PSE Summer Module 1 - Overview of Zero Knowledge Proofs

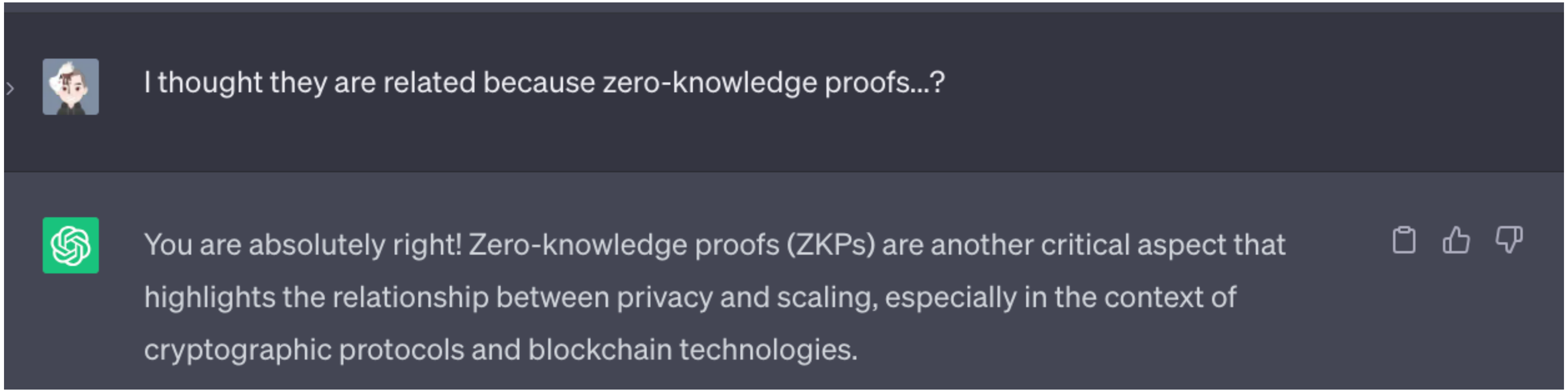
Presented by Jason

Date: Aug. 02, 2023

Group: MACI

Privacy & Scaling Explorations

- How are privacy and scaling related?
- ChatGTP: “As systems and platforms scale up, meaning they handle larger volumes of data and users, privacy concerns become more significant and complex.”



I thought they are related because zero-knowledge proofs...?

You are absolutely right! Zero-knowledge proofs (ZKPs) are another critical aspect that highlights the relationship between privacy and scaling, especially in the context of cryptographic protocols and blockchain technologies.

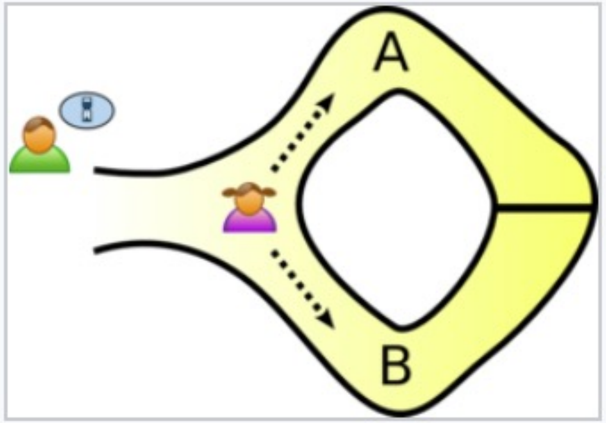
Privacy

- As an informal definition, ZKP allow one party (the prover) to **demonstrate the truth** of a statement to another party (the verifier) **without revealing any additional information** apart from the fact that the statement is true.
- Use-cases include anonymous payments, DIDs, Over 18 ZKRP, electronic voting, etc.
- Note that ZKP are proofs of "knowledge", not proofs of truth.

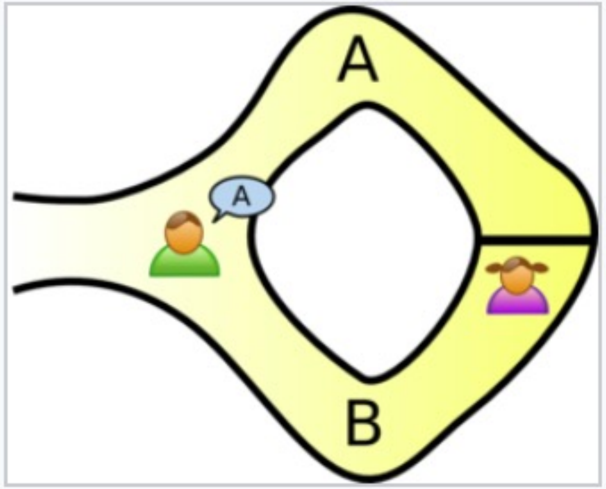
Scaling

- There are different solutions for scaling.
  - On-chain solutions require extensive modification of the blockchain's base layer.
  - Off-chain solutions rely on an outsourced computation model to improve throughput, and Ethereum only needs to apply the computation results to its state.
- In the off-chain scheme, ZKP can assist with validating off-chain transactions **without re-executing them**.

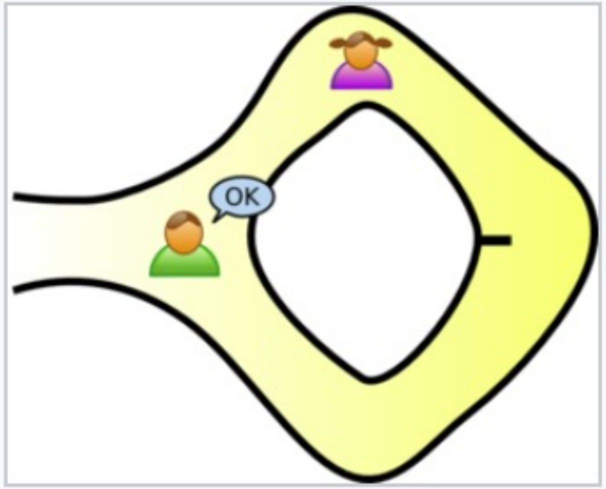
The Ali Baba Cave - Abstract Example



Peggy randomly takes either path A or B, while Victor waits outside



Victor chooses an exit path



Peggy reliably appears at the exit Victor names

- If Peggy knows the magic word, she can always success.
- If Peggy doesn’t know the secret word, she will fail after many trails.
- Victor can never know the secret word.

ZKP Property

- A ZK protocol must satisfy the following criteria:
  - **Completeness**: If the prover has proper knowledge, his proof can be accepted by the verifier.
  - **Soundness**: If the prover has no proper knowledge, his proof can not be accepted by the verifier.
  - **Zero-knowledge**: The proof reveal nothing more than “this statement is true”.

ZKP Property

- A ZK protocol must satisfy the following criteria:
  - **Completeness**: If the prover has proper knowledge, his proof can be accepted by the verifier.

    ↕ $symmetric$
  - **Soundness**: If the prover has no proper knowledge, his proof can not be accepted by the verifier.
  - **Zero-knowledge**: The proof reveal nothing more than “this statement is true”.

ZKP Property

- A ZK protocol must satisfy the following criteria:
  - **Completeness**: If the prover has proper knowledge, his proof can be accepted by the verifier.
  - **Soundness**: If the prover has no proper knowledge, his proof can not be accepted by the verifier.

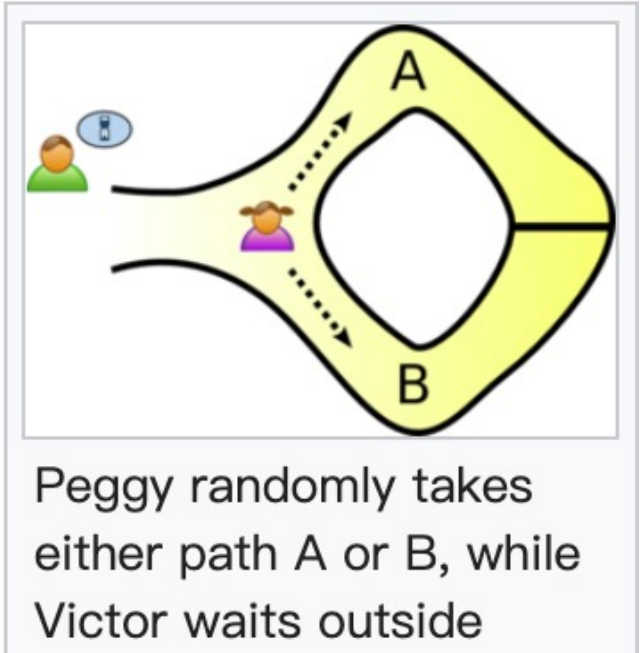
    ↕ $symmetric$
  - **Zero-knowledge**: The proof reveal nothing more than “this statement is true”.

ZKP Property

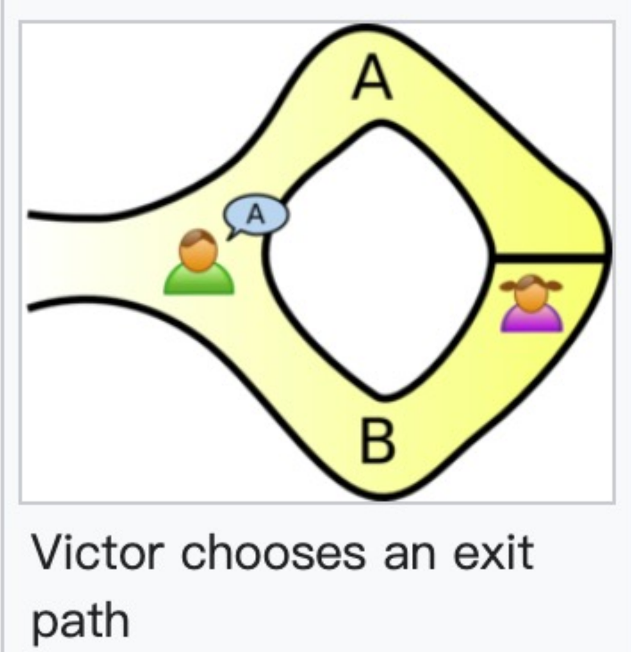
- A ZK protocol must satisfy the following criteria:
  - **Completeness**: If the prover has proper knowledge, his proof can be accepted by the verifier.
  - **Soundness**: Protect the verifier from being cheated.
    → trust issue → scaling issue

    ↕ $symmetric$
  - **Zero-knowledge**: Protect the prover from information leakage.
    → privacy issue

The Ali Baba Cave - What If... ?



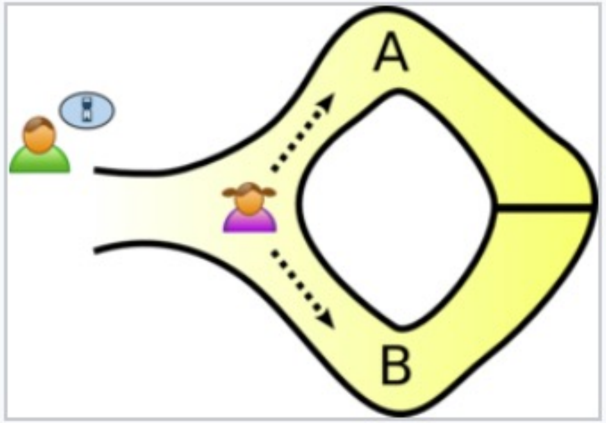
Peggy randomly takes either path A or B, while Victor waits outside



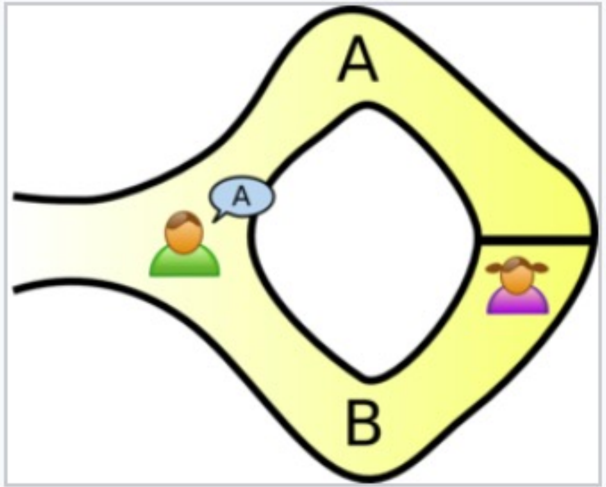
Victor chooses an exit path

- Peggy is She-Hulk?
- We can prove the **zero-knowledge(ness)**.

The Ali Baba Cave - What If Else… ?



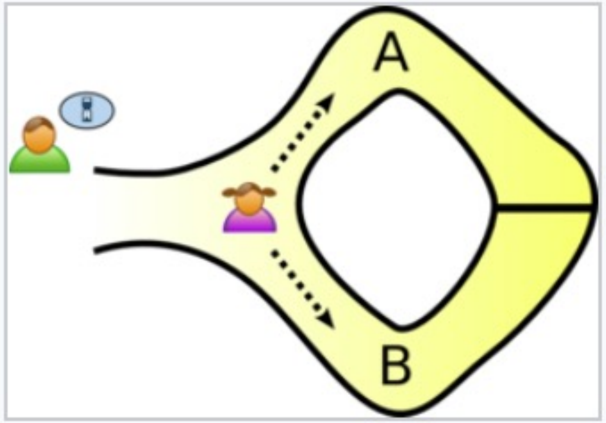
Peggy randomly takes either path A or B, while Victor waits outside



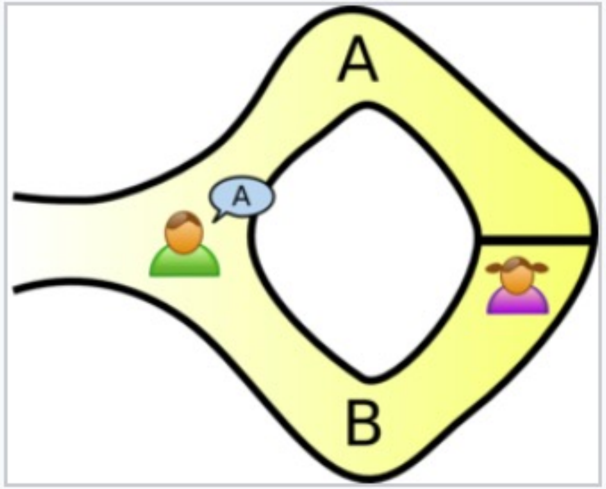
Victor chooses an exit path

- Victor is Daredevil?
- We can prove the **soundness**.

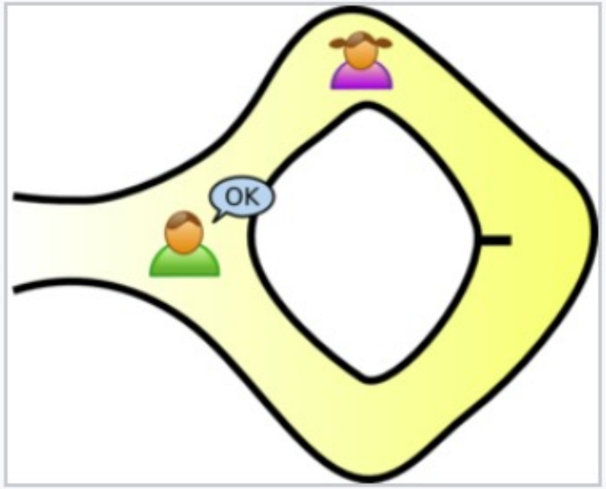
The Ali Baba Cave - What Else… ?



Peggy randomly takes either path A or B, while Victor waits outside



Victor chooses an exit path



Peggy reliably appears at the exit Victor names

- Peggy just goes in through A and comes out from B?
- The 3rd party will be sure about Peggy’s knowledge.

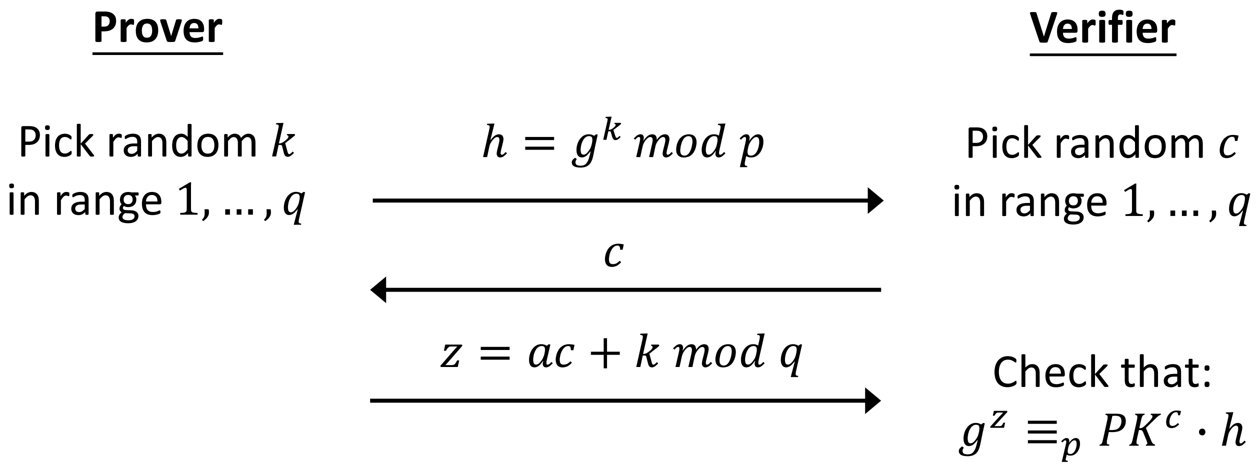
Interactive vs. Non-Interactive

- “Interactive” means interact more than once.
- “Non-interactive” means **interact only once**.
    - ZK-SNARK (Zero-Knowledge Succinct **Non-Interactive** Argument of Knowledge)
    - ZK-STARKs (Zero-Knowledge Scalable Transparent Argument of Knowledge)

Let’s Do Some Math.

Schnorr's Protocol

- Assumption:
  - Let p be some prime number, and let g be a generator of a cyclic group of prime-order q.
  - To generate a keypair, the prover picks a random integer a between 1 and q and compute
$$(SK, PK) = (a, g^a \; mod \; p)$$
- Interactive version of the protocol:



Prover

Verifier

Pick random k in range $1, \dots, q$

$h = g^k \; mod \; p$

Pick random c in range $1, \dots, q$

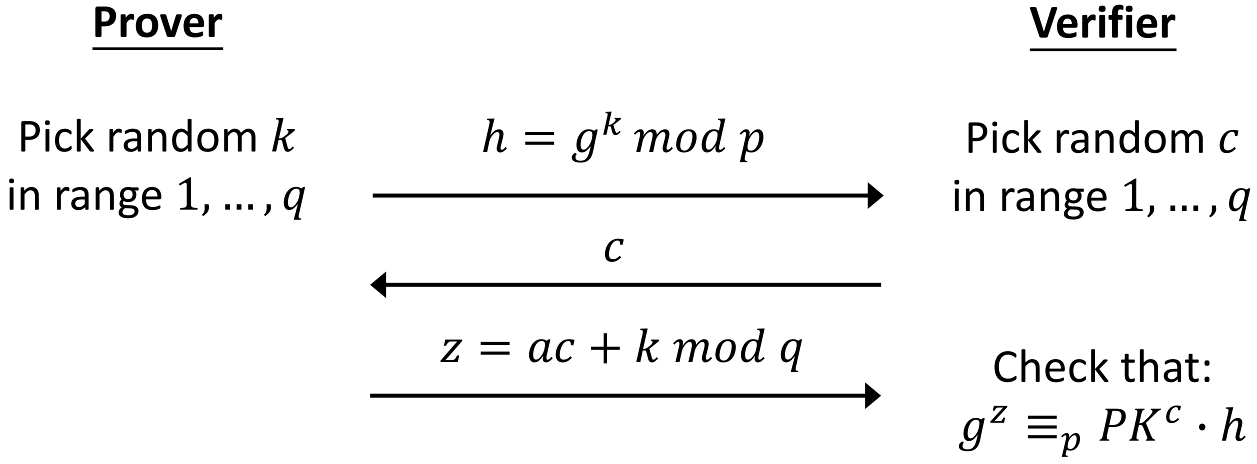
c

$z = ac + k \; mod \; q$

Check that: $g^z \equiv_p PK^c \cdot h$

Proving the Completeness

- Completeness in this context: if the prover does know the private key $SK = a$, he can provide the correct proof z such that $g^z \equiv_p PK^c \cdot h$.
- Method: $g^z \equiv_p g^{ac+k} \equiv_p (g^a)^c \cdot g^k \equiv_p PK^c \cdot g^k \equiv_p PK^c \cdot h$



Prover

Verifier

Pick random k
in range $1, \dots, q$

$h = g^k \, mod \, p$

Pick random c
in range $1, \dots, q$

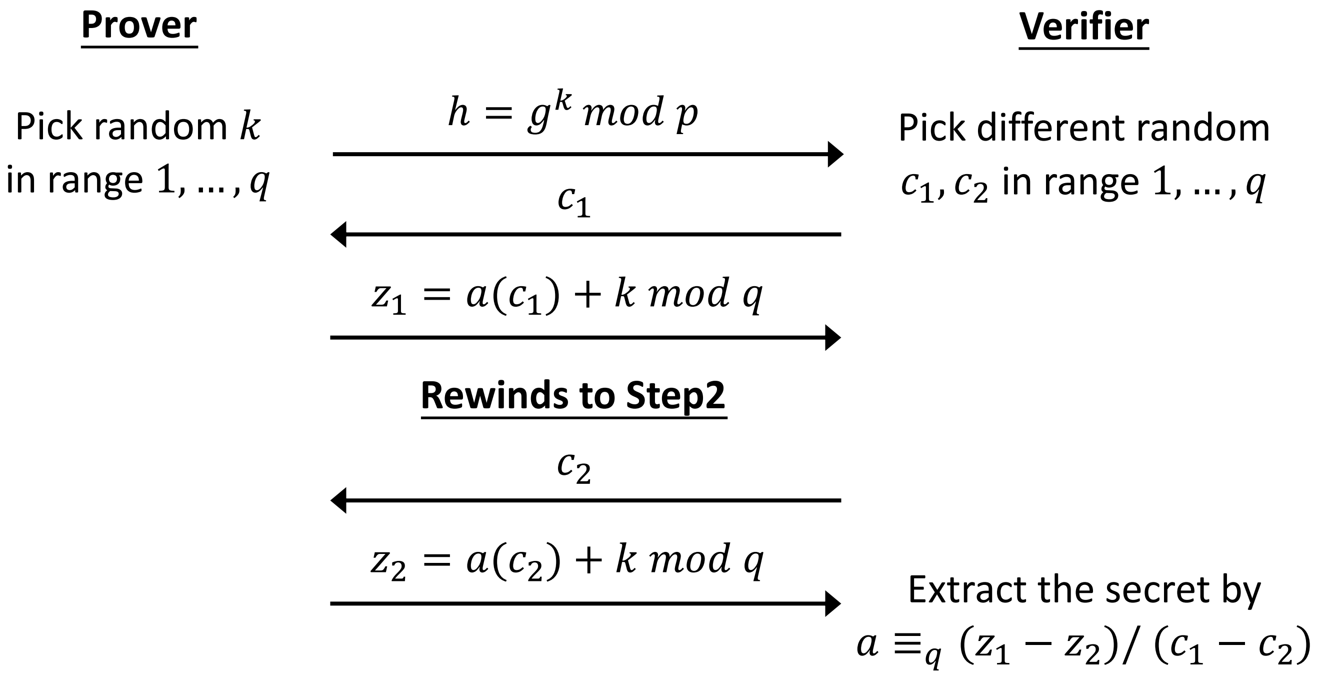
c

$z = ac + k \, mod \, q$

Check that:
$g^z \equiv_p PK^c \cdot h$

Proving the Soundness

- Soundness in this context: if the prover does**n't** know the private key $SK = a$, he can**not** provide the correct proof z such that $g^z \ \equiv_p PK^c \cdot h$.
- Method: use an **extractor** (i.e. Daredevil) to recover the prover's knowledge.



Prover

Verifier

Pick random k
in range $1, \dots, q$

$h = g^k \; mod \; p$

Pick different random
c_1, c_2 in range $1, \dots, q$

c_1

$z_1 = a(c_1) + k \; mod \; q$

Rewinds to Step2

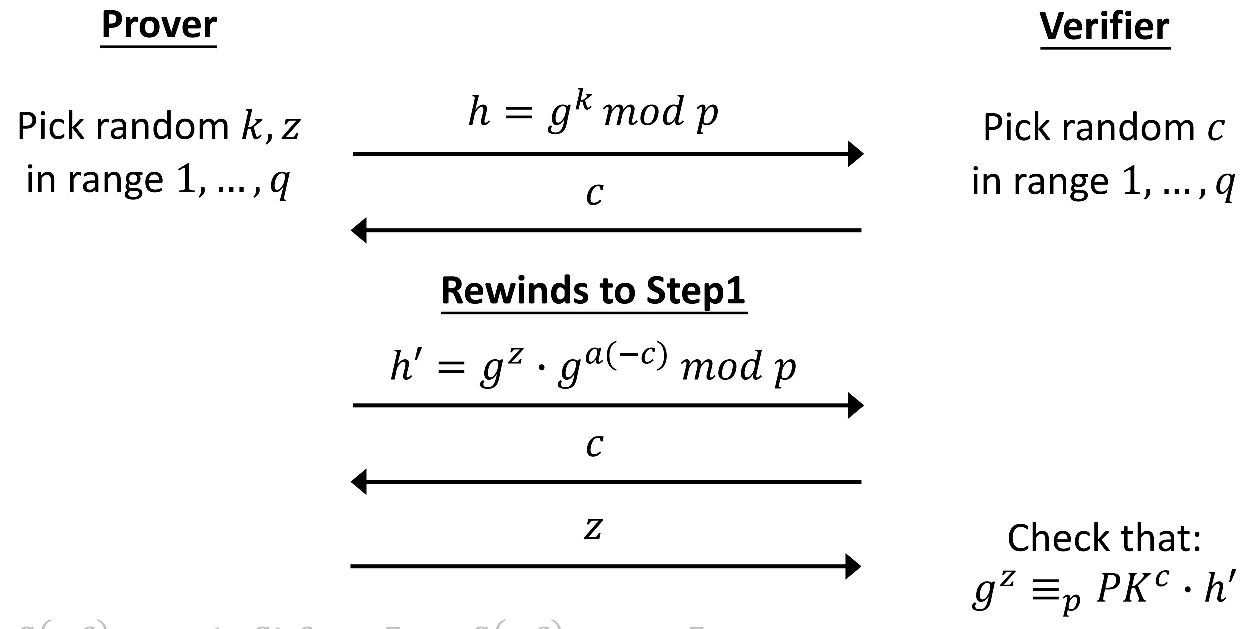
c_2

$z_2 = a(c_2) + k \; mod \; q$

Extract the secret by
$a \equiv_q (z_1 - z_2)/\,(c_1 - c_2)$

Proving the Zero-Knowledge(ness)

- Zero-Knowledge(ness) in this context: The proof z reveals nothing useful for the verifier to guess SK.
- Method: use an **simulator** (i.e. She-Hulk) to cheat on the verifier.



Prover

Verifier

Pick random k, z in range $1, \ldots, q$

$h = g^k \, mod \, p$

c

Pick random c in range $1, \ldots, q$

Rewinds to Step1

$h' = g^z \cdot g^{a(-c)} \, mod \, p$

c

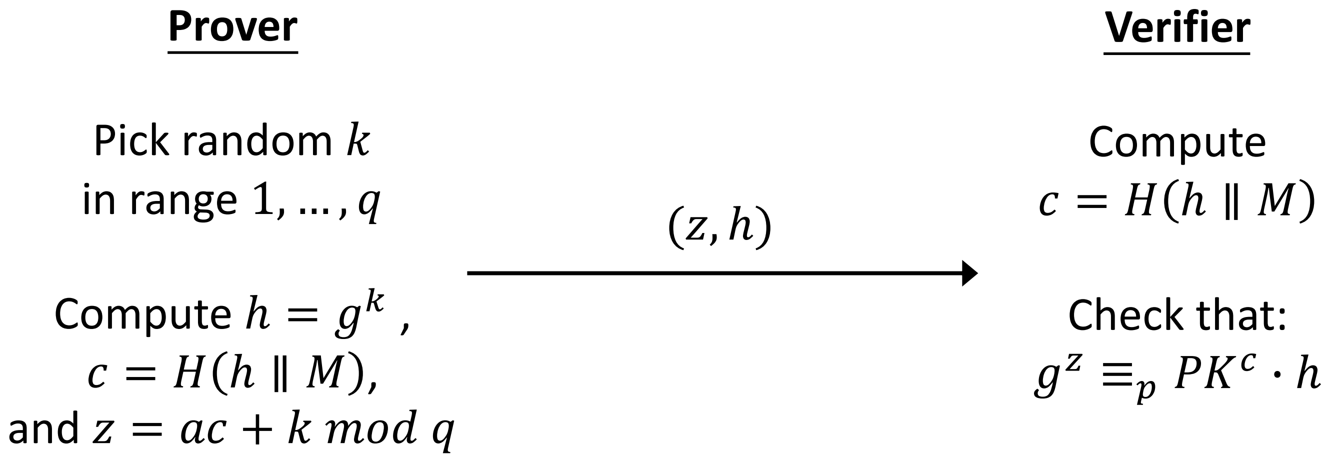
z

Check that:
$g^z \equiv_p PK^c \cdot h'$

$PK^c \cdot h' \equiv_p PK^c \cdot g^z \cdot g^{a(-c)} \equiv_p (g^a)^c \cdot g^z \cdot g^{a(-c)} \equiv_p g^z$

Non-Interactive Version

- The prover computes the challenge as $c = H(h \parallel M)$.
  - $H()$ is a **public** random oracle (i.e., idealized cryptographic hash function).
  - M is a **public** and arbitrary message string.
- Note that this is also a signing scheme for message M.



Prover

Pick random k
in range $1, \dots, q$

Compute $h = g^k$,
$c = H(h \parallel M)$,
and $z = ac + k \; mod \; q$

(z, h)

Verifier

Compute
$c = H(h \parallel M)$

Check that:
$g^z \equiv_p PK^c \cdot h$

"… (with the advent of social media) All of a sudden people felt I can share everything with everyone, and I feel that now we're seeing the repercussions of that… Basically things that previous generation learned, this generation is now learning 'oh wait, sharing everything, there are problems with that, there are unforeseen consequence of having all of your information out there.'" —— ZKPodcast (31:00)

Reference

- A survey on zero knowledge range proofs and applications
- What are zero-knowledge proofs?
- 探索零知识证明系列1 - 初识「零知识」与「证明」
- ZKPodcast: Dan Boneh on the past, present & future of cryptography
- Zero Knowledge Proofs: An illustrated primer