R2P2: Making RPCs first-class datacenter citizens

Marios Kogias  George Prekas  Adrien Ghosn  Jonas Fietz
Edouard Bugnion

EPFL
Datacenter Communication

- Data-stores, search, etc...
- Complex Fan-in/Fan-out patterns
- μs-scale Remote Procedure Calls
- Tight tail-latency SLOs
RPC Multiplexing

• Use TCP (byte-stream-oriented) to send RPCs (message-oriented)
  • gRPC, Thrift, etc...

• End-host problems:
  • Head-of-line blocking
  • Scheduling
  • Buffer-bloat
  • Application-level RPC parsing

• Infrastructure-wide problems:
  • Load-balancing
    • Software-based L7 loadbalancing
  • Inability to leverage network programmability

Let’s make RPCs first class datacenter citizens
R2P2 – Request Response Pair Protocol

- Expose the RPC abstraction to the network and the end-hosts
- Independent request-response pairs
- Request/Response - aware middleboxes
- Decouple RPC target selection from request and reply streaming
- Use programmable switches for in-network RPC scheduling
  - JBSQ policy
R2P2: Making RPCs first-class datacenter citizens

Marios Kogias  George Prekas  Adrien Ghosn  Jonas Fietz
Edouard Bugnion

EPFL