Zero Knowledge Proofs

ROUNAK DAS

What is that?

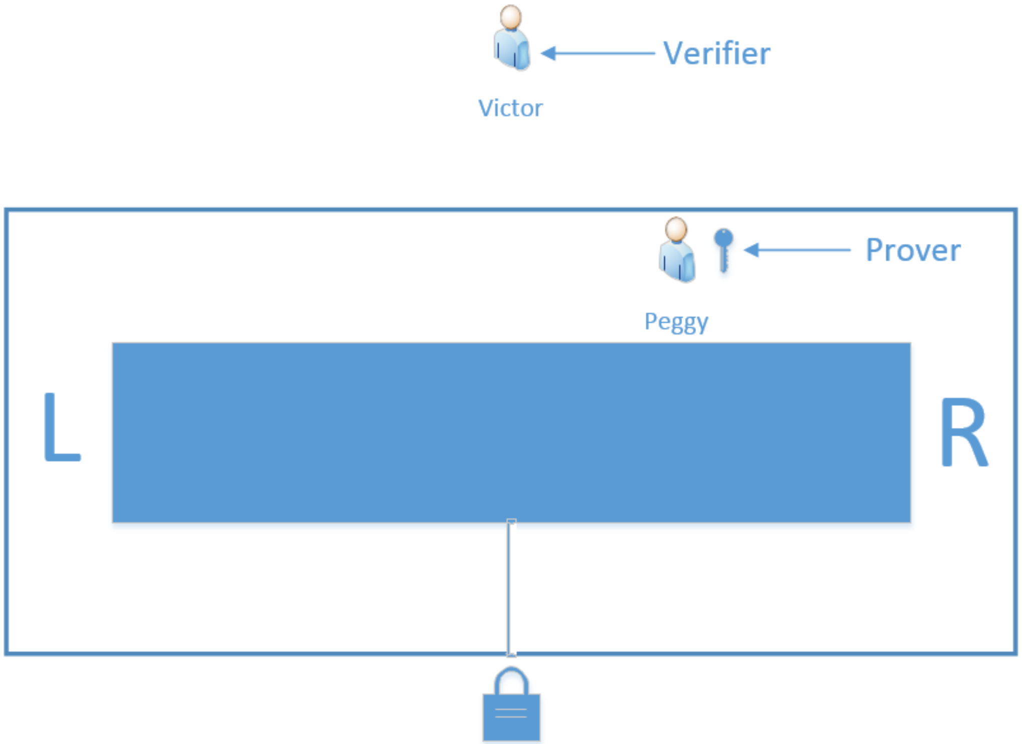
- An interactive proof system
  - Prover
  - Verifier
  - Messages: Commitment, Challenge and Response
  - Verify response

Fig: An Interactive proof System

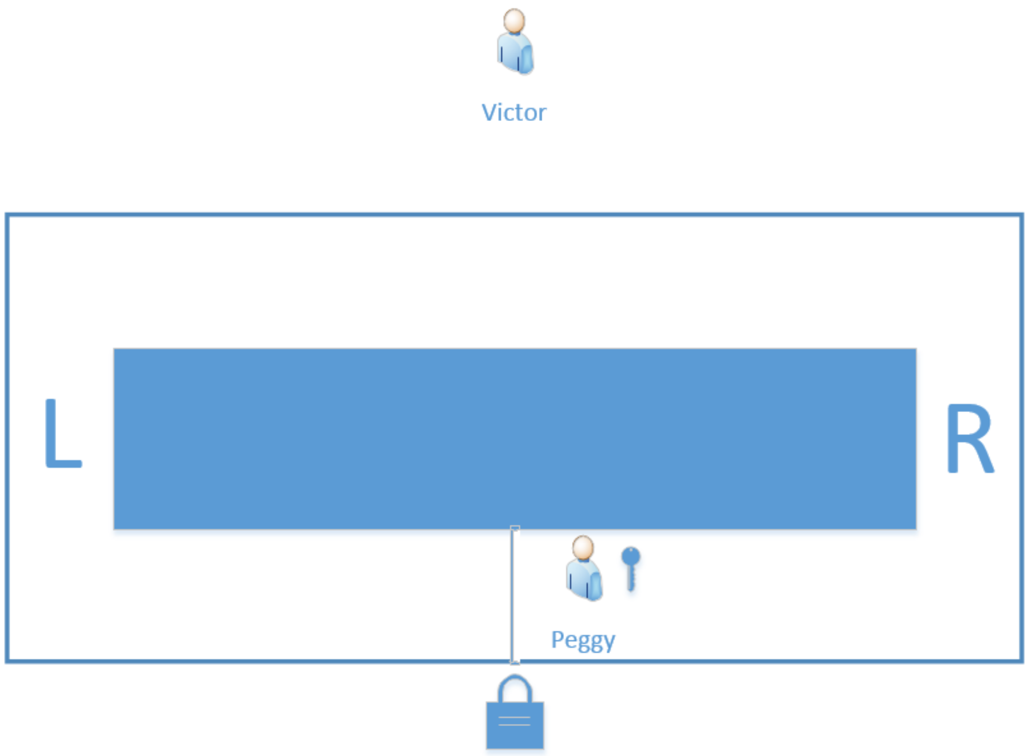
What is that?

- Zero Knowledge Interactive Proofs
  - 1985, Goldwasser, Micali and Rackoff
  - Extension of Interactive Proofs
- Interactive Proofs may leak the information being proved
  - Prove: 26781 is not a prime
  - $26781 = 113 \times 237$
  - But now the verifier knows the factorization!
  - ZKPs try to convince without revealing
- Probabilistic
  - Always a non-zero probability that the Prover just guessed
  - But typically very small

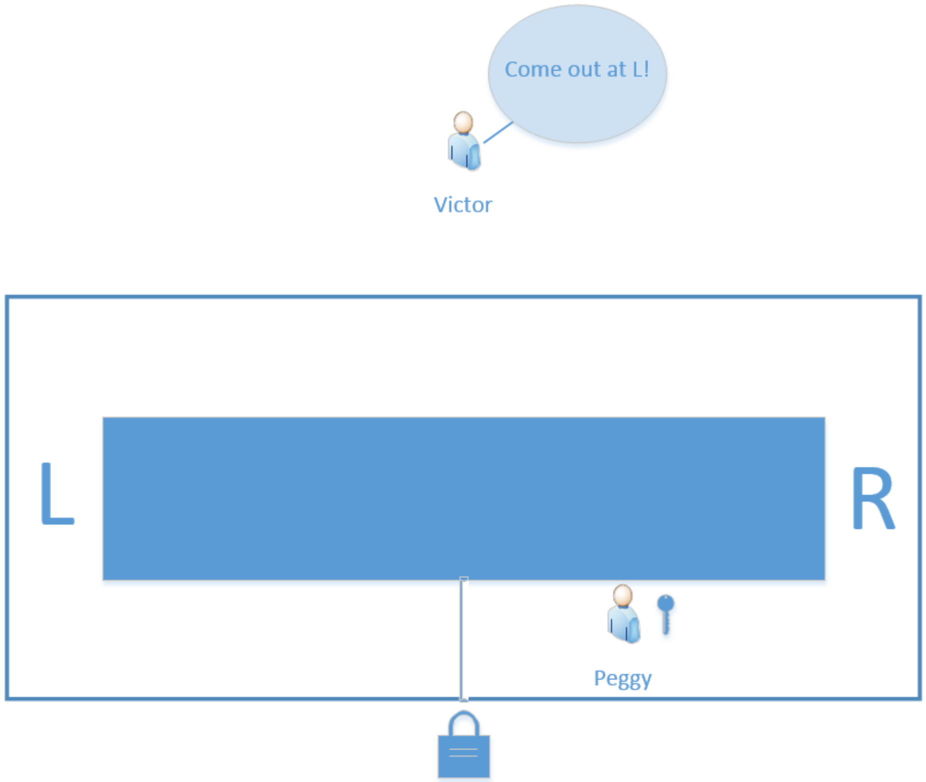
Enter The Cave



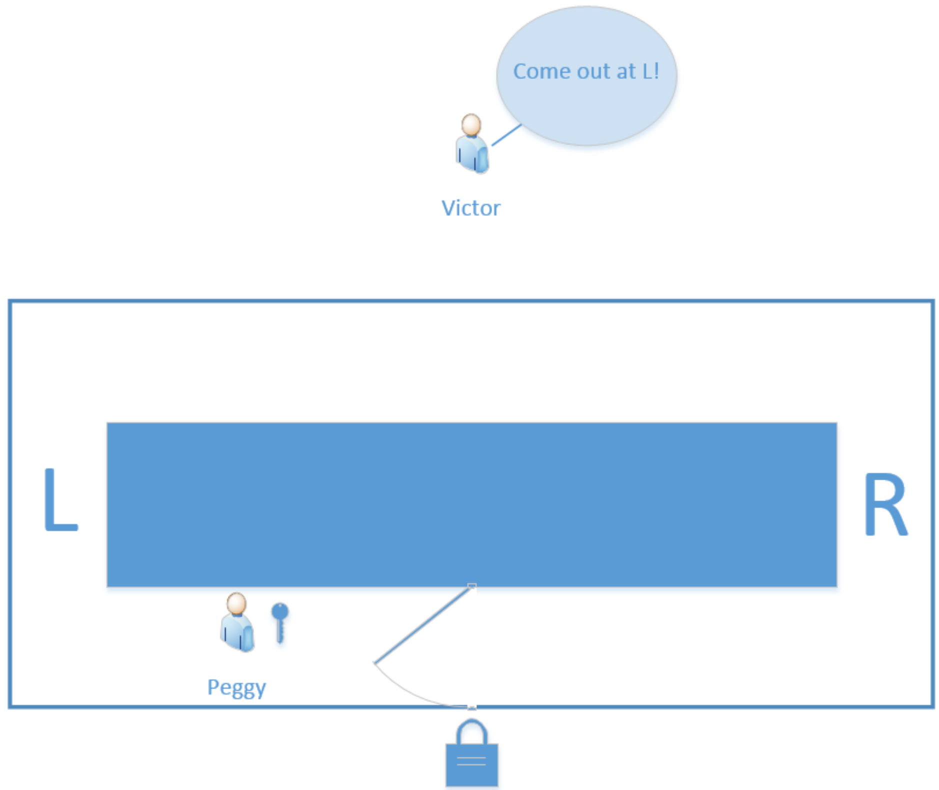
The Commitment



The Challenge



The Response



Many Rounds Later…

- 1 round
  - Cheating probability: ½
  - 50% convinced
- 2 rounds
  - Cheating probability: ¼
  - 75% convinced

…

- 40 rounds
  - Cheating probability: one in a million million
  - Highly convinced

Essential Properties

- ## Completeness
  - If the statement is true, then the verifier will be convinced of it.
- ## Soundness
  - If the statement is false, then cheating provers cannot convince the verifier that it is true.
- ## Zero Knowledge
  - If the statement is true, then no cheating verifier learns anything except that the statement is true.

Zero Knowledge?

- How do we know a protocol is zero-knowledge?
- Does it leak the secret?
- Can we simulate a proof without the secret?
- Prover and Verifier collude
- Make fake transcripts
  - Edit out failures
  - Have preplanned sequences
- Fakes are indistinguishable from original!
- Knowledge of secret not required – original proof did not leak it!

Practical Examples

- Different Types:
  - Proof of Membership
  - Proof of knowledge
  - Proof of Identity
  - Computational Zero Knowledge
  - Perfect Zero Knowledge
- Fiat-Shamir Identity Scheme
- Schnorrs Scheme
- Guillou-Quisquarter
- Graph Isomorphisms
- Graph 3-colorings

Example: Fiat-Shamir

- Based on the Quadratic Residue problem
  - Computing x, given $x^2 \pmod{n}$ is hard if factorization of n is not known

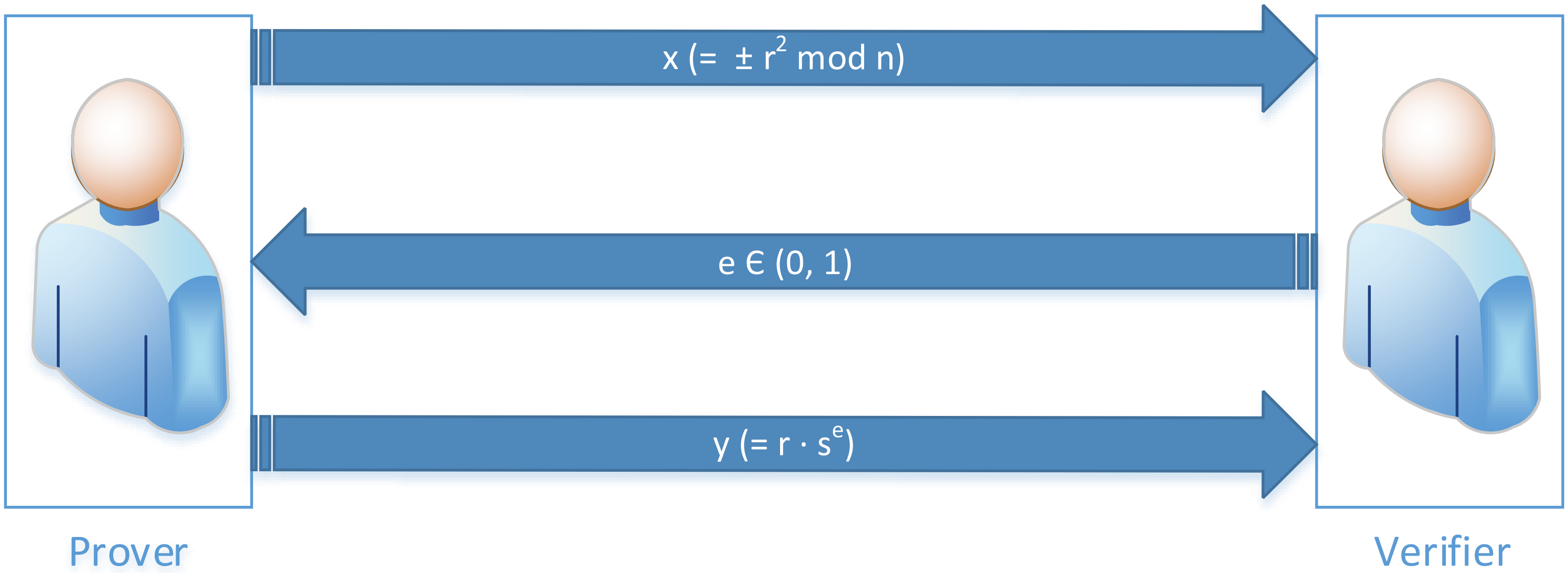
- Setup:
  - Requires trusted central authority ***T***
  - ***T*** selects ***n*** s. t. it is a Blum integer
    - $n = p \cdot q$
    - ***p*** and ***q*** are kept secret
  - Select ***t*** (number of rounds)

Example: Fiat-Shamir

- Each entity computes their key-pair
  - Choose secret key
    - Choose ***s***
    - $1 \leq s \leq n-1$, $gcd(s, n) = 1$
    - Usually use k-vectors instead of just one value.
  - Calculate public key
      - Compute ***v***
      - $v = 1/s^2 \pmod{n}$
    - Publish ***v***
    - Keep ***s*** secret

Example: Fiat-Shamir

- Commitment, Challenge and Response:



- Verifier checks if: $x == (\pm y^2 \cdot v)$

Example: Fiat-Shamir

- Challenge-Response continues for ***t*** rounds
  - Probability of successfully cheating: 2^{-t}
- Zero Knowledge Proof?
  - Complete
    - Prover knows ***s***, can compute both ***y = r*** (***e=0***) and ***y = rs*** (***e=1***) easily
    - Verifier is always convinced
  - Sound
    - Prover doesn't know ***s***, can only compute either ***y = r*** or ***y = rs*** (by choosing $x = r^2/v$)
    - Needs to know Verifiers choice in advance or be able to compute square root!
  - Zero Knowledge
    - Only things revealed are $x = r^2$ mod n and either ***y = r*** or ***y = rs***
    - Can simulate by defining $x = y^2$ or $x = y^2/v$
    - Indistinguishable!

Applications and Attacks

- Applications:
  - Digital signature schemes (Fiat-Shamir heuristic)
  - e-voting (honest behavior in a mix-net)
  - Anonymous auctions
  - … and many more

- Attacks:
  - Man-in-the-Middle
  - Impersonation
  - Replay attack

References

- Goldwasser, Micali, Rackoff “The Knowledge Complexity of Interactive Proof Systems” (1985)
- Feige, Fiat, Shamir “Zero Knowledge Proofs of Identity” (1989)
- Guillou, Quisquater “How to Explain Zero-Knowledge Protocols to your Children” (1998)
- Menezes, Oorschot; Chapter 10 from “Handbook of Applied Cryptography” (1996)
- Trappe, Washington; Chapter 14 from “Introduction to Cryptography with Coding Theory” (2002)

Thank you!

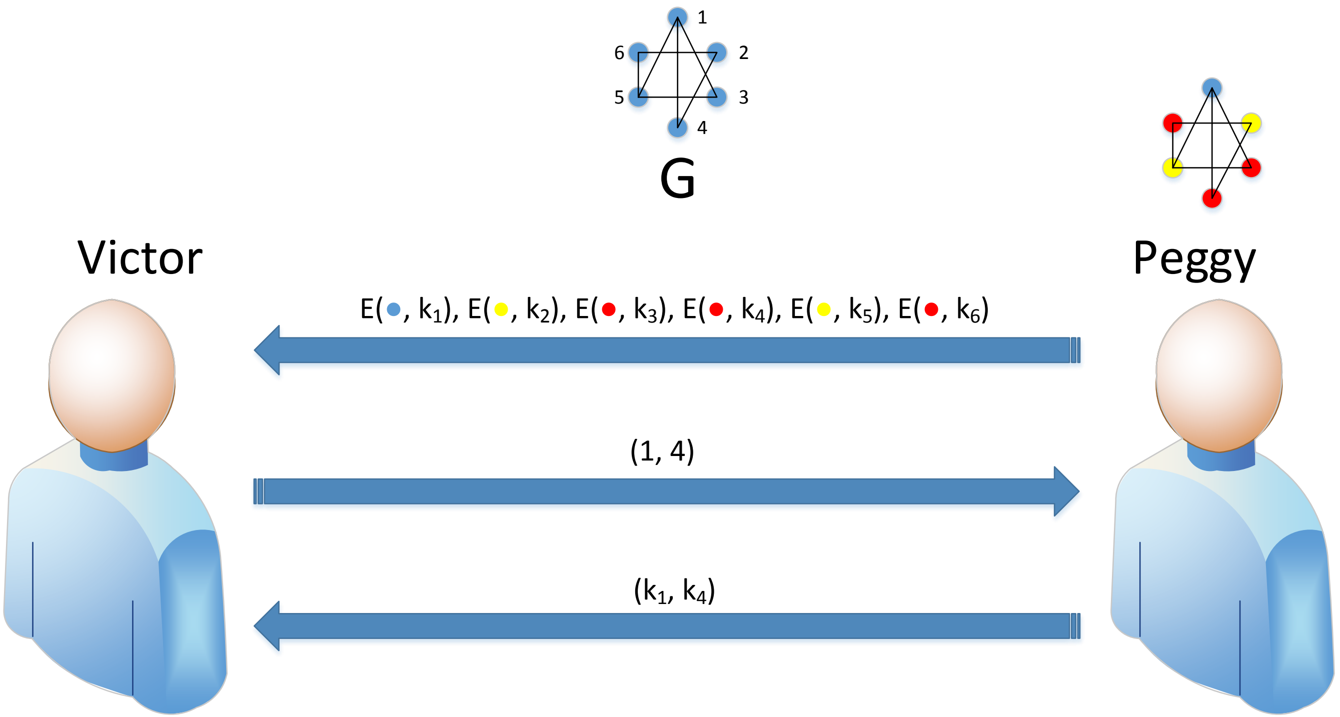
Bonus: Fiat-Shamir Example

- p = 683, q = 811 so that n = 553913
- 3 challenges per round: k = 3, single round: t = 1
- Alice selects key-pair:
  - s_1 = 157, s_2 = 43215, s_3 = 4646 (private key)
  - b_1 = 1, b_2 = 0, b_3 = 1
  - v_1 = 441845, v_2 = 338402, v_3 = 124423 (public key)
- Challenge Response:
  - Alice chooses r = 1279, x = 25898; sends this to Bob
  - Bob sends back 3-bit vector: (0,0,1)
  - Alice computes response: $y = r \cdot s_3 \bmod n$ = 403104
  - Bob verifies: $y^2 v_3 \bmod n$ = 25898 = x => ***Accept!***

Bonus: Graph 3-Coloring

- Checking if a graph is 3-Colorable is hard
  - Also hard to 3-Color a graph
- Peggy: "I have a 3-Coloring for graph ***G***!"
- Victor: "Prove it!"
- Both parties know ***G*** and the vertex labels ***i*** ($1 \leq i \leq |G.V|$)
- Commitment: $E(color(v_i), k_i)$ for each vertex v_i
  - k_i is particular to v_i for this round
  - Peggy chooses new keys next round
- Challenge: (i,j) where v_i and v_j are adjacent ($1 \leq i, j \leq |G.V|$)
- Response: k_i and k_j
- Verify: $D(color(v_i), k_i) \neq D(color(v_j), k_j)$

Bonus: Graph 3-Coloring



Verify:

1. D(E(●, k_1)), k_1 = ●
2. D(E(●, k_4)), k_4 = ●
3. ● ≠ ●
4. Accept!