On The Security of Unique-Witness Blind Signature Schemes

December 2013
ASIACRYPT, Bangalore, India

Foteini Baldimtsi, Anna Lysyanskaya

BROWN

Blind Signatures [Chaum'82]

Blind signatures are a special type of digital signatures.

- Signer is different that the message author.
- Author “blinds” the message before sending it to the signer.
- Signer learns nothing about the message.

Applications

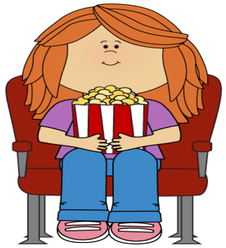
Values need to be certified but anonymity should be preserved.

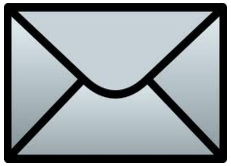
Security for Blind Signatures

Pointcheval and Stern ('96):

- definition of security for blind signatures
- reduction for proving security of blind signatures

1. blindness: signer is unable to view the messages he signs and a malicious signer cannot link signatures to specific executions.



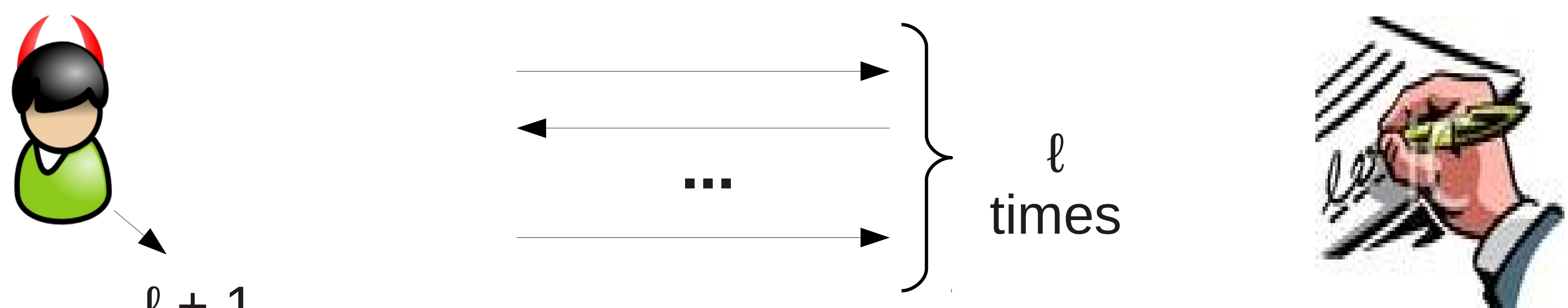


Security for Blind Signatures

Pointcheval and Stern ('96):

- definition of security for blind signatures
- reduction for proving security of blind signatures

2. one-more unforgeability: a user interacting with a signer cannot output an additional, valid message/ signature pair no matter how many pairs of (messages, signatures) of the signer he has seen.



$\ell + 1$
Valid signatures

Motivation for our work

The security of some of the oldest (and most efficient) blind signatures [GQ'88, Schnorr'89, Brands'93] is an open problem...

Some of them are used in practice!

Brands blind signature is used in Microsoft's UProve system

What can we show about the security of these blind signature schemes?

Related Work

- Pointcheval, Stern 1996: constructed and proved secure a multi-witness variant of the Schnorr blind signature

- Schnorr, Jakobsson, 1999: Schnorr blind signature is secure in the generic group model

- Fischlin, Schroder 2011: impossible to prove unique witness blind signatures secure *in the standard model* for *non-interactive* assumptions

- Pass 2011: showed that Schnorr ID scheme (and therefore blind signature) cannot be proven secure under unbounded composition based on a bounded-round assumption in the standard model

Our results

We rule out a wide class of reductions for proving one-more unforgeability of certain blind signature schemes in the RO model no matter what assumption one makes.

- Define Generalized Blind Schnorr Signatures (GBSS)
- Random Oracle replay reductions [PS'96]
- Meta-reduction technique
- Perfect naive and L-naive reductions
- Proof for Perfect Naive

Generalized Blind Schnorr Signatures

1. Unique witness relation between (sk,pk)

i.e. sk in Z_q and pk $=g^{sk}$
for g, pk members of G of order q

Generalized Blind Schnorr Signatures

1. Unique witness relation between (sk,pk)
2. Signer's side is like a Σ-protocol
3. The signature σ(a,c,r) has identical distribution to a transcript of a Σ-protocol
4. User makes a Hash query to compute c

Prover (sk,pk=g^{sk}) **Verifier** (pk)

$\xrightarrow{a}$

$\xleftarrow{c}$

$\xrightarrow{r}$

decides to accept on (pk,a,c,r)

- (a,c,r) & (a,c,r) ⇨ efficiently compute sk
- exists simulator S that on input (pk,c) outputs accepting (a,c,r) with same distribution as honest discussion

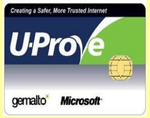
Generalized Blind Schnorr Signatures

1. Unique witness relation on (sk,pk)
2. Signer's side is like a Σ-protocol
3. The signature σ(a,c,r) has identical distribution to a transcript of a Σ-protocol
4. User makes a Hash query to compute c
5. There exists efficient algorithm s.t. on input (sk,pk), valid (a,c,r) and random c computes r such that: (a,c,r) is also valid

Generalized Blind Schnorr Signatures

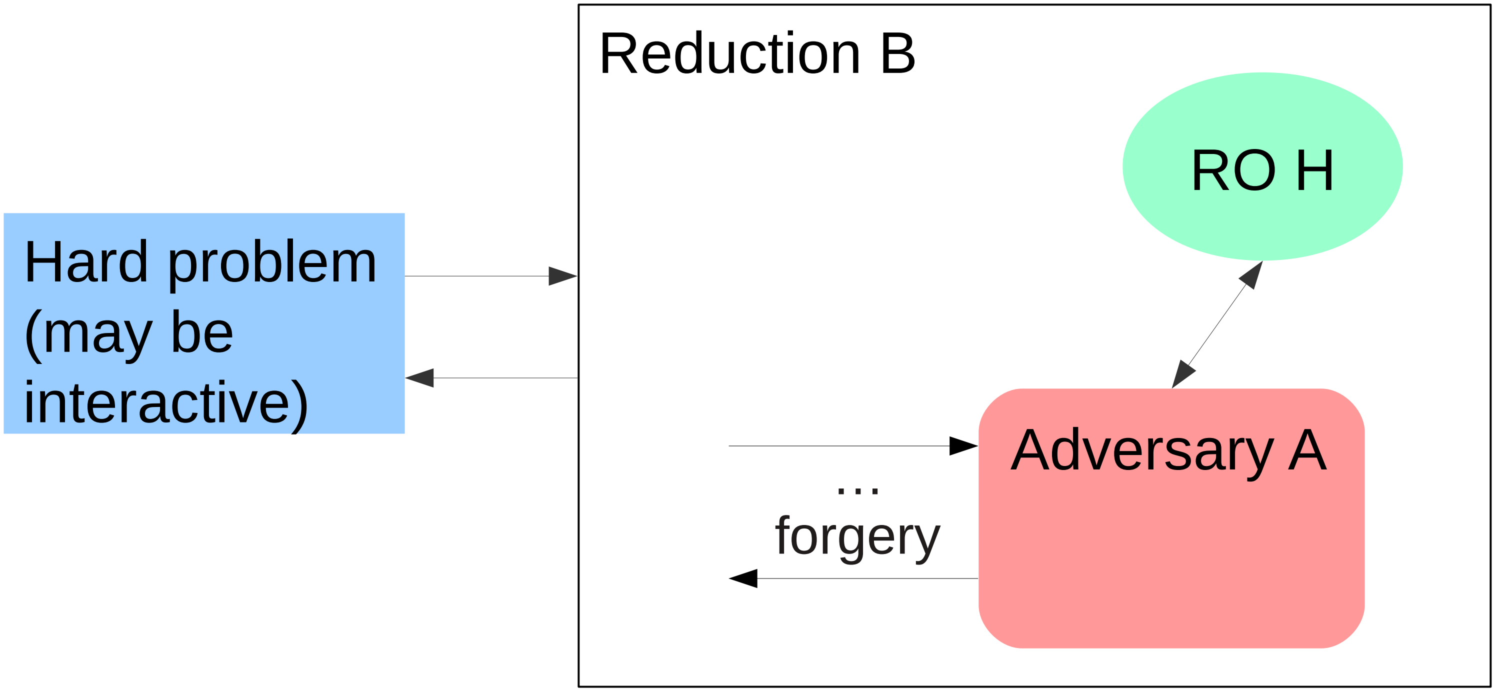
1. Unique witness relation on (sk,pk)
2. Signer's side is like a Σ-protocol
3. The signature σ(a,c,r) has identical distribution to a transcript of a Σ-protocol
4. User makes a Hash query to compute c
5. There exists efficient algorithm s.t. on input (sk,pk), valid (a,c,r) and random c computes r such that: (a,c,r) is also valid

- Blind Schnorr Sign. [Okamoto '91]
- GQ Blind Sign. [Okamoto '91]
- Brands Blind Sign. [Brands '93]

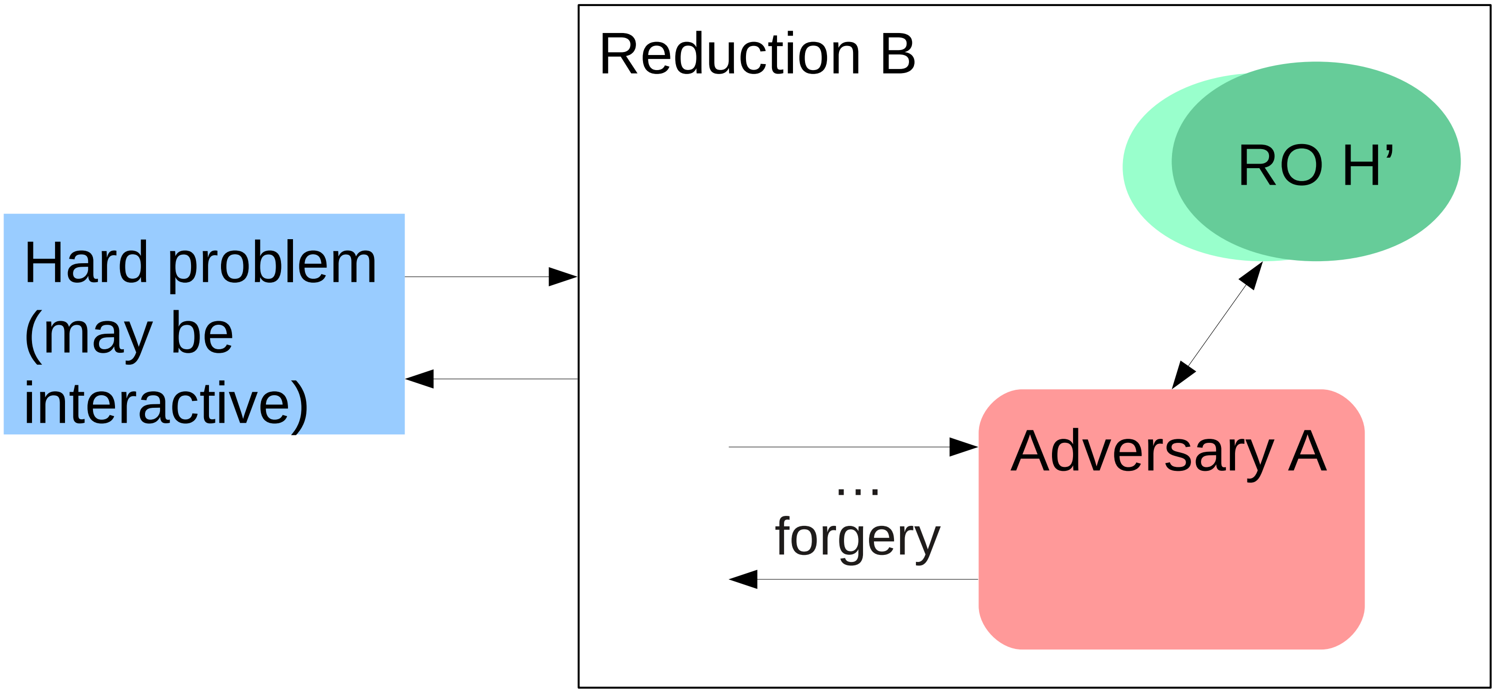


Generalized Blind Schnorr Signatures GBSS

Random Oracle Replay Reduction [PS'96] Unforgeability



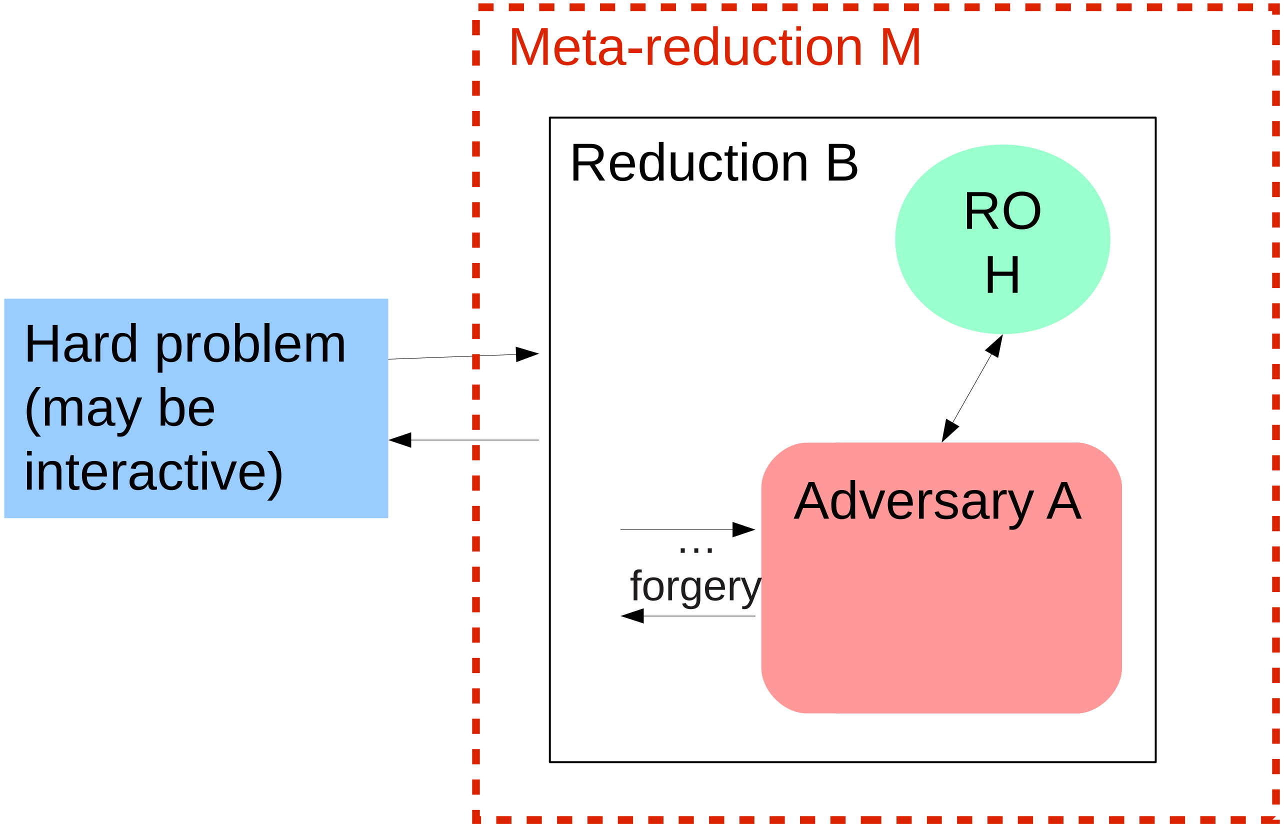
Random Oracle Replay Reduction [PS'96] Unforgeability



With non-negligible probability get ***σ(m)=(a,c,r)*** *and* ***σ(m)=(a,c,r)*** on the same message ***m*** and break the hard problem!

How do we rule out reductions?

Meta-reduction paradigm: “reduction against the reduction”

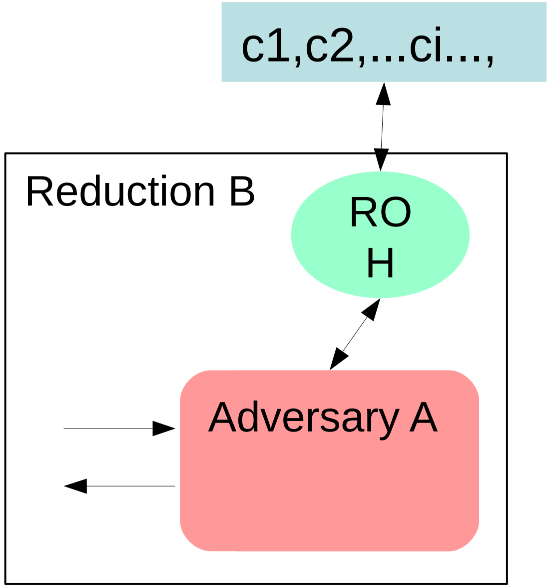


Goal: construct poly-time A so that A+B solves the problem, then it can be solved in poly-time CONTRADICTION

Which reductions do we rule out?

Perfect Naive and L-naive Replay Reductions

Naive Replay Reductions special tape for RO queries, always answers with next value on tape or some function of it	
Perfect Naive A gets same view inside B as it would get “in the wild” Not true for many reductions	**L- Naive** for all A, B runs A at most L times True for all reductions I know (PS'96, AO'04, Coron'00, BR'93 etc.)



Proof Outline: the Tale of Two Adversaries

<u>super adversary sA:</u>
can compute SK from PK
(we don't know how
to do this in poly-time)

$\approx$

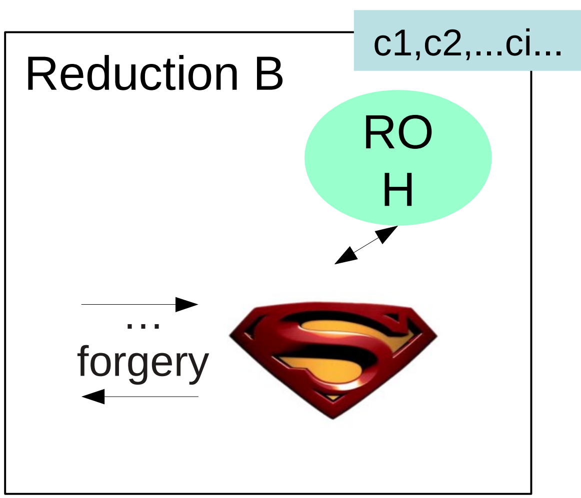
statistically,
as far as B
can tell

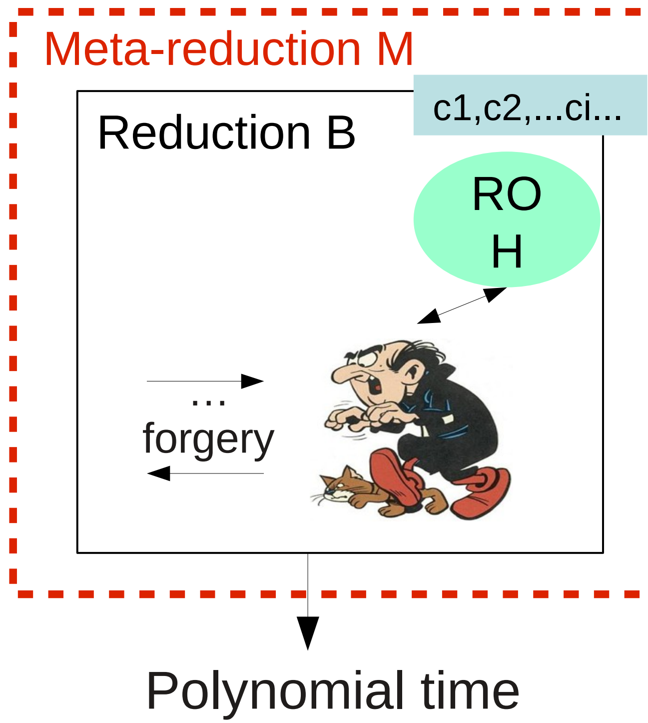
<u>B's personal nemesis pA:</u>
has special powers:
1) can see RO-tape
2) can remember its
past lives
(pA is poly-time)

If B works at all, it works with adversary sA. But then it also works with pA, since they are indistinguishable to B. Both B and pA are poly-time, therefore together they break the assumption (CONTRADICTION).

Proof Outline: the Tale of Two Adversaries

- pA and sA attack the unforgeability property of Generalized Blind Schnorr Signatures
- Interact with B to receive one signature and output two valid signatures (forgery)





sA for Perfect Naive Reduction

Reduction B

c1,c2,...,ci,...,

PK, a $\longrightarrow$

1. Find SK from PK
2. Compute two forgeries σ1 = (a1,c1,r1), σ2=(a2,c2,r2)

sA for Perfect Naive Reduction

Reduction B

c1,c2,...,ci,...,

PK, a

2 RO queries:
(m1,pk,a1),
(m2,pk,a2)

1. Find SK from PK
2. Compute two forgeries σ1 = (a1,c1,r1), σ2=(a2,c2,r2)

sA for Perfect Naive Reduction

Reduction B

c1,c2,...,ci,...,

PK, a

c

r

2 RO queries:
(m1,pk,a1),
(m2,pk,a2)

1. Find SK from PK
2. Compute two forgeries σ1 = (a1,c1,r1), σ2=(a2,c2,r2)
3. c ⇦ PRF(transcript)
4. If r correct
   output σ1, σ2

sA for Perfect Naive Reduction

what happens if sA is reset by B?

Same queries? depends on (pk,a)

Reduction B

c1,c2,...,ci,...,

Different with high prob.

2 RO queries: (m1,pk,a1), (m2,pk,a2)

c

r

1. Find SK from PK
2. Compute two forgeries σ1 = (a1,c1,r1), σ2=(a2,c2,r2)
3. c ⇦ PRF(transcript)
4. If r correct
   output σ1, σ2

pA for Perfect Naive Reduction

Reduction B

c1,c2,...,ci,...,

PK, a

1. look at RO tape: get c1,c2
2. pick random r1,r2 & solve for a1,a2 using the simulator of the Σ-protocol

pA for Perfect Naive Reduction

Reduction B

c1,c2,...,ci,...,

PK, a

2 RO queries:
(m1,pk,a1),
(m2,pk,a2)

1. look at RO tape: get c1,c
2. pick random r1,r2 & solve for a1,a2 using the simulator of the Σ-protocol

pA for Perfect Naive Reduction

Reduction B

c1,c2,...,ci,...,

2 RO queries:
(m1,pk,a1),
(m2,pk,a2)

PK, a →

← c

r →

1. look at RO tape: get c1,c
2. pick random r1,r2 & solve for a1,a2 using the simulator of the Σ-protocol
3. set $\sigma 1 = (a1,c1,r1)$, $\sigma 2=(a2,c2,r2)$
4. c ⇦ PRF(transcript)
5. If r correct output $\sigma 1,\sigma 2$

pA for Perfect Naive Reduction

what happens if pA is reset by B?

Reduction B

c1,c2,...,ci,...,

same
PK, a

pA for Perfect Naive Reduction

what happens if pA is reset by B?

Reduction B c1,c2,...,ci,...,

same
PK, a →

← c

r →

1. look at RO tape: get c3,c4
2. <u>same</u> RO queries: (m1,pk,a1),(m2,pk,a2)
3. cannot compute his forgeries for these RO queries
4. c ⇦ PRF(transcript)
5. If r correct: previous conversation was (pk,a,c,r), current is (pk,a,c,r) ⇨ sk
6. Output forgeries σ1,σ2

pA for Perfect Naive Reduction

what happens if pA is reset by B?

Reduction B

c1,c2,...,ci,...,

same
PK, a

c

r

1. look at RO tape: get c3,c4
2. same RO ,pk,a2)
3. cannot c ese
   RO quer
4. c ⇦ PRF(transcript)
5. If r correct: previous conversation was (pk,a,c,r), current is (pk,a,c,r) ⇨ sk
6. Output forgeries σ1,σ2

Get stuck if previous run wasn't perfect: didn't include r!

pA ≈ sA for Perfect Naive Reduction

super adversary sA:
- always outputs 2 (pseudo) random signatures

≈

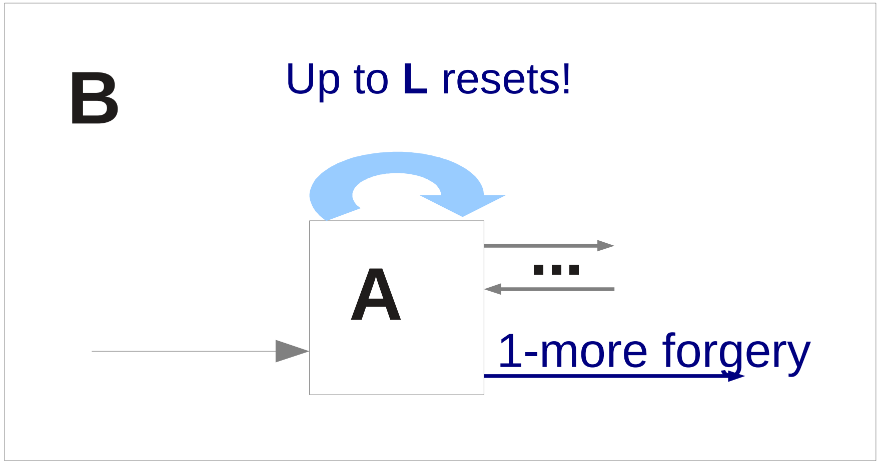
as far as B can tell

B's personal nemesis pA:
- outputs 2 (pseudo) random signatures when c ≠ c

Ruling Out More Reductions

~~Assumption: *B* is **perfect --** it always gives valid responses to *A.*~~

L-Naive RO replay reduction



- pA and sA succeed in forging with some probability
- pA also has write access to B's RO tape

Conclusion

Theorem: No perfect or L-naive RO replay reduction can prove Generalized Blind Schnorr signatures unforgeable under any assumption (even an interactive one!)

- Interesting fact: our meta-reduction doesn't need to reset the reduction.
- Brands, GQ, Schnorr blind signature cannot be proven unforgeable using a perfect or L-naive reduction.

Thanks for your attention!

http://eprint.iacr.org/2012/197