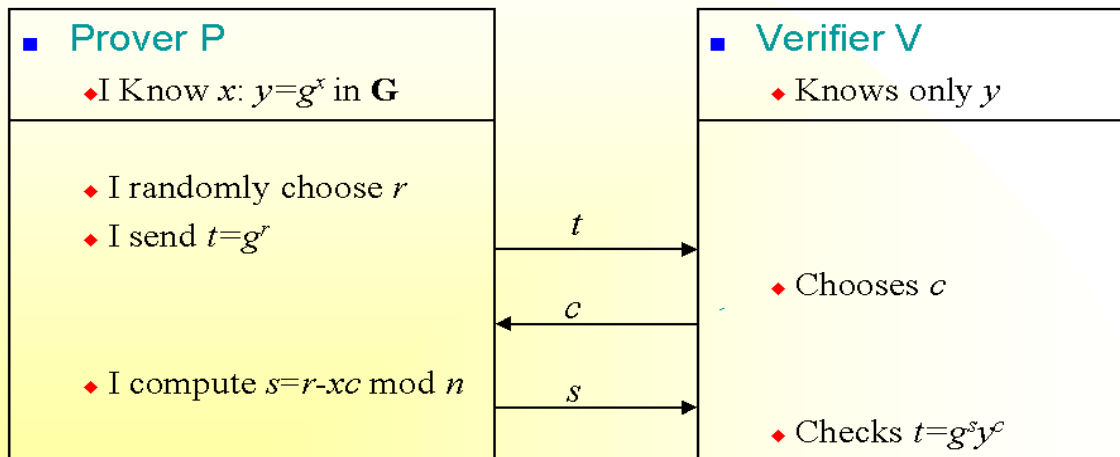
8

# Discrete-log based predicates

- Basic statements
  - ◆ atomic predicates.  $y = \prod_{i=1}^k g_i^{x_i}$
  - ◆ with  $g_1, \dots, g_m$  elements of Group  $G$ .
- Combined by OR, AND
- Monotone formulae only:
  - ◆ *no NOT connector.*

9

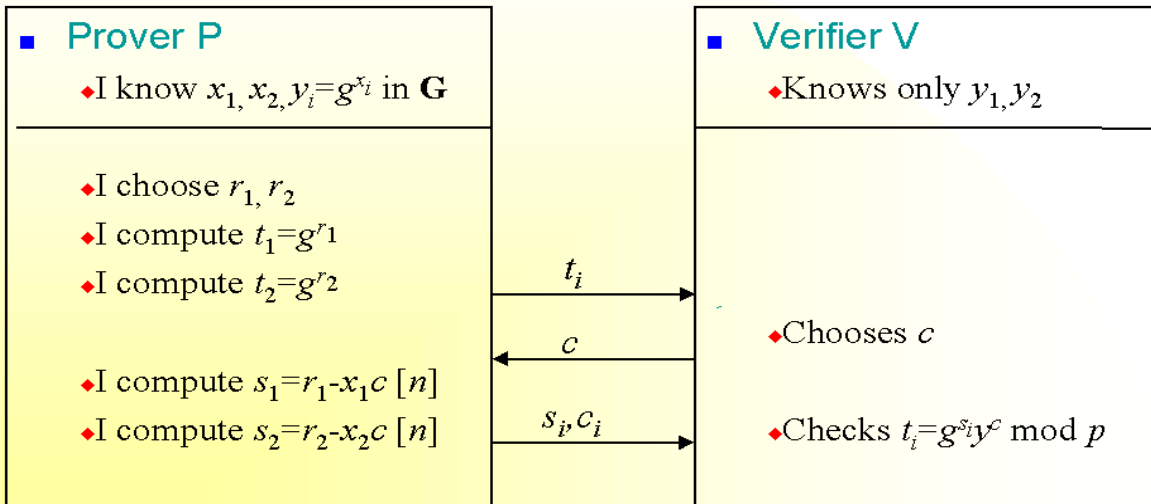
# Proof of knowledge of a discrete log



A pair  $(c, s)$  verifying :  $c = H(g || y || g^s y^c)$

10

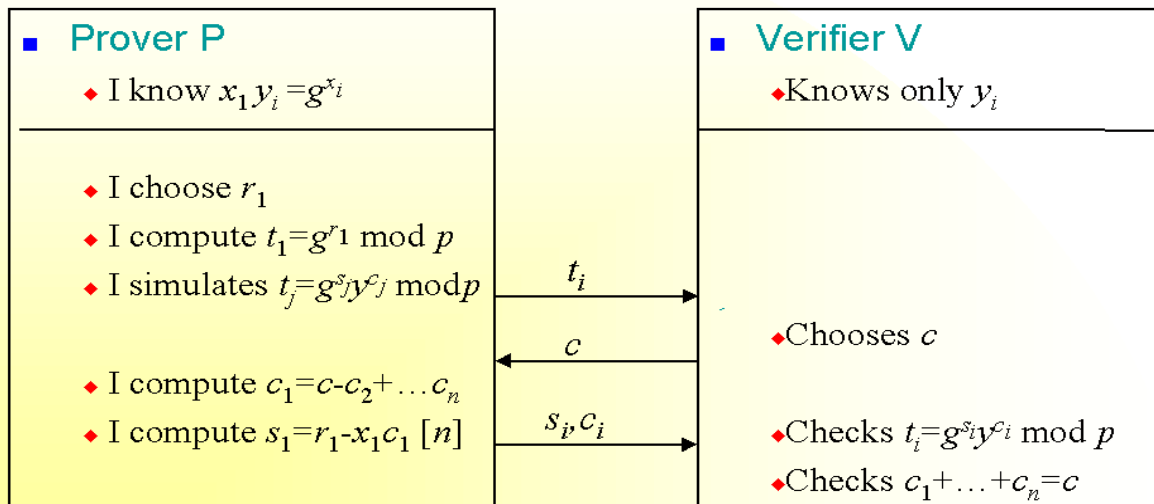
# Proof of AND



A “parallel” proof using the same challenge

11

# Proof of OR (1-out-of- $n$ )



A “share” of the challenge allows to simulate all parts but one

12

# Proving negations

- Proving inequalities on secret values, committed through discrete-log predicates
- Proving that two discrete logs are different
- Proving that two encryptions correspond to different plaintexts
- Proving that two encryptions have been performed using two different random values
- Earlier partial solution by Brands [Bra97]:

13

# Typical example

- Alice:
  - ◆ knows secret key  $x_A$  such that  $Y_A = g^{x_A}$ ,
- Bob:
  - ◆ knows secret key  $x_B$  such that  $Y_B = h^{x_B}$ ,
- Without knowing  $x_B$ , Alice knows that  $x_A \neq x_B$  by computing  $h^{x_A}$  (which is  $\neq Y_B$ )
- Our method shows how to prove the fact

Not feasible by previous techniques

14

# Our solution

- Trick:
  - ◆ Compute a “witness” value:  $w = (h^{x_A} / Y_B)^r$
  - ◆ Verifier: check that  $w \neq 1$
  - ◆ Alive proves that she knows  $s, z$  such that  $w = h^s Y_B^z$  AND  $1 = g^s Y_A^z$
  - ◆ It follows that  $\log_h Y_B \neq \log_g Y_A = x_A$

15

# Member revocation (1)

- Adding members: easy
  - ◆ Creation of a new private key for new member
  - ◆ The group public key remains the same (membership updated)
- Excluding a member
  - ◆ Many reasons (departure or exclusion)
  - ◆ Prevent a member to sign in the future
  - ◆ Keep anonymity of past signatures

16

# Member Revocation (2)

- Without external help
  - ◆ Not changing the group public key
- Prevent Alice to sign
  - ◆ Without revealing her secret key
  - ◆ → By requiring a stronger proof for a membership certificate, that only non-revoked members can compute.

17

# Revocation in group signatures

- New signature

Bob



Alice has been revoked  
her committed secret key was  $Y_A$



- New group



I know  $x, y : C(x, y)$   
I know that  $g^y \neq Y_A$   
I encrypt  $Y$  for Judge

I write a non-interactive ZK proof of knowledge of  $x, y$  depending on message  $m$ . I prove at the same time that I have correctly encrypted  $Y = g^y$  for Judge and that  $g^y \neq Y_A$ .  
This is my signature of  $m$

18

# Threshold mechanism in group signatures

Bob and Alice  
want to sign together



We know  $x, y : C(x, y)$   
and also  $x', y' : C(x', y')$   
We encrypt  $Y \neq Y'$  for Judge

We write a non-interactive ZK proof of knowledge of  $x, y, x', y'$  depending on message  $m$ . We prove at the same time that we have correctly encrypted  $Y = g^y$  and  $Y' = g^{y'}$  for Judge and that  $Y \neq Y'$ .  
This is our signature of  $m$

19

## Conclusion

- We have shown a new class of provable discrete log predicates
  - ◆ Includes non-monotone formulae
  - ◆ Includes multiplicative formulae (not covered in talk)
  - ◆ More general than [Bra97]
- Enhanced group signatures with revocation
  - ◆ Efficient schemes
  - ◆ Adapted for dynamic groups with limited number of revocations
- Designed new group signature schemes
  - ◆ Threshold features

20