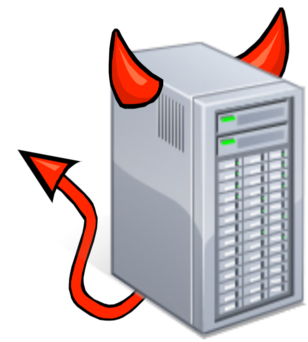
Physical Zero-Knowledge Proofs of Physical Properties

Ben Fisch

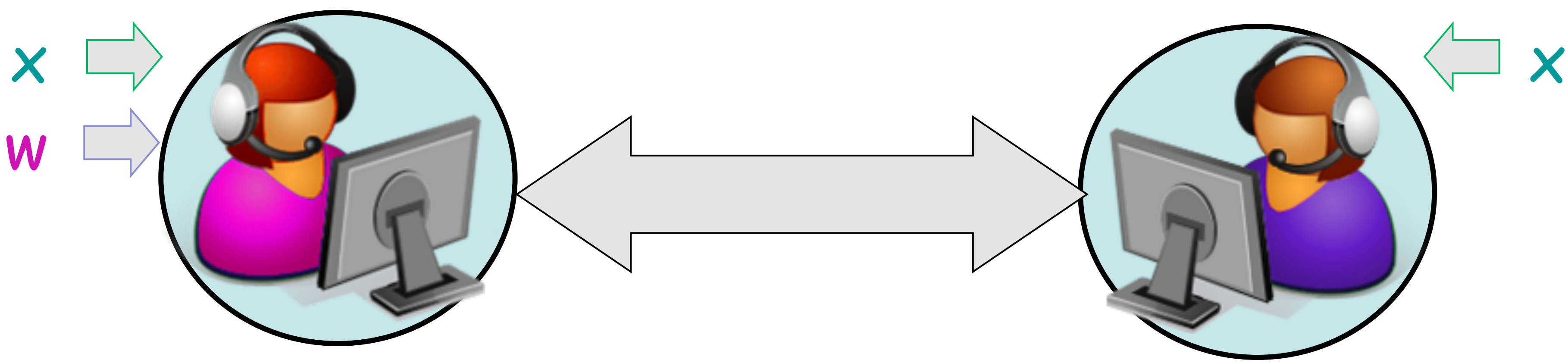
Joint work with: Moni Naor and Daniel Freund

TCC Rump Session, February 2014

Typical Zero-Knowledge Scenario



- Alice and Bob receive input x
- Alice has input w
- Alice wants to convince Bob that
  - There is a w such that $R(x,w) = 1$
  - She knows such a w
- Alice and Bob exchange messages



What if R is a physical property?

- Suppose the input x is physical, and R is a physical property Π
- There is a **physical measurement** M that verifies: Π (x) = 1, i.e. "x has property Π"
- Can Alice convince Bob without revealing anything more about x?

More difficult to formalize the Zero-Knowledge property

Simple Example

Alice claims she can distinguish Coke from Pepsi:

Bob selects randomly from {Coke, Pepsi}

Is it Zero-Knowledge??

What happens if Alice gets a mixture?

Which one did I give you?

? ? ?

Um..Coke!

If Alice **cannot** distinguish, she succeeds only with probability 1/2

Repeat t times

Probability Alice succeeds is $1/2^t$

Setting not Inherently physical

Related Work

- Physical techniques for aiding cryptographic protocols
  - **Tamper-proof tokens, tamper-evident seals (envelopes), physically uncloneable functions, more examples...[GO96, GLM+04, MS08, HL08, GIS+10, GKR08, BFSK11]**
- Can we find simple cryptographic protocols that humans can physically implement unaided?
  - **Visual Cryptography [Naor-Shamir'94], Applied Kid Cryptography [Naor-Naor-Reingold'99], Computations with a Deck of Cards [Stiglic'01], Zero-Knowledge for Sudoku Puzzles [Gradwohl-Naor-Pinkas-Rothblum'09]**
  - It's hard to see what's going on inside a computer
  - **Very relevant to voting!**
    - **Polling with envelopes [Moran-Naor'06]**

Inherently physical setting

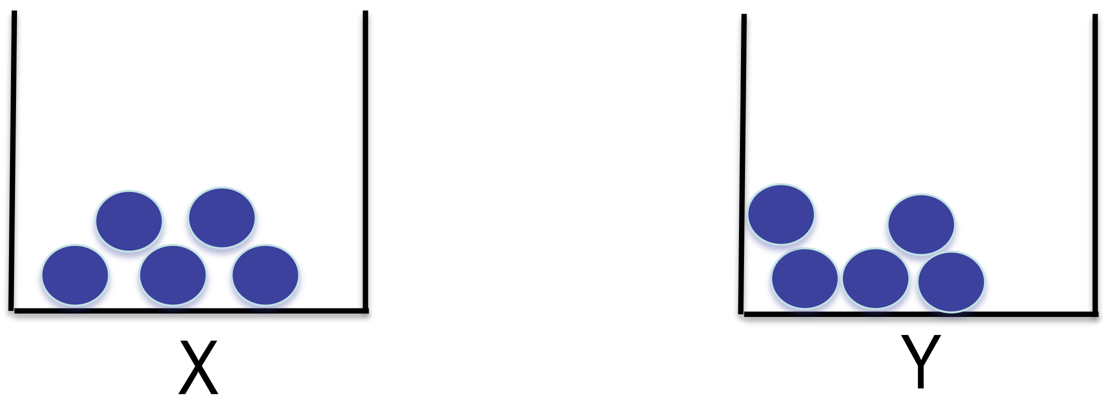
Related Work

- Distance bounding protocols [Brands-Chaum'93]
  - Prove that you are close to a certain location
  - Use timing (speed of light)
- Boaz Barak, Alex Glaser, and Rob Goldston [GBG12] applied a zero-knowledge style technique to nuclear warhead verification
- Inherently *physical*. Not just using physical tools to construct a low complexity solution to a digital problem.

Nuclear Warhead Verification

- Nuclear Disengagement: plan to reduce nuclear weapon stockpiles worldwide.
- START treaty, Russia and US
  - Alice promises to dismantle some of her warheads
  - How does Bob know that Alice's warhead is authentic?
  - Can Alice ensure that Bob doesn't learn, (too, much) about the design of her warhead?
- Barak et. al. reduce the problem to a protocol for Bins and Balls

Bins and Balls



Do bins X and Y contain the same number of balls?

This Work

- Paradigm for formally defining, modeling, analyzing physical zero-knowledge protocols
- Nuclear Disarmament: perfect physical zero-knowledge proofs for arms-control
  - Barak et. al. gave ε-knowledge
- DNA Privacy: zero-knowledge proofs for DNA profiling

Modeling physical protocols

- Separate into *logical layer* and *physical layer*
- *Physical layer:* Physical operations assumed to achieve ideal functionalities (physical assumptions)
- *Logical layer:* Hybrid world protocol obtained by replacing all physical operations with calls to their ideal functionalities.

Modeling Example

Operation: pour *x* balls into a bin, and seal it

- $\mathcal{T}$ stores tuples (value, id, creator, holder, state)
- Upon receiving commands **Create**(x, id) and **Seal** (id) from party P_i, $\mathcal{T}$ stores (*x, id,* P_i, P_i, *sealed*)
- $\mathcal{T}$ only accepts **Open**(id) from the holder
- **Force**(id) causes $\mathcal{T}$ to return entire tuple of *id*, and send the message "cheater" to all parties

Emulates real behavior of party that forcefully breaks open the seal without permission

Ideal functionality ZK^{Π}

- Oracle access to ideal functionality M^{Π}
- Obtains "access" to input x
- Queries M^{Π} with input x
- Outputs $\Pi(x)$ to Verifier

Measurement verifying Π

Full definition accounts for cheating

Security: Show that the *logical layer* (hybrid world translation of physical protocol) emulates ZK^{Π}

Differences from standard ZK

- No witness
  - Asymmetry between Prover and Verifier is in *access permission*, not secret knowledge or computational resources
- Ideal functionality performs verification on its own
  - It is given access permission to the input
  - Normally, Prover is required to supply a witness
- Verifier can forcefully cheat
  - Similar to covert model

Physical ZK in UC framework

- **Benefits:**
  - Modular design and analysis of physical protocols
  - Arbitrary composition of physical and computational subprotocols
- **Feasibility:**
  - Sim does not need to do a straight-line extraction of a witness from the real world prover

Public coin and publicly executable proofs

- *Public-coin protocols*:
  - In public-coin protocols, the verifier's messages consist only of public coin flips
  - Public-coin physical protocols are **publicly executable**
  - The verifier can sit behind a glass screen throughout the execution

Makes a huge difference for physical security!

- **We construct a publicly executable DNA inequality protocol**

Feasibility of publicly executable proofs?

- In the standard digital setting, public-coin ZK = private-coin ZK [Oka96, GSV98, GV99, Vad04]
- General result for physical zero-knowledge?
- Techniques for explicit conversions of private-coin protocols to public-coin protocols don't translate well in the physical setting
  - Universal hashing of physical messages?
  - Physically concealed messages

Public coin => publicly executable
Publicly executable =>? public coin

Summary and Further Research

Physical Cryptography is

- Relevant
- Fun
- **Structured**(?): connections with known crypto/complexity techniques
- Many foundational questions remain