II – Zero-Knowledge Proofs and Applications

David Pointcheval
Ecole normale supérieure/PSL, CNRS & INRIA

Outline

Zero-Knowledge Proofs of Knowledge

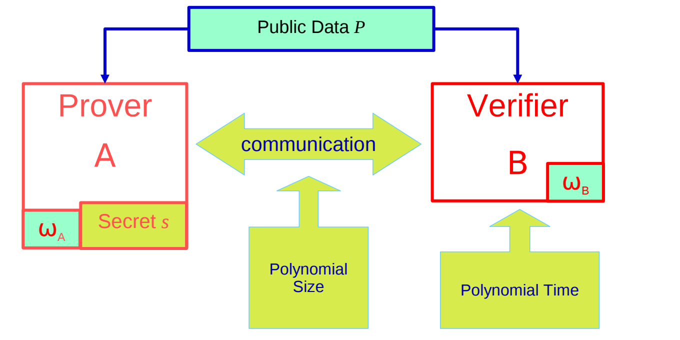
Outline

Proof of Knowledge

How do I prove that I know a solution s to a problem P?

Proof of Knowledge

How do I prove that I know a solution s to a problem P?



Proof of Knowledge: Soundness

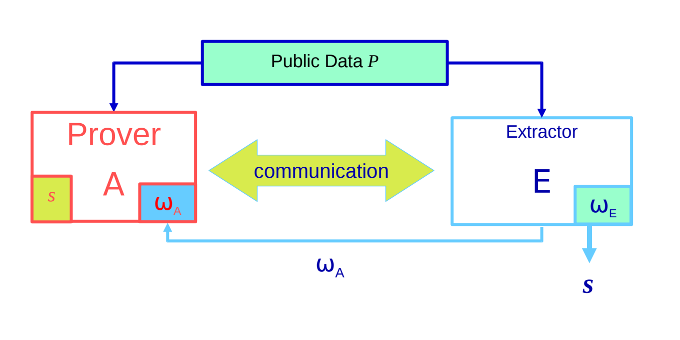
$\mathcal{A}$ knows something... What does it mean?

the information can be extracted: extractor

Proof of Knowledge: Soundness

$\mathcal{A}$ knows something... What does it mean?

the information can be extracted: extractor



Proof of Knowledge: Zero-Knowledge

How do I prove that I know a solution s to a problem P?

I reveal the solution...

Proof of Knowledge: Zero-Knowledge

How do I prove that I know a solution s to a problem P?

I reveal the solution. . .

How can I do it without revealing any information?

Zero-knowledge: simulation and indistinguishability

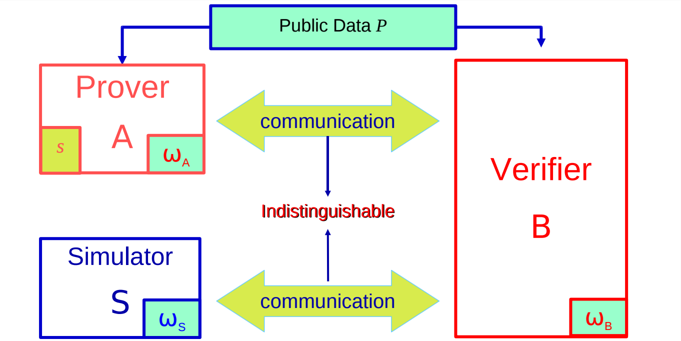
Proof of Knowledge: Zero-Knowledge

How do I prove that I know a solution s to a problem P?

I reveal the solution...

How can I do it without revealing any information?

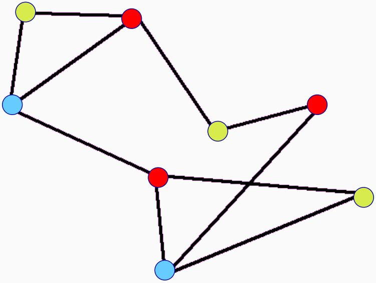
Zero-knowledge: simulation and indistinguishability



Outline

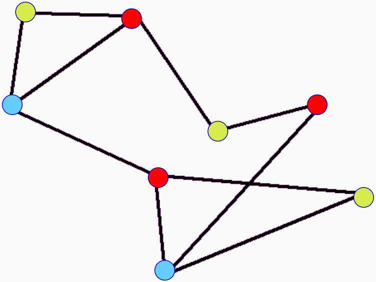
Proof of Knowledge

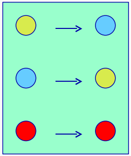
How do I prove that I know a 3-color covering,
without revealing any information?



Proof of Knowledge

How do I prove that I know a 3-color covering,
without revealing any information?

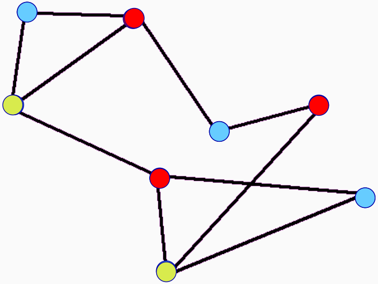


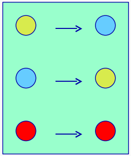


I choose a random permutation on the colors

Proof of Knowledge

How do I prove that I know a 3-color covering,
without revealing any information?

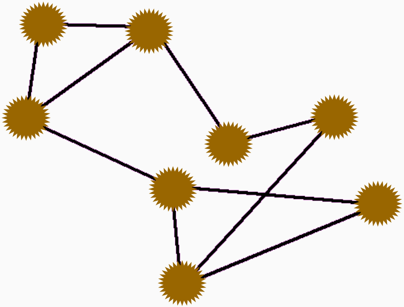




I choose a random permutation on the colors
and I apply it to the vertices

Proof of Knowledge

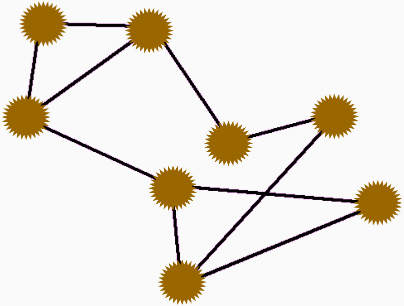
How do I prove that I know a 3-color covering,
without revealing any information?



I mask the vertices

Proof of Knowledge

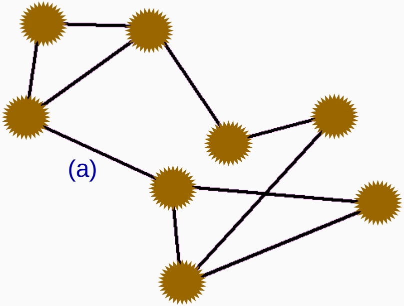
How do I prove that I know a 3-color covering,
without revealing any information?



I mask the vertices
and send it to the verifier

Proof of Knowledge

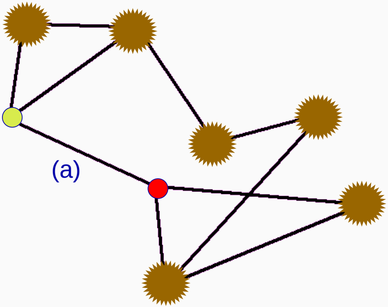
How do I prove that I know a 3-color covering,
without revealing any information?



The verifier chooses an edge

Proof of Knowledge

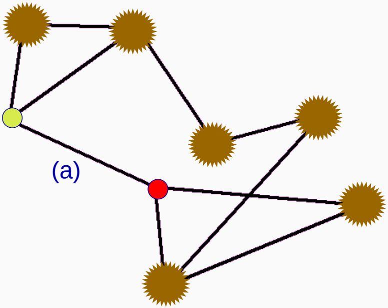
How do I prove that I know a 3-color covering,
without revealing any information?



The verifier chooses an edge
I open it

Proof of Knowledge

How do I prove that I know a 3-color covering,
without revealing any information?



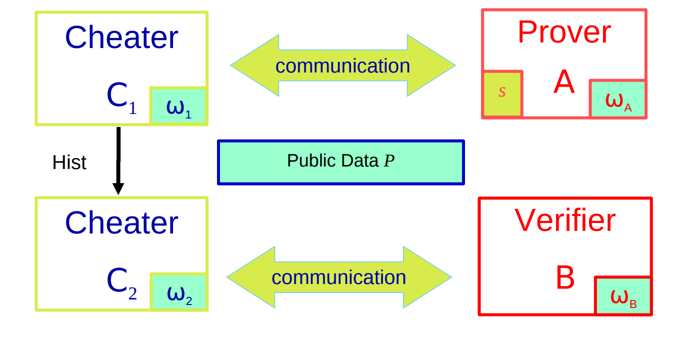
The verifier chooses an edge

I open it

The verifier checks the validity: 2 different colors

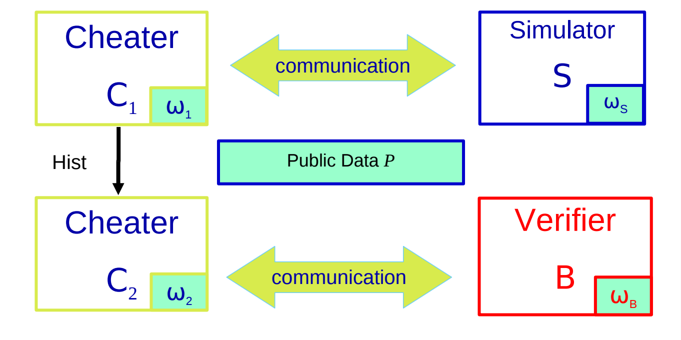
Secure Multiple Proofs of Knowledge: Authentication

If there exists an efficient adversary,
then one can solve the underlying problem:



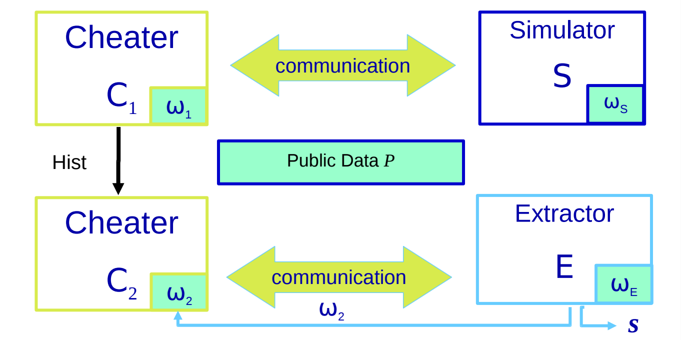
Secure Multiple Proofs of Knowledge: Authentication

If there exists an efficient adversary,
then one can solve the underlying problem:



Secure Multiple Proofs of Knowledge: Authentication

If there exists an efficient adversary,
then one can solve the underlying problem:



Outline

Zero-Knowledge Proofs of Knowledge

3-Pass Zero-Knowledge Proofs

Generic Proof

- Proof of knowledge of x, such that $\mathcal{R}(x, y)$
- $\mathcal{P}$ builds a commitment r and sends it to $\mathcal{V}$
- $\mathcal{V}$ chooses a challenge $h \stackrel{R}{\leftarrow} \{0,1\}^k$ for $\mathcal{P}$
- $\mathcal{P}$ computes and sends the answer s
- $\mathcal{V}$ checks (r, h, s)

Σ-Protocol

- Proof of knowledge of x
- $\mathcal{P}$ sends a commitment r
- $\mathcal{V}$ sends a challenge h
- $\mathcal{P}$ sends the answer s
- $\mathcal{V}$ checks (r, h, s)

3-Pass Zero-Knowledge Proofs

Generic Proof

- Proof of knowledge of x, such that $\mathcal{R}(x, y)$
- $\mathcal{P}$ builds a commitment r and sends it to $\mathcal{V}$
- $\mathcal{V}$ chooses a challenge $h \stackrel{R}{\leftarrow} \{0,1\}^k$ for $\mathcal{P}$
- $\mathcal{P}$ computes and sends the answer s
- $\mathcal{V}$ checks (r, h, s)

Σ-Protocol

- Proof of knowledge of x
- $\mathcal{P}$ sends a commitment r
- $\mathcal{V}$ sends a challenge h
- $\mathcal{P}$ sends the answer s
- $\mathcal{V}$ checks (r, h, s)

Special soundness

If one can answer to two different challenges $h \neq h'$:
$\Longrightarrow$ s and s' for a unique r
$\Longrightarrow$ one can extract x

SQRT Fiat-Shamir Proof

[Fiat-Shamir – Crypto '86]

- Setting: $n = pq$
  $\mathcal{P}$ knows x, such that $X = x^2 \bmod n$ and wants to prove it to $\mathcal{V}$
- $\mathcal{P}$ chooses $r \stackrel{R}{\leftarrow} \mathbb{Z}_n^\star$, sets and sends $R = r^2 \bmod n$
- $\mathcal{V}$ chooses $b \stackrel{R}{\leftarrow} \{0, 1\}$ and sends it to $\mathcal{P}$
- $\mathcal{P}$ computes and sends $s = x^b \times r \bmod n$
- $\mathcal{V}$ checks whether $s^2 \stackrel{?}{=} X^b R \bmod n$

One then reiterates t times

For a fixed R, two valid answers s and s' satisfy

$$s^2/X = R = (s')^2 \bmod n \Longrightarrow X = (s/s')^2 \bmod n$$

And thus $x = s/s' \bmod n \Longrightarrow$ Special Soundness

SQRT Fiat-Shamir Proof

[Fiat-Shamir – Crypto '86]

- Setting: $n = pq$
  $\mathcal{P}$ knows x, such that $X = x^2 \bmod n$ and wants to prove it to $\mathcal{V}$
- $\mathcal{P}$ chooses $r \stackrel{R}{\leftarrow} \mathbb{Z}_n^\star$, sets and sends $R = r^2 \bmod n$
- $\mathcal{V}$ chooses $b \stackrel{R}{\leftarrow} \{0, 1\}$ and sends it to $\mathcal{P}$
- $\mathcal{P}$ computes and sends $s = x^b \times r \bmod n$
- $\mathcal{V}$ checks whether $s^2 \stackrel{?}{=} X^b R \bmod n$

One then reiterates t times

For a fixed R, two valid answers s and s' satisfy

$$s^2/X = R = (s')^2 \bmod n \Longrightarrow X = (s/s')^2 \bmod n$$

And thus $x = s/s' \bmod n \Longrightarrow$ Special Soundness

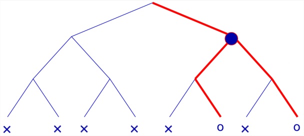
Fiat-Shamir Proof: Extraction

More precisely: the execution of t repetitions depends on

- $(b_1, \ldots, b_t)$ from the verifier $\mathcal{V}$
- ω that (together with the previous b_i $(i < k)$)
  determines R_k from the prover $\mathcal{P}$

If $\Pr_{\omega,(b_i)}[\mathcal{V} \text{ accepts } \mathcal{P}] > 1/2^t + \varepsilon$,
there is a good fraction of ω (more than $\varepsilon/2$)
such that $\Pr_{(b_i)}[\mathcal{V} \text{ accepts } \mathcal{S}] \geq 1/2^t + \varepsilon/2$.

For such a good ω: a good node along the successful path



Fiat-Shamir Proof: Simulation

Honest Verifier

Simulation of a triplet: $(R = r^2 \bmod n, b, s = x^b \times r \bmod n)$
for $r \xleftarrow{R} \mathbb{Z}_n^\star$ and $b \xleftarrow{R} \{0,1\}$
Similar to: $(R = s^2/X^b \bmod n, b, s)$
for $s \xleftarrow{R} \mathbb{Z}_n^\star$ and $b \xleftarrow{R} \{0,1\}$
Simulation: random s and b, and set $(R = s^2/X^b \bmod n, b, s)$

Any Verifier

Simulation of a triplet: $(R = r^2 \bmod n, b = \mathcal{V}(\text{view}), s = x^b \times r \bmod n)$
for $r \xleftarrow{R} \mathbb{Z}_n^\star$ only!
Similar to: $(R = s^2/X^b \bmod n, b = \mathcal{V}(\text{view}), s)$ for $s \xleftarrow{R} \mathbb{Z}_n^\star$
Simulation: random s and β, and set $R = s^2/X^\beta \bmod n$
upon reception of b: if $b = \beta$, output s, else rewind
b and β independent: rewind once over 2 $\Longrightarrow$ linear time

RSA GQ Proof

[Guillou-Quisquater – Crypto '87 – Eurocrypt '88]

- Setting: $n = pq$ and an exponent e
  $\mathcal{P}$ knows x, such that $X = x^e \bmod n$ and wants to prove it to $\mathcal{V}$
- $\mathcal{P}$ chooses $r \stackrel{R}{\leftarrow} \mathbb{Z}_n^\star$, sets and sends $R = r^e \bmod n$
- $\mathcal{V}$ chooses $b \stackrel{R}{\leftarrow} \{0,1\}^t$ and sends it to $\mathcal{P}$
- $\mathcal{P}$ computes and sends $s = x^b \times r \bmod n$
- $\mathcal{V}$ checks whether $s^e \stackrel{?}{=} X^b R \bmod n$

For a fixed R, two valid answers s and s' satisfy

$$s^e / X^b = R = (s')^e / X^{b'} \bmod n \Longrightarrow X^{b'-b} = (s'/s)^e \bmod n$$

If e prime and bigger than 2^t, then e and $b' - b$ are relatively prime:
Bezout: $ue + v(b'-b) = 1 \Longrightarrow X^{v(b'-b)} = (s'/s)^{ve} = X^{1-ue} \bmod n$
As a consequence: $X = ((s'/s)^v X^u)^e \Longrightarrow$ Special Soundness

DL Schnorr Proof

[Schnorr – Eurocrypt '89 - Crypto '89]

- Setting: $\mathbb{G} = \langle g \rangle$ of order q
  $\mathcal{P}$ knows x, such that $y = g^{-x}$ and wants to prove it to $\mathcal{V}$
- $\mathcal{P}$ chooses $k \stackrel{R}{\leftarrow} \mathbb{Z}_q^\star$, sets and sends $r = g^k$
- $\mathcal{V}$ chooses $h \stackrel{R}{\leftarrow} \{0,1\}^t$ and sends it to $\mathcal{P}$
- $\mathcal{P}$ computes and sends $s = k + xh \bmod q$
- $\mathcal{V}$ checks whether $r \stackrel{?}{=} g^s y^h$

For a fixed r, two valid answers s and s' satisfy

$$g^s y^h = r = g^{s'} y^{h'} \Longrightarrow y^{h'-h} = g^{s-s'}$$

And thus $x = (s - s')(h' - h)^{-1} \bmod q \Longrightarrow$ Special Soundness

Signatures

Outline

Σ-Protocols

Zero-Knowledge Proof

- Proof of knowledge of x
- $\mathcal{P}$ sends a commitment r
- $\mathcal{V}$ sends a challenge h
- $\mathcal{P}$ sends the answer s
- $\mathcal{V}$ checks (r, h, s)

Signature

- Key Generation $\rightarrow (y, x)$
- Signature of $m \rightarrow (r, h, s)$
  Commitment r
  Challenge $h = \mathcal{H}(m, r)$
  Answer s
- Verification of (m, r, s)
  compute $h = \mathcal{H}(m, r)$
  and check (r, h, s)

Special soundness

If one can answer to two different challenges $h \neq h'$: s and s'
for a unique commitment r, one can extract x

Σ-Protocols

Zero-Knowledge Proof

- Proof of knowledge of x
- $\mathcal{P}$ sends a commitment r
- $\mathcal{V}$ sends a challenge h
- $\mathcal{P}$ sends the answer s
- $\mathcal{V}$ checks (r, h, s)

Signature

- Key Generation $\rightarrow (y, x)$
- Signature of $m \rightarrow (r, h, s)$
  Commitment r
  Challenge $h = \mathcal{H}(m, r)$
  Answer s
- Verification of (m, r, s)
  compute $h = \mathcal{H}(m, r)$
  and check (r, h, s)

Special soundness

If one can answer to two different challenges $h \neq h'$: s and s'
for a unique commitment r, one can extract x

Outline

Forking Lemma

[Pointcheval-Stern – Eurocrypt '96]

The Forking Lemma shows an efficient reduction between the signature scheme and the identification scheme, but basically,
if an adversary $\mathcal{A}$ produces, with probability $\varepsilon \geq 2/2^k$,
a valid signature (m, r, h, s), then within $T' = 2T$,
one gets two valid signatures (m, r, h, s) and (m, r, h', s'),
with $h \neq h'$ with probability $\varepsilon' \geq \varepsilon^2/32q_H^3$.

The special soundness provides the secret key.

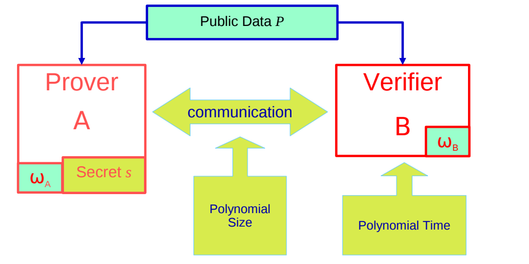
Zero-Knowledge Proofs of Membership

Outline

Proof of Membership

How do I prove that a word w lies in a language $\mathcal{L}$: $P = (w, \mathcal{L})$?

- if $\mathcal{L} \in \mathcal{NP}$: a witness s can help prove that $w \in \mathcal{L}$



If $w \notin \mathcal{L}$:

- Proof (perfect soundess): a powerful $\mathcal{A}$ cannot cheat
- Argument (computational soundness): a limited $\mathcal{A}$ cannot cheat

Proof of Membership

Soundness

$w \in \mathcal{L} \ldots$ what does it mean?

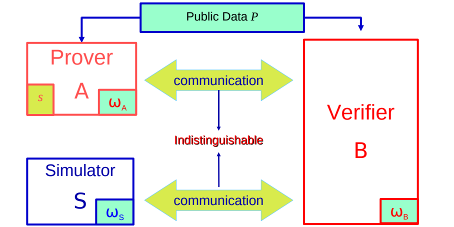
a witness exists, different from knowing it: no need of extractor

Zero-Knowledge

How do I prove there exists a witness s? I reveal it...

How can I do it without revealing any information?

Zero-knowledge:
simulation
and indistinguishability



Outline

Diffie-Hellman Language

In a group $\mathbb{G} = \langle g \rangle$ of prime order q,

the $\mathbf{DDH}(g, h)$ assumption states it is hard to distinguish

$$\mathcal{L} = (u = g^x, v = h^x) \text{ from } \mathbb{G}^2 = (u = g^x, v = h^y)$$

- $\mathcal{P}$ knows x, such that $(u = g^x, v = h^x)$ and wants to prove it to $\mathcal{V}$
- $\mathcal{P}$ chooses $k \stackrel{R}{\leftarrow} \mathbb{Z}_q^\star$, sets and sends $U = g^k$ and $V = h^k$
- $\mathcal{V}$ chooses $h \stackrel{R}{\leftarrow} \{0,1\}^t$ and sends it to $\mathcal{P}$
- $\mathcal{P}$ computes and sends $s = k - xh \bmod q$
- $\mathcal{V}$ checks whether $U \stackrel{?}{=} g^s u^h$ and $V \stackrel{?}{=} h^s v^h$

For a fixed (U, V), two valid answers s and s' satisfy

$$g^s u^h = U = g^{s'} u^{h'} \quad h^s v^h = V = h^{s'} v^{h'}$$

- if one sets $y = (s - s')(h' - h)^{-1} \bmod q \Longrightarrow u = g^y$ and $v = h^y$
- there exists a witness: Perfect Soundness