Hidden Diversity and Secure Multiparty Computation

Juan A. Garay
AT&T Labs – Research

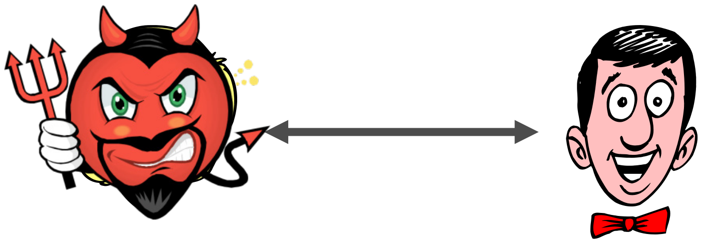
With D. Johnson (AT&T), A. Kiayias (U. Athens) and M. Yung (Google)

Adversaries and Cryptography

- Computing in the presence of an adversary is at the heart of modern cryptography
- “Completeness theorems” for distributed cryptographic protocols:
  - An adversary controlling any *minority* of the parties cannot prevent the secure computation of *any efficient functionality* defined over their inputs [Yao82, GMW87]
  - Similar results hold over secure channels (and no add’l crypto) with an (computationally unbounded) adversary controlling less than a *third* of the parties [BGW88, CCD88]

Resource-based Corruptions

- Adversaries *corrupt* parties...



...for FREE!

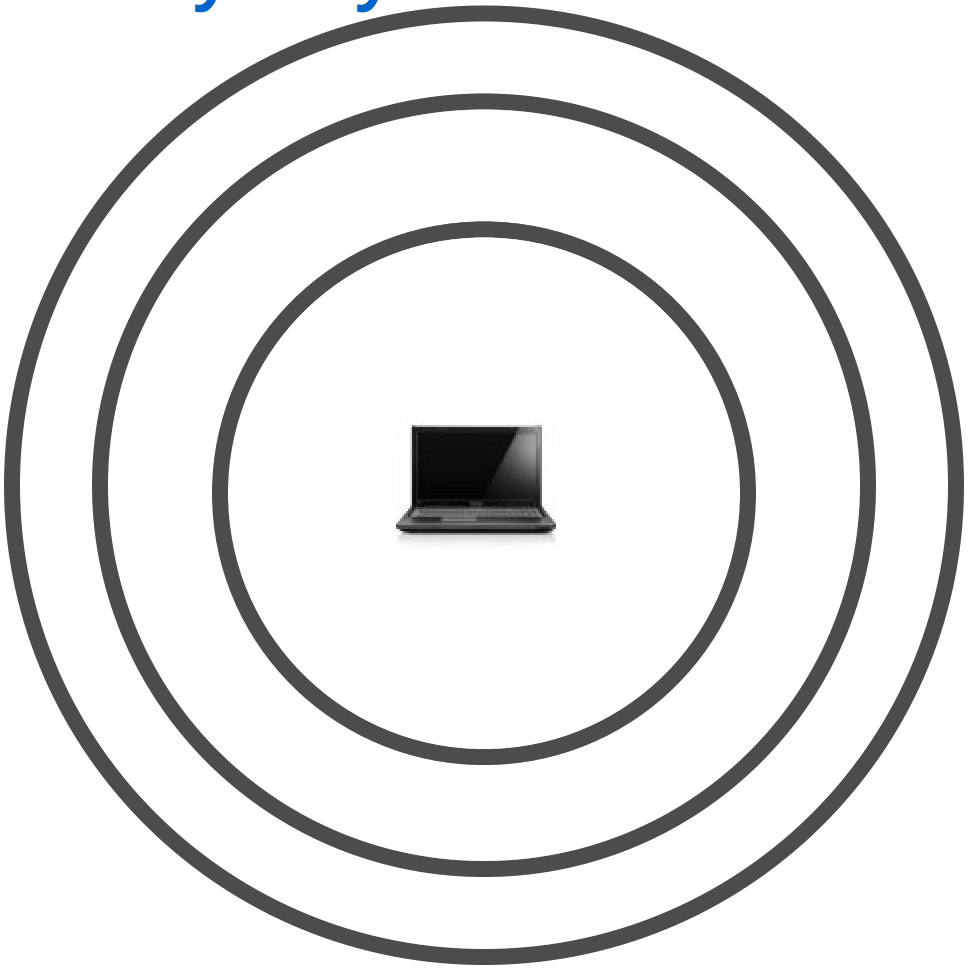
- Corrupted party does not necessarily follow protocol – in addition to trying to find the secrets of other parties, it may aim to disrupt the computation so it results in an incorrect answer

Resource-based Corruptions (cont'd)

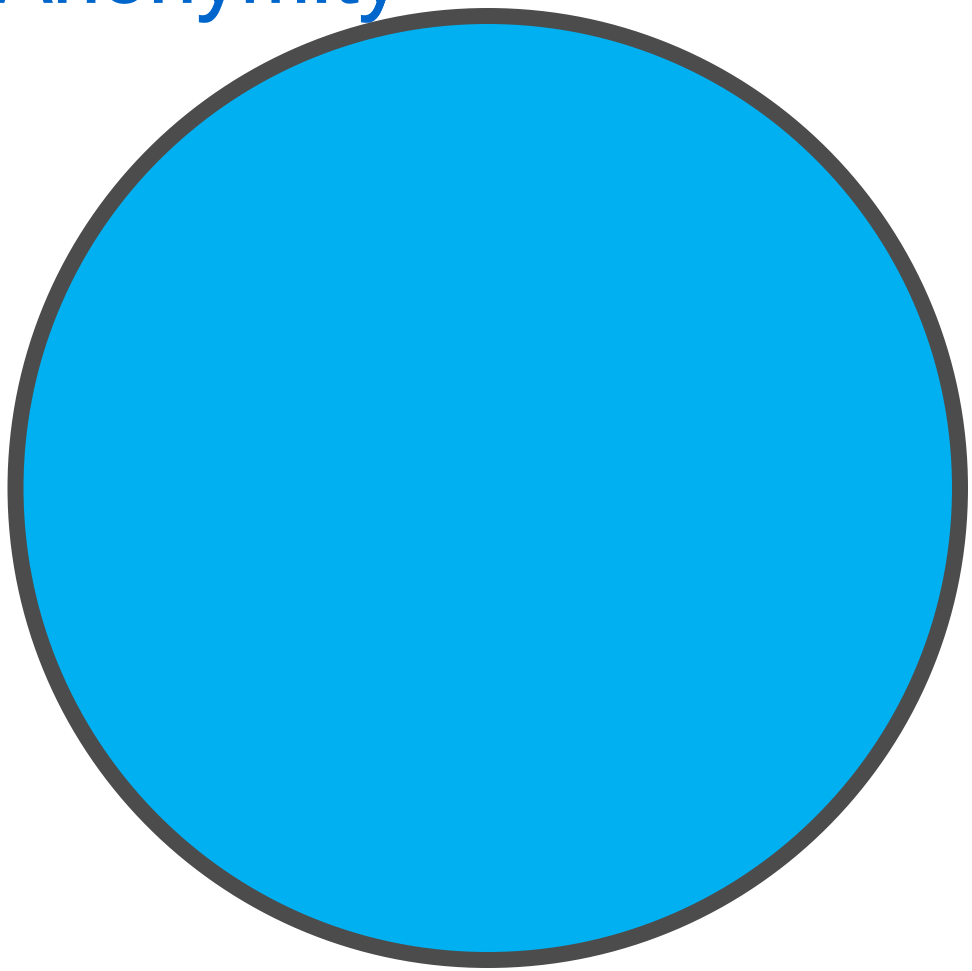
Our new questions:

- How does an adversary turn a law-abiding party into a malicious saboteur?
- Bribe them, hack them, …?
- How much does it cost?
  - Different parties may require different "resources" to get corrupted
- Can "anonymity" be used to raise those costs?

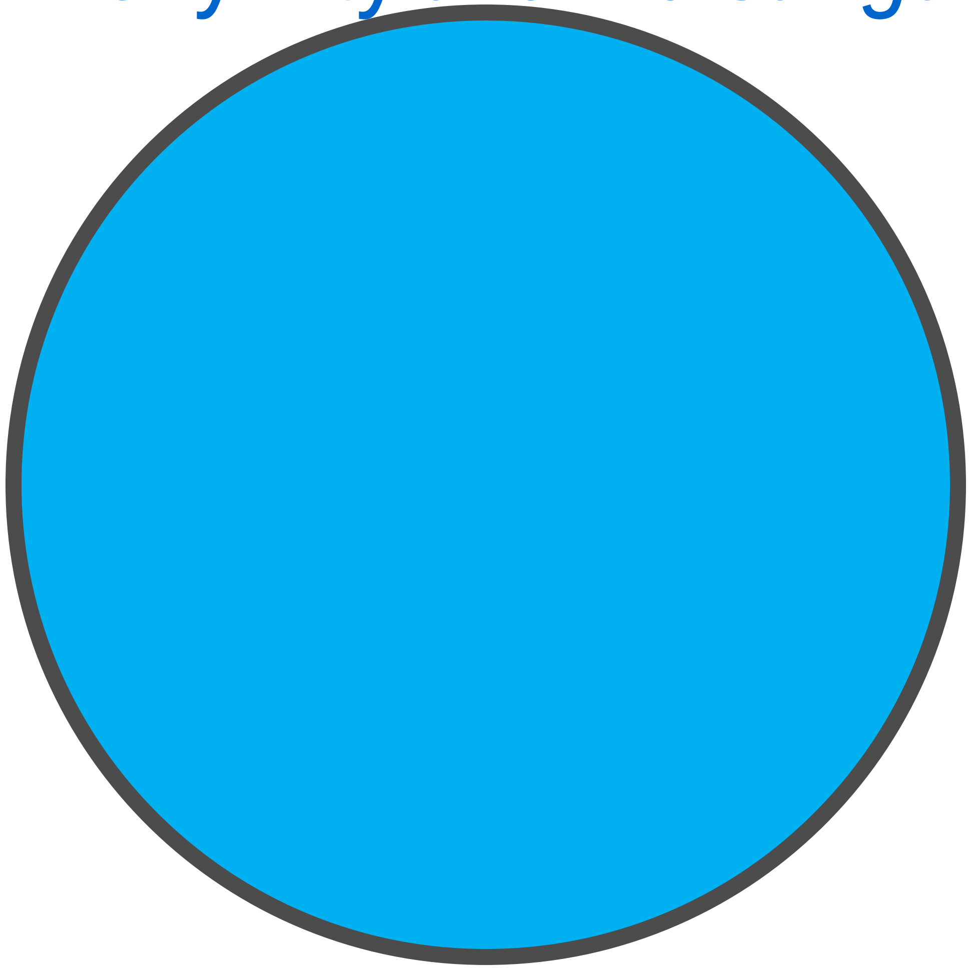
Resource Anonymity



Resource Anonymity



Resource Anonymity *and Indistinguishability*



A Combinatorial Game

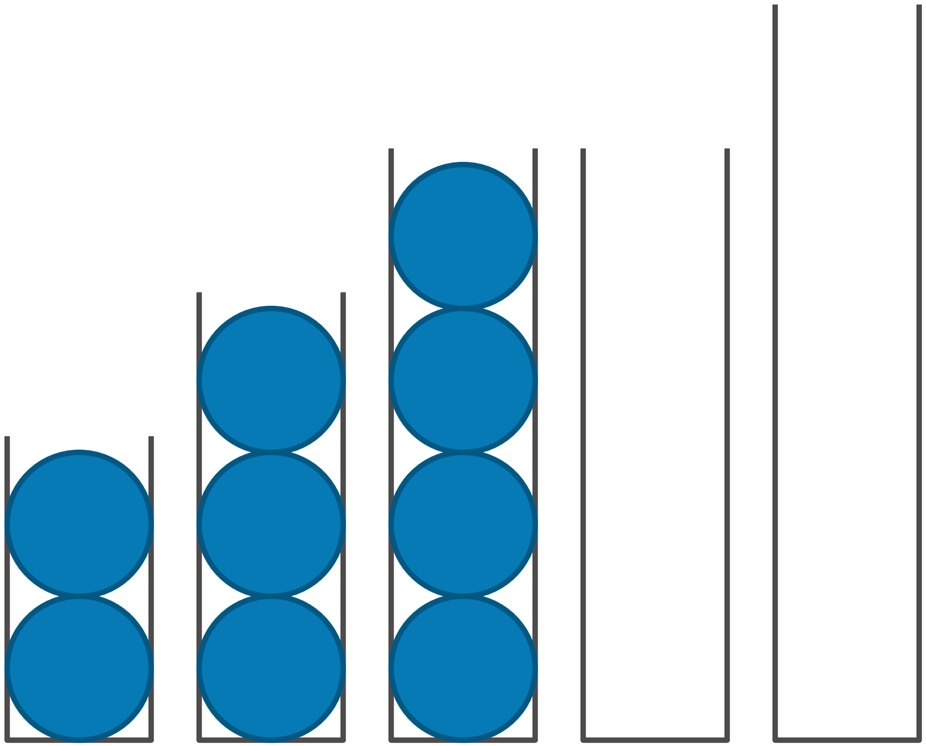
- GIVEN: Set $B_1, B_2, \ldots, B_n$ of buckets, with bucket B_i having non-negative integer size s_i, and a target fraction α, $0 < \alpha < 1$.

- GOAL: Fill $\lceil \alpha n \rceil$ of the buckets using as few balls as possible, where a bucket of size s_i is filled if it receives s_i balls.

Balls and Buckets

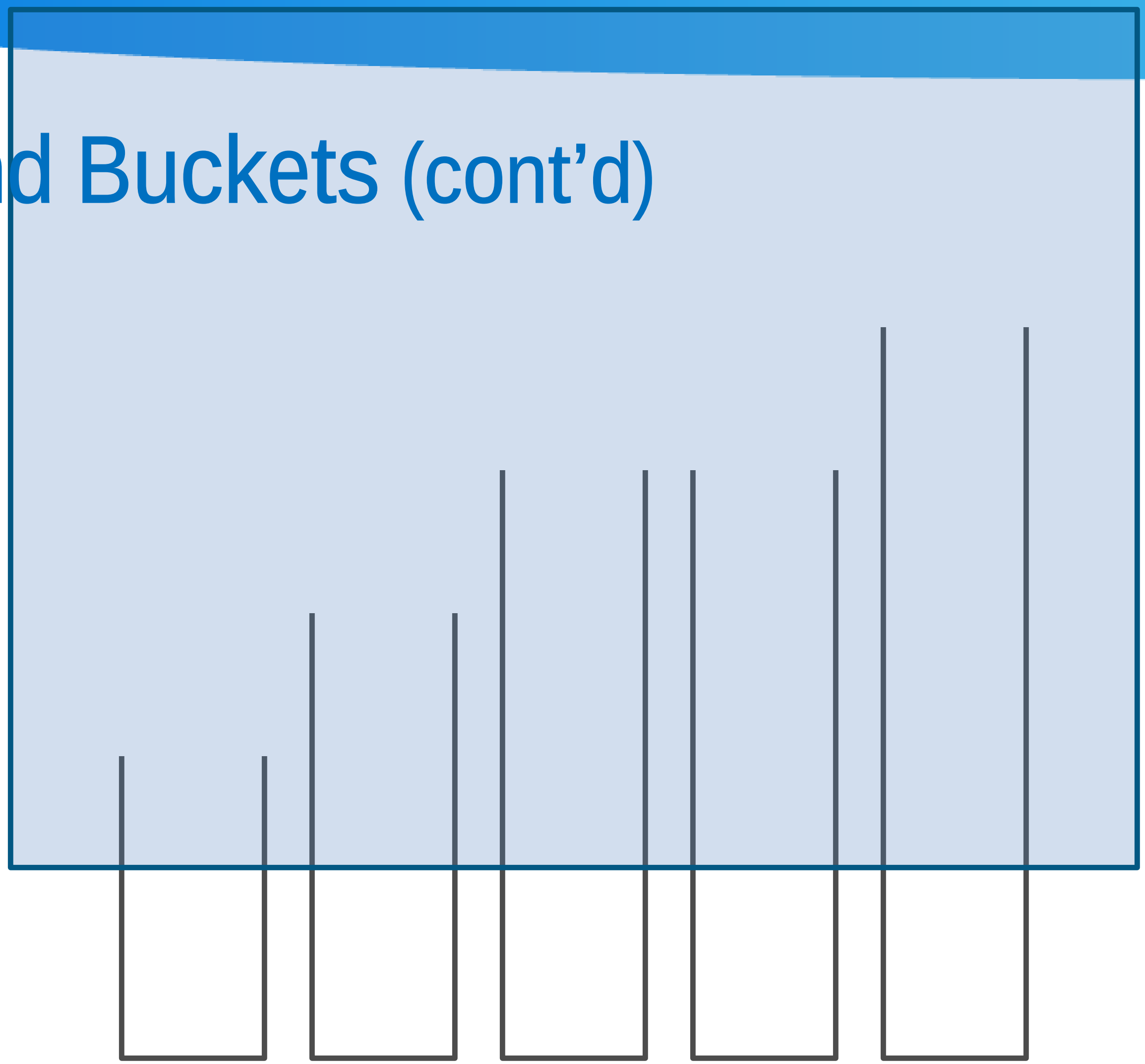
- Buckets = Participants in the protocol
- Bucket size = Number of *corruption tokens* required to break into the participant's machine and take it over
- Ball = corruption token
- Adversary = placement algorithm
- α = 1/2, 1/3, …

Balls and Buckets (cont'd)



$n = 5, \alpha = ½, \lceil \alpha n \rceil = 3$

Balls and Buckets (cont'd)



Only Feedback from Placing a Ball:
"Bucket Now Full" or

"Bucket Not Yet Full"

How many balls?

States of Ignorance

Adversary knows:

- Only n [No-Information]
- n and $\max\{s: s = s_i \text{ for some } i\}$ [Max-Only]
- $\{s: s = s_i \text{ for some } i\}$ [Sizes-Only]
- $\{(s,k): |\{i:s_i = s\}| = k > 0\}$ [Profile-Only]
- $s_1, s_2, \ldots, s_n$ in order [Full-Information]

States of Ignorance

Adversary knows:

- Only n [No-Information]
- n and $\max\{s: s = s_i \text{ for some } i\}$ [Max-Only]
- $\{s: s = s_i \text{ for some } i\}$ [Sizes-Only]
- $\{(s,k): |\{i: s_i = s\}| = k > 0\}$ [Profile-Only]
- $s_1, s_2, \ldots, s_n$ in order [Full-Information]

Evaluating Adversary's Cost: Notation

- Instance: $\mathbf{s} = (s_1, s_2, \ldots, s_n)$
- $\mathrm{Opt}_\alpha(\mathbf{s}) = \min(\sum_{i \in C} s_i : C \subseteq \{1,2,\ldots,n\} \text{ and } |C| = \lceil \alpha n \rceil)$
- $A_\alpha(\mathbf{s})$: number of balls used by (deterministic) algorithm A when it has filled $\lceil \alpha n \rceil$ buckets, when the bucket sizes are hidden

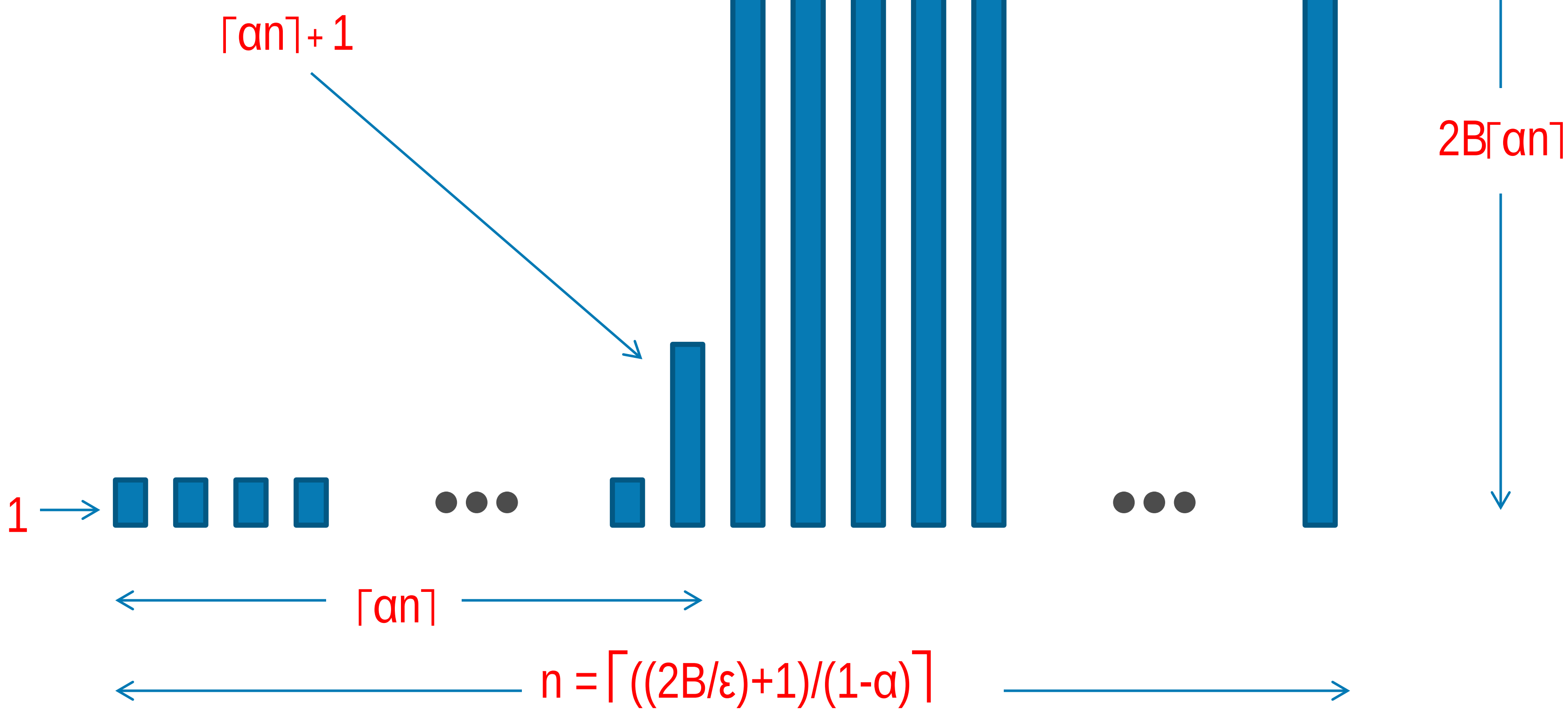
Some Initial Good News (Bad News for the Adv.)

Theorem: For any profile-only adversary A, and any constants α, $0 < \alpha < 1$, $B > 1$, and $\varepsilon > 0$, there exist instances **s** such that

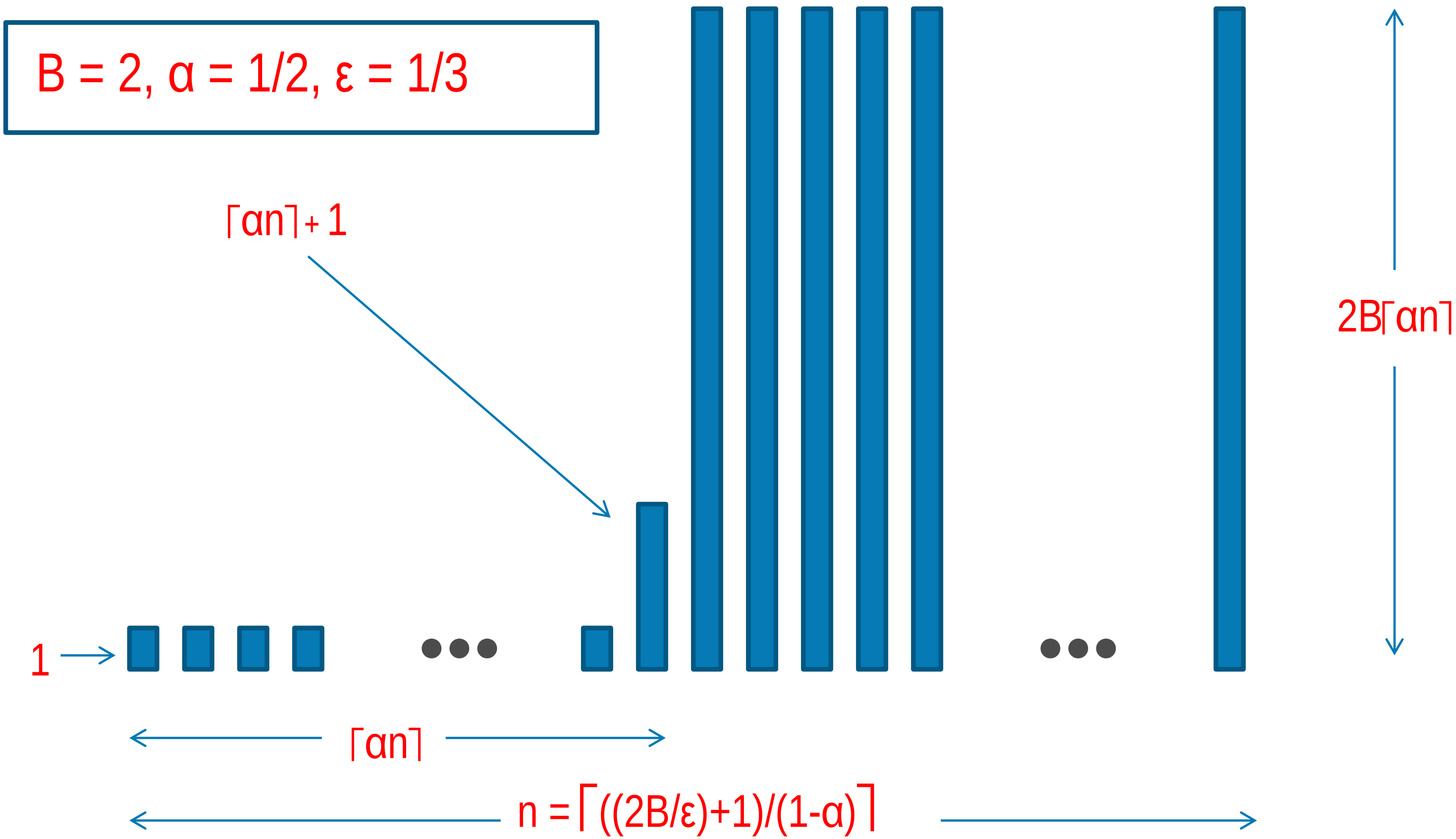
$$\Pr\left[A_\alpha(\mathbf{s}) < B \cdot Opt_\alpha(\mathbf{s})\right] < \varepsilon$$

Proof by Picture (not to scale):

$$\mathrm{OPT}_\alpha(s) = 2\lceil \alpha n \rceil$$

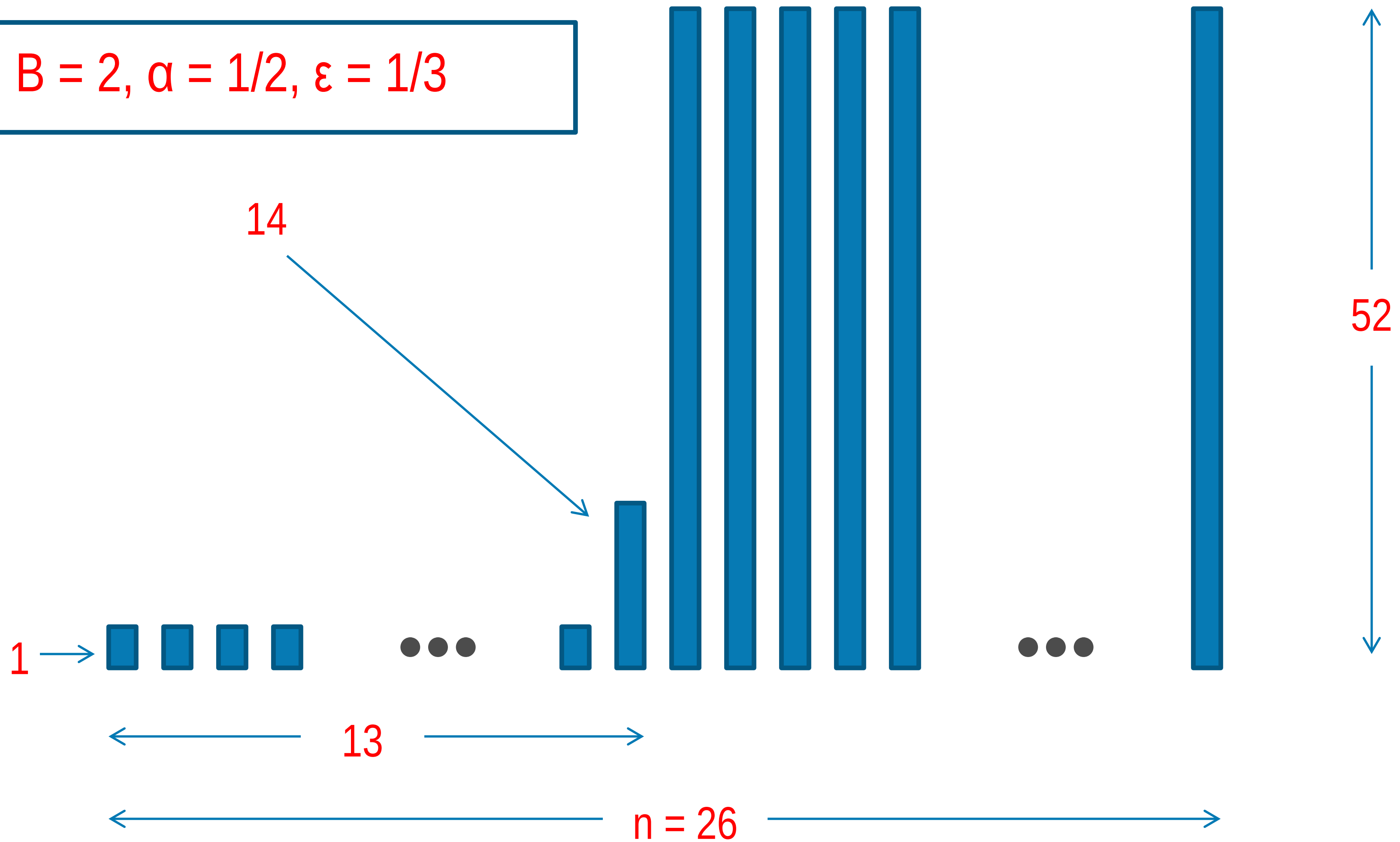


Proof by Picture (not to scale):



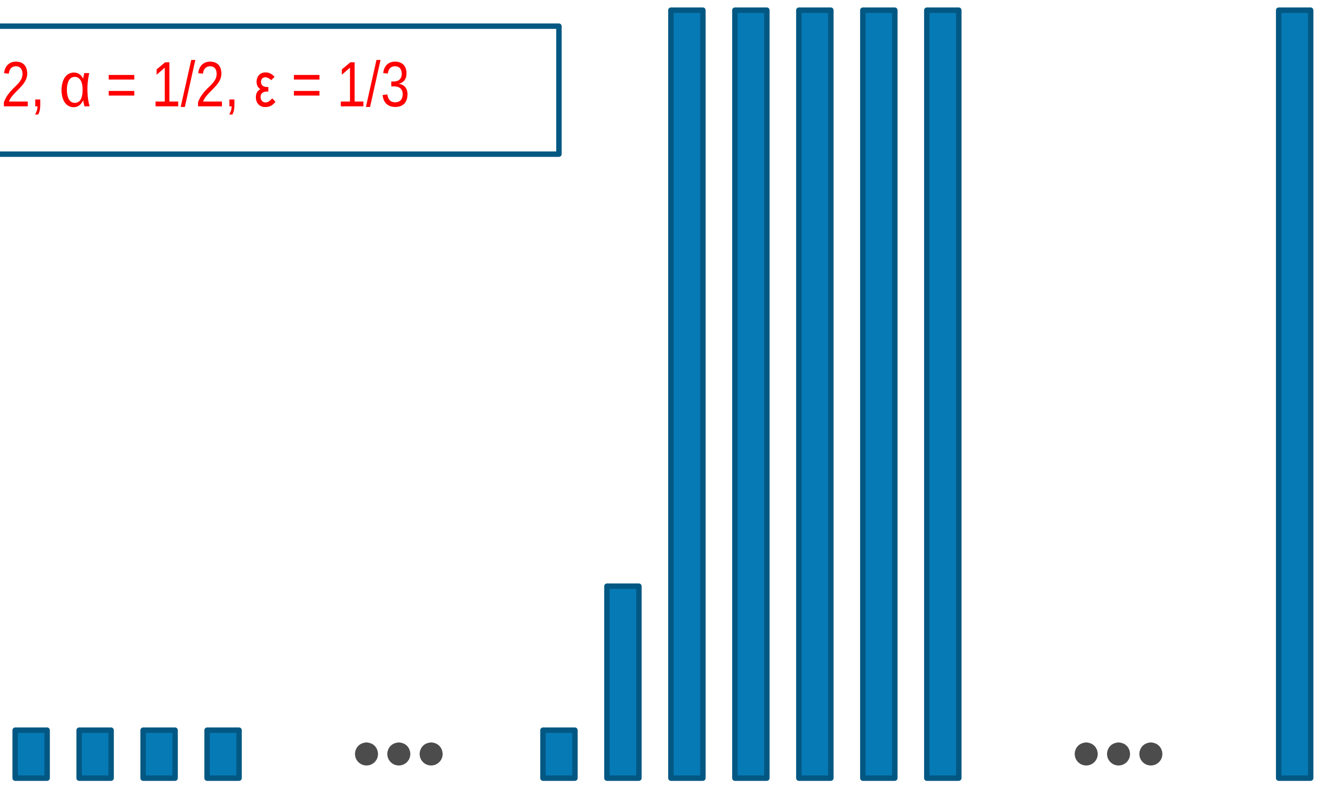
Proof by Picture (not to scale):

B = 2, α = 1/2, ε = 1/3



Proof by Picture (not to scale):

$B = 2$, $\alpha = 1/2$, $\varepsilon = 1/3$



For fixed B and α, $\varepsilon = O(1/n)$

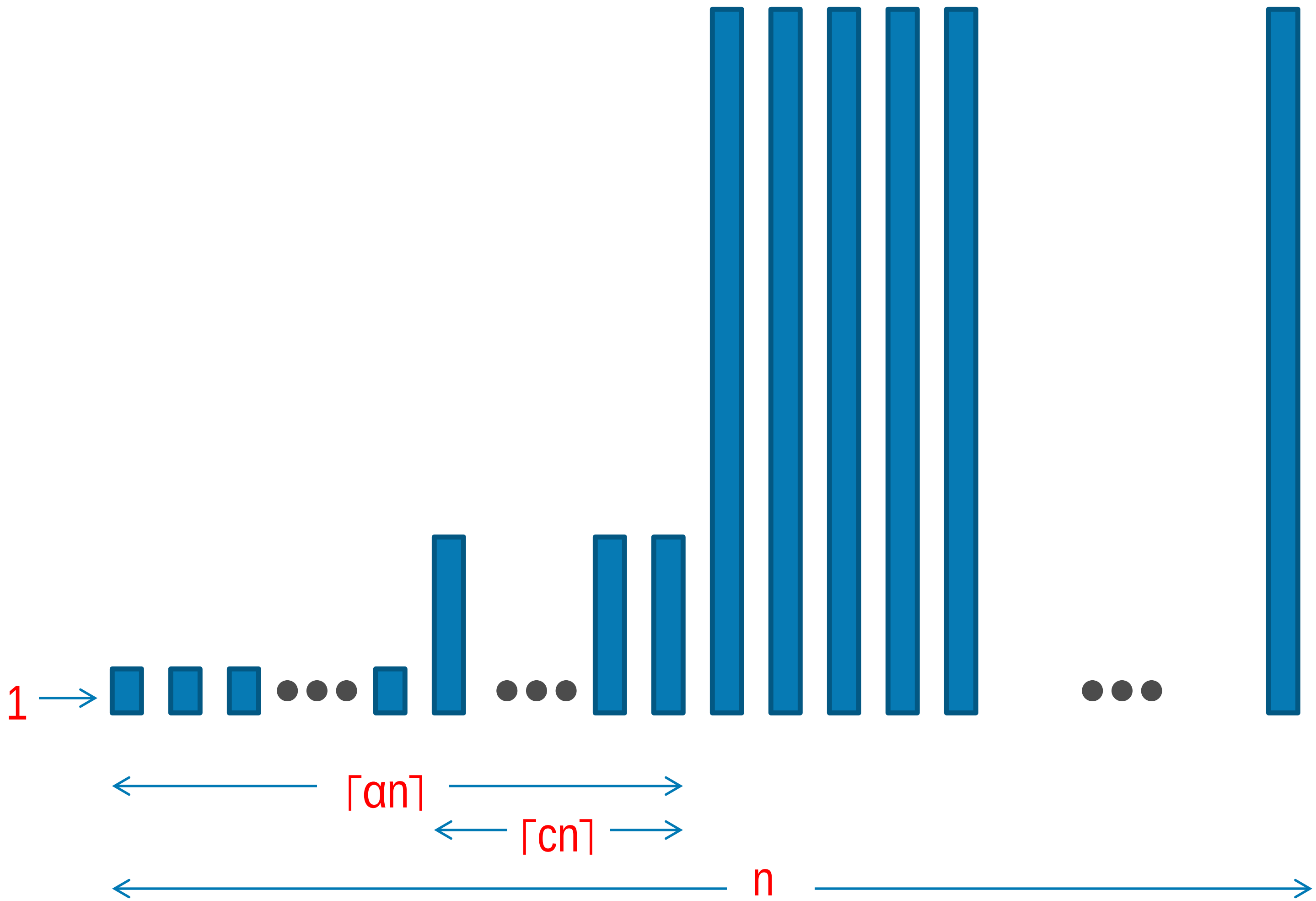
Even Better News (But Worse News for the Adv.)

Theorem: For any constants α, $0 < \alpha < 1$, and $B > 1$, there exist instances s_n, $n > 8B/(1-\alpha)$, such that for any profile-only adversary A

$$\Pr\left[A_\alpha(s_n) < B \cdot Opt_\alpha(s_n)\right] < \varepsilon$$

[ε: negligible]

Proof by Picture (not to scale):



Rest of the Talk

- Framework for realization of above abstraction
  - Computational corruptions
- Sufficient conditions for abstraction
  - *Information-Effort-Preserving* (IEP) functions
  - *Hardness Indistinguishability*
  - *Exact Hardness*
- Restricted instances, efficiency gains, and more

Exact Hardness

- A notion to compare functions according to their inversion difficulty
  - I.e., compute x given $y = f(x)$
- The *exact hardness* of a function, parameterized by ε, is the number of steps that needs to be surpassed in order to achieve prob. of success at least ε
- **Definition:** For any $\varepsilon \in (0,1)$ and a function $f : X \rightarrow Y$, the *exact hardness* of f w. prob. ε is the maximum $H \in \mathbb{N}$ s.t. for any A and $t \leq H$, it holds that

$$p_{A,t} < \varepsilon$$

[Denoted $H_{f,\varepsilon}(\lambda)$]

Exact Hardness (cont'd)

- Related notions:
  - Boolean functions [NW94], (t,ε)-security [BR96]
  - One-way functions
  - One-wayness with hardness μ [HHR06]
- How easy is it to calculate $H_{f,\varepsilon}$?
  - Idealized computational models (random functions, exponentiation maps in the generic group model)
  - Under cryptographic assumptions (e.g., factoring), reasonable ranges for $H_{f,\varepsilon}$ can be stated

Inversion-Effort-Preserving (IEP) Functions

- A *set* of functions are to be inverted
- IEP: Measure of "combined" hardness
- **Definition:** Let $\varepsilon > 0$ and τ be a monotonically increasing function. A sequence of functions $\{ f_i \}$ is τ-inversion effort preserving (τ-IEP) if

$$H_{f^{[n]},\varepsilon} \geq \tau\Big(\Sigma_i \, H_{f_i,\varepsilon} \Big)$$

- **Related notions:** Hardness amplification [Yao86], direct-product theorems [IJKW10]

Hardness Indistinguishability

- Hides the function's hardness, "blinding" the adversary as to what function(s) to attack first
- **Definition:** Let $\varepsilon > 0$ and $t \in \mathbb{N}$. Two functions $f_1 : X_1 \rightarrow Y_1$ and $f_2 : X_2 \rightarrow Y_2$ are (t,ε)-indistinguishable if

$$\left|\Pr[D_t(f_1(x_1)) = 1] - \Pr[D_t(f_2(x_2)) = 1]\right| < \varepsilon$$

D_t: statistical test runing in t steps; x_i uniformly drawn from X_i

- "Interesting" when, say, $H_{f_1,\varepsilon} < H_{f_2,\varepsilon}$ for some ε

Candidate Functions

- Random functions
  - "Random oracles" [BR93]
- Exponentiation
  - $f : \mathbb{Z}_q \rightarrow S$; q: λ-bit prime number; S: (generic) multiplicative group
  - $\tau(\cdot) = (\cdot)^{1/2}$
- Multiplication
  - $f_{mult} : P_\lambda \times P_\lambda \rightarrow \mathbb{N}$
  - $\tau(x) = e^{(\ln x)^{2/3}}$

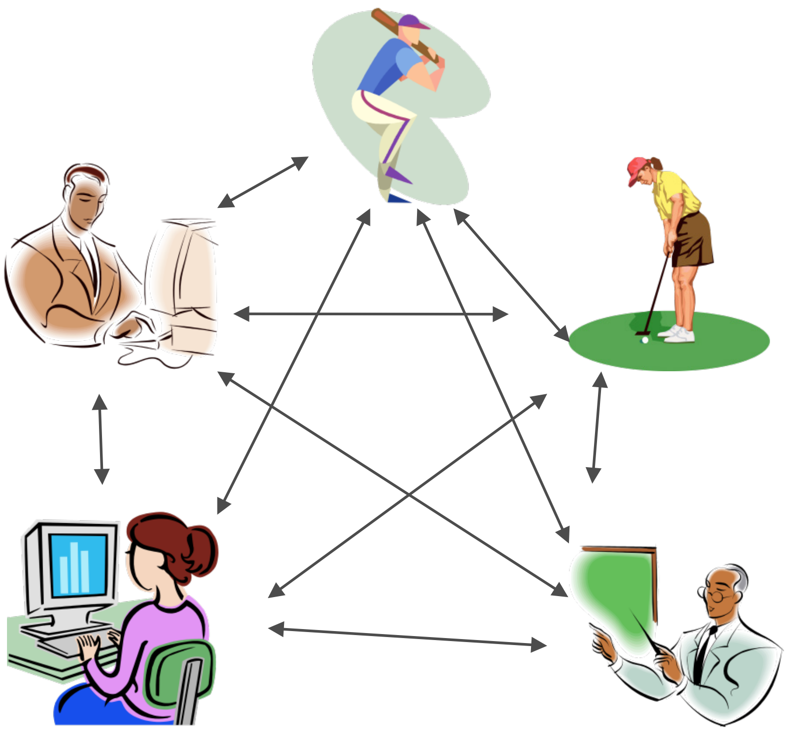
Rest of the Talk

- Framework for realization of above abstraction
  - Computational corruptions
- Sufficient conditions for abstraction
  - *Information-Effort-Preserving* (IEP) functions
  - *Harness Indistinguishability*
  - *Exact Hardness*
- Restricted instances, efficiency gains, and more

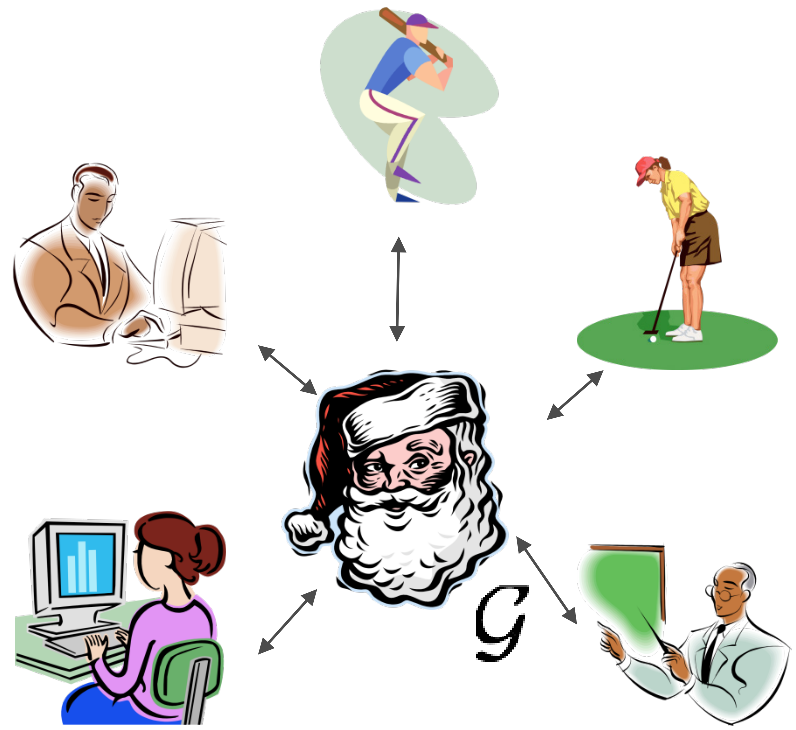
Rest of the Talk

- Framework for realization of above abstraction
  - Computational corruptions
- Sufficient conditions for abstraction
  - *Information-Effort-Preserving* (IEP) functions
  - *Harness Indistinguishability*
  - *Exact Hardness*
- Restricted instances, efficiency gains, and more

The Simulation Paradigm [GMW87,Can01-05]



Real-world cryptographic protocol π



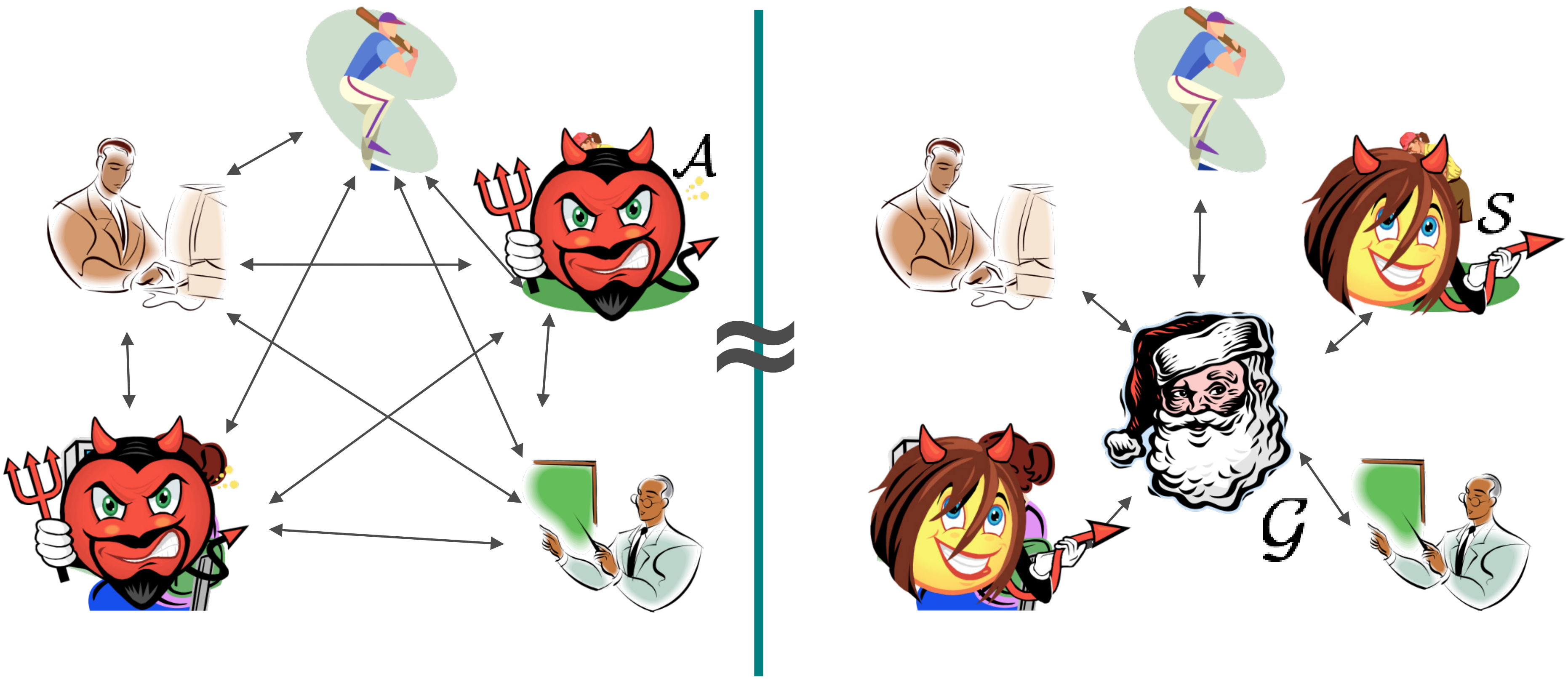
Ideal world with a Trusted Party carrying out task $\mathcal{G}$ in a secure way

The Simulation Paradigm [GMW87,Can01-05]

A protocol is secure for some task if it “emulates” an “ideal process” where the parties hand their inputs to a “trusted party,” who locally computes the desired outputs and hands them back to the parties.

(Aka the “trusted-party paradigm”)

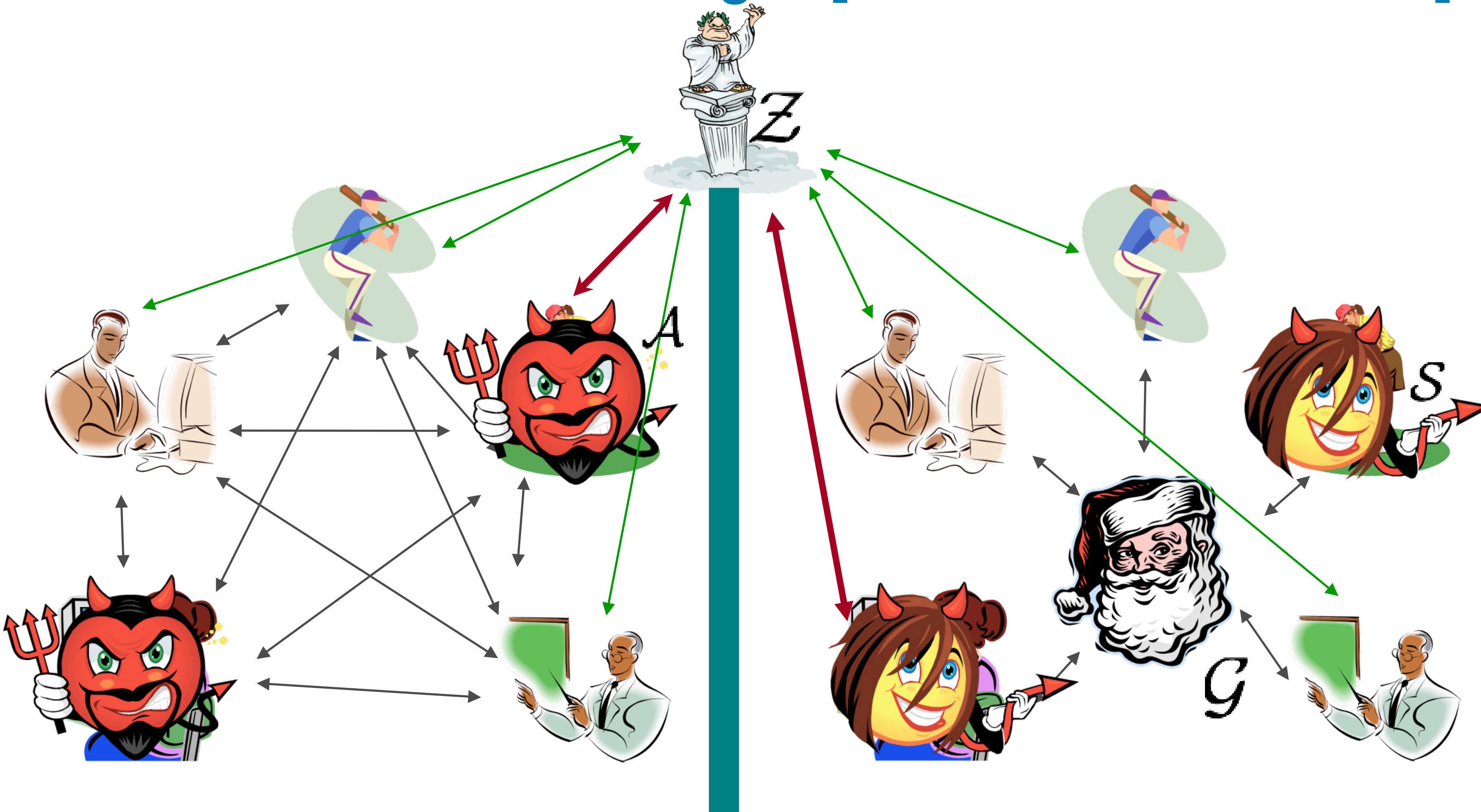
The Simulation Paradigm [GMW87,Can01-05]



REAL

IDEAL

The Simulation Paradigm [GMW87, Canetti 01-05]



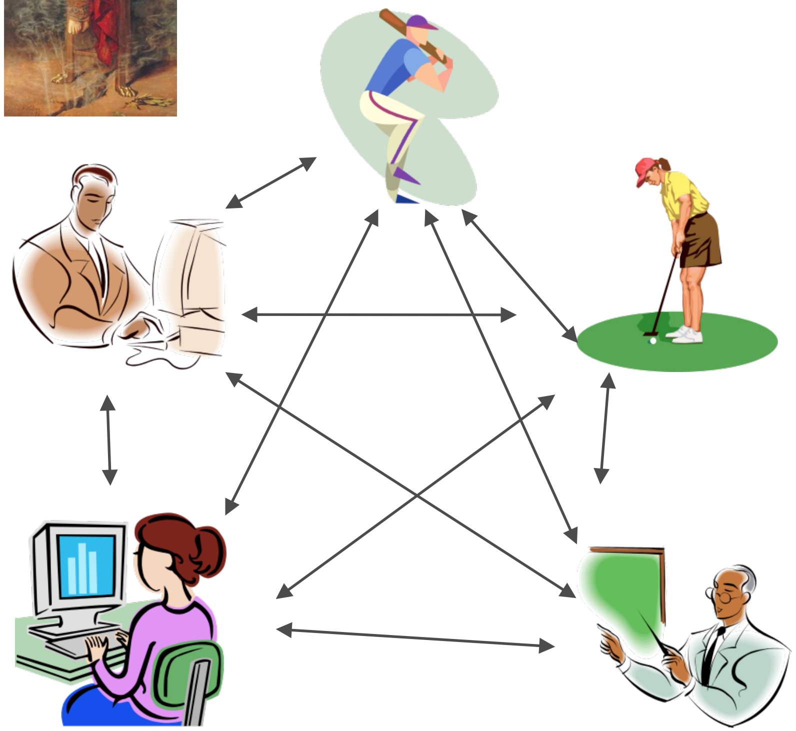
REAL

IDEAL

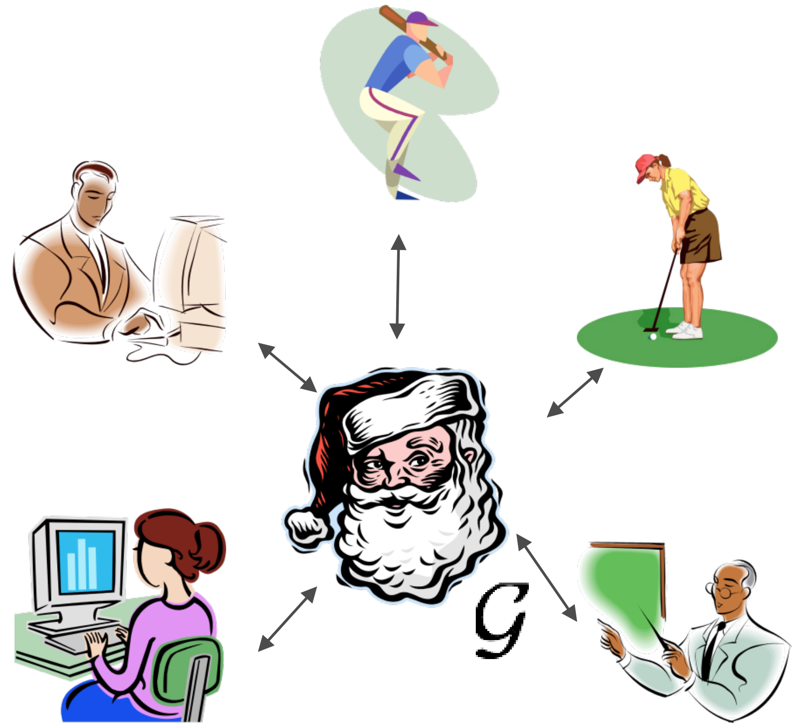
Definition:
We say protocol π UC realizes task $\mathcal{G}$, if $\forall \mathcal{A} \exists \mathcal{S} \forall \mathcal{Z}$ such that $\mathsf{REAL}_{\pi,\mathcal{A},\mathcal{Z}} \approx \mathsf{IDEAL}_{\mathcal{G},\mathcal{S},\mathcal{Z}}$

Corruption Oracles

$\mathcal{C}$

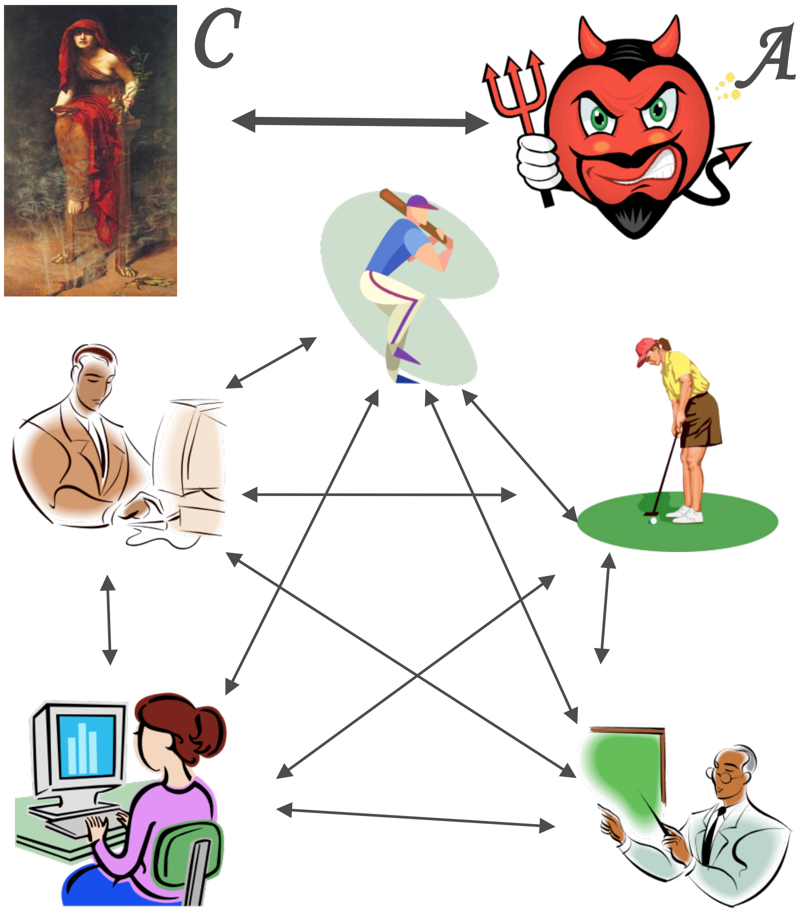


Real-world cryptographic protocol π



Ideal world with a Trusted Party carrying out task $\mathcal{G}$ in a secure way

Corruption Oracles



Real-world cryptographic protocol π

Ideal world with a Trusted Party carrying out task $\mathcal{G}$ in a secure way

Corruption Oracles (cont'd)

- Standard cryptographic corruption: $\mathcal{C}^{std}(\alpha)$
  - Corruption protocol: (**Corrupt**,P_i); oracle checks whether $ctr+1 < \lceil \alpha n \rceil$
- (Blinded) Token-based corruption: $\mathcal{C}^{(b)tk}(\mathbf{s},k)$
  - Counters $ctr_1,\ldots,ctr_n$; (**Corrupt**,P_i,v); oracle checks whether $ctr_i + v \geq s_i$
  - Blinded: Oracle performs update operations on $P_{\pi(i)}$
- (Blinded) Computational corruption: $\mathcal{C}^{(b)cc}(\mathbf{f})$
  - Oracle initialized with $f_1,\ldots,f_n$; gives adversary $(y_i = f_i(x_i))_{1,\ldots,n}$
  - (**Corrupt**,P_i,x); if y_i, $f_i(x)$ then P_i gets corrupted
  - Blinded: Oracle gives adversary $(y_{\pi(1)},\ldots, y_{\pi(n)})$

Relations between Corruption Oracles

- **Definition:** A corruption oracle $\mathcal{C}$ is *safe* if for all functionalities $\mathcal{F}$ there is a protocol π that securely $\mathcal{F}$ with respect to $\mathcal{C}$
  - E.g., $\mathcal{C}^{std}(½)$ is safe
- **Definition:** Oracle $\mathcal{C}_2$ *dominates* oracle $\mathcal{C}_1$ (denoted $\mathcal{C}_1 \leq_{t,\varepsilon} \mathcal{C}_2$) if for any protocol π there is an adversary $\mathcal{S}$ such that for all t-bounded $(\mathcal{Z},\mathcal{A})$

$$\mathrm{EXEC}_{\pi,\mathcal{A}^{\mathcal{C}_1},\mathcal{Z}} \approx_{\varepsilon} \mathrm{EXEC}_{\pi,\mathcal{S}^{\mathcal{C}_2},\mathcal{Z}}$$

Relations between Corruption Oracles (cont'd)

- **Theorem:** Let $\varepsilon > 0$. Given a τ-IEP sequence of functions $f_1,\ldots,f_n$ we have that for any t there exist $\mathbf{s}$, k such that

$$\mathcal{C}^{(b)cc}(\mathbf{f}) \leq_{t,\varepsilon} \mathcal{C}^{(b)tk}(\mathbf{s},k)$$

where $\mathbf{s} = (s_1,\ldots,s_n)$ and $s_i = H_{f_i,\varepsilon}$, and $k = \lceil \tau^{-1}(t) \rceil$.

Rest of the Talk

- Framework for realization of above abstraction
  - Computational corruptions
- Sufficient conditions for abstraction
  - *Information-Effort-Preserving* (IEP) functions
  - *Hardness Indistinguishability*
  - *Exact Hardness*
- Restricted instances, efficiency gains, and more

Rest of the Talk

- Framework for realization of above abstraction
  - Computational corruptions
- Sufficient conditions for abstraction
  - *Information-Effort-Preserving* (IEP) functions
  - *Hardness Indistinguishability*
  - *Exact Hardness*
- Restricted instances, efficiency gains, and more

Results

Increased **security**:

- Let OPT be optimal corruption budget for which the completeness of MPC is violated
- For any B, the completeness of MPC holds against any adversary with less than B·OPT budget assuming a sufficient number of parties ($n = \Omega(\log(1/\varepsilon)\cdot B)$)
- Let M bound the hardness of individual corruptions. Then the completeness of MPC holds against any adversary with less than $\sim \sqrt{M}\cdot OPT/(\log(1/\varepsilon)$, assuming $n \geq \sqrt{M}$

Increased **efficiency**: Fix adversary budget $k < OPT_{½}(s)$

- With resource anonymity, can force corruption threshold to drop from 1/2 to 1/3, and run *information-theoretic* MPC protocol instead!

Summary

- Formulated natural notion of *resource-based corruptions*, which imposes a cost to the adversary to take over parties
- Introduced notion of *hidden diversity* ("resource anonymity"), based on
  - *Exact hardness* of functions
  - *Information-Effort-Preserving* (IEP) functions
  - *Hardness Indistinguishability*
- Showed that the gain of hidden diversity/resource anonymity can be substantial (*unbounded* in some cases)
- Reference:

  J. Garay, D. Johnson, A. Kiayias, and M. Yung, `"Resource-based Corruptions and the Combinatorics of Anonymity.'' 2011; submitted for publication.

Thanks!