Taking the Trust out of Global-Scale Web Services

Nikolaos Michalakis Robert Soulé Robert Grimm

New York University

Less trust for more servers

- Goal: build global-scale web services
  - massive replication (both content and code)
  - decentralized management
- Problem: Don't trust replicas to execute correctly
  - Just signing the content is ineffective
- Our approach: *Repeat and Compare*
  - Repeat computations at other replicas (verifiers)
  - Compare the results to detect misbehavior

Repeat and Compare

- Replicas send responses to clients
- Clients forward a *fraction* to *random* verifiers
- Verifiers repeat, compare, and publish results
- Analogous to voting/reputation systems
  - Focus on *computations* instead of data
- Trade-off BFT-type dependability for scalability
  - Flux of hosted applications
  - Agreement is infeasible on a global-scale

Repeat: remove randomness, add explicitness

- Challenges
  - Non-deterministic computations
  - External inputs
  - Implicit configuration parameters
- Exploit the constraints of web-based architectures
  - Stylized and bounded functionality
- Make inputs and configuration parameters explicit
- Open problem: databases

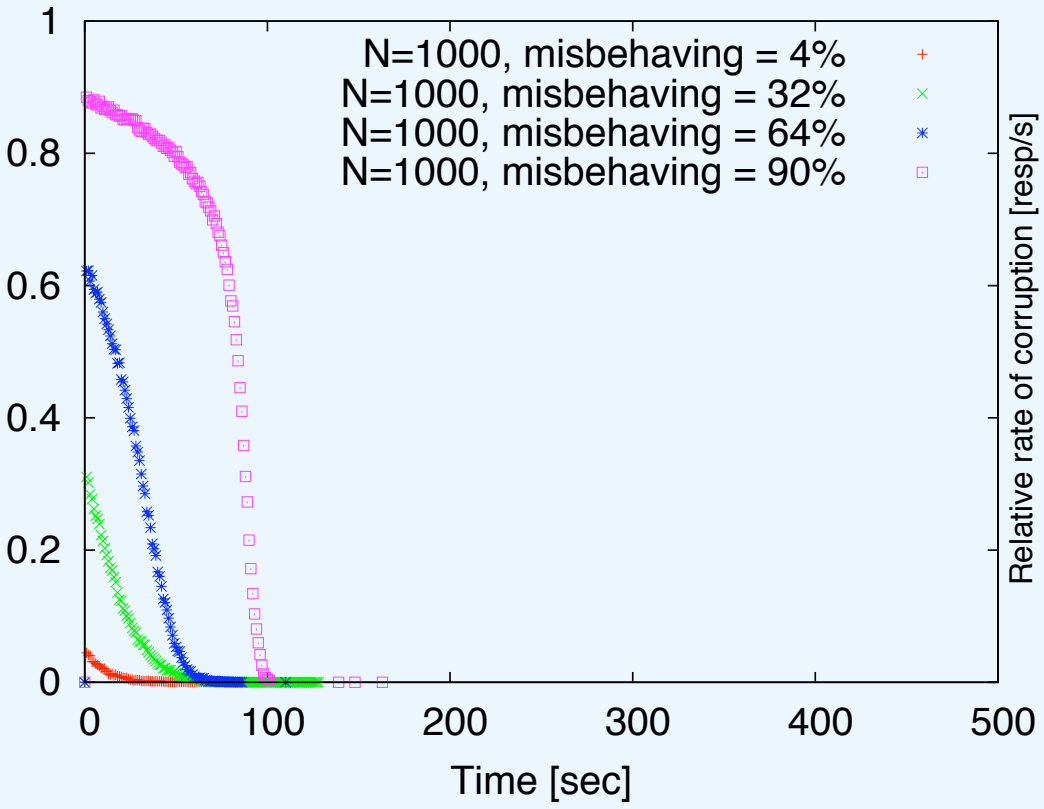
Compare: enforce accountability

- Challenge: "he said, she said" conflicts
  - Replicas can lie (about what they send)
  - Clients can lie (about what they receive)
  - Verifiers can lie (about who is misbehaving)
- Enforce accountability through *attestation records*
  - Cryptographically bind nodes to their statements
    - Including inputs, code, and outputs

Collect and distribute results

- Repeat and Compare is effective at isolating misbehaving nodes.

Trusted Core



Decentralized