

BFT for the skeptics

Yee Jiun Song, Flavio Junqueira, Benjamin Reed  
Cornell University, Yahoo! Research

# BFT What is it good for?



Clients



Replicas

- Industry has been handling crash failures more often

# BFT What is it good for?



Clients



Replicas

- Industry has been handling crash failures more often
- Can we do more?
  - Corruptions
  - Partial failures
  - Bugs

# BFT What is it good for?



Clients



Replicas

- Industry has been handling crash failures more often
- Can we do more?
  - Corruptions
  - Partial failures
  - Bugs
  - Malicious actors

# But wait, do we need it?

- We already use checksums to detect corruption and translate to crash
- Timeout and sanity checks allows us to catch a range of non-malicious byzantine faults
- We still have the software bugs and malicious attacks
  - Of course byzantine only helps if these bugs are independent, otherwise we exceed the failure threshold
  - We get to implement the system once
- So, how often do we get faults that could be handled by BFT?

# Real world ZooKeeper failures

- ZooKeeper is a replicated mission critical coordination service
- For over a year and a half Yahoo!'s crawler has used ZooKeeper

# The bugs

- Misconfiguration: 5 issues
  - System configuration and ZK configuration
  - e.g. network device config, DNS name clash
- Application bugs: 2 issues
  - Misunderstanding of the API semantics
  - e.g. race condition using async API
- ZooKeeper bugs: 2 issues
  - Our fault, affected all replicas
  - e.g. bug on committing commands

# Could it hurt?

- Misconfigurations is the category with the most faults, BFT has more things to configure if things such as keys are used

# Summary

- This is just one data point meant to motivate a question not answer it.
- Until we show that BFT really solves a problem, industry is not going to pick it up.
- Can I build it? (yes) Does it solve my problem? (???) Can I run it? (???)