An Improved Multicast Protocol for Paranoid Stateful Lambdas

Marcus Plutowski, Vivek Bharadwaj, Willis Wang

Problem Statement

- Paranoid stateful lambdas (PSL) provide a function-as-a-service utility that allows execution of trusted code with confidential data on potentially untrusted edge devices with eventually-consistent semantics
- Code executed within **secure enclaves**, which prevent compromised / malicious actors from reading privileged information
- As edge devices communicate with each other, messages must be protected between enclaves. PSL uses DataCapsules for this purpose, taking advantage of their embedded hash chains to track history.
- **Goal:** to improve efficiency of all-to-all communication involved in the PSL key-value store by improving the multicast protocol

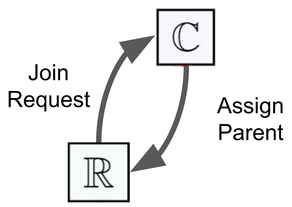
Metrics for Success

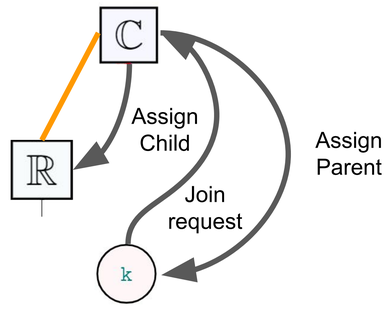
+ **Minimize # of enclave messages:**
  - Costly to decrypt/analyze on the edge: computation delays >> network delays
+ **Reduce network congestion:**
  - Linear growth of network load in relation to multicast tree size.
  - Can't overload bottleneck nodes

− **Avoid major latency penalties:**
  - Single-message latency cost should not be more than O(n) in # of subscribers
− **Maintain security predicates:**
  - Maintain data security
  - Non-credentialed actors should not be able to disrupt data propagation within the multicast tree

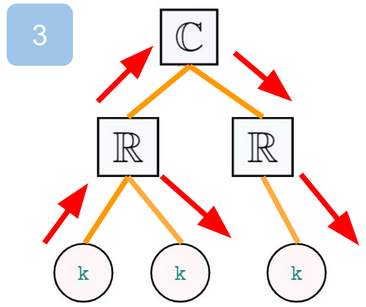
Proposed Multicast Protocol

1 Coordinator sets up root router and listens for join requests





Routers, clients ask coordinator to join multicast tree. Coordinator responds with parent ID, informs parents of new children



Multicast messages propagated up and through the tree via routers

Types of Network Nodes

Routers: intermediary nodes used to route data in flight to other routers and clients

Coordinator: root router delegating multicast tree structure. Holds a global view of the tree

Clients: worker node with onboard enclave, working set of shared resources (e.g. distributed k-v store)

Protocol Characteristics

Assume n clients, where n potentially >> 1000

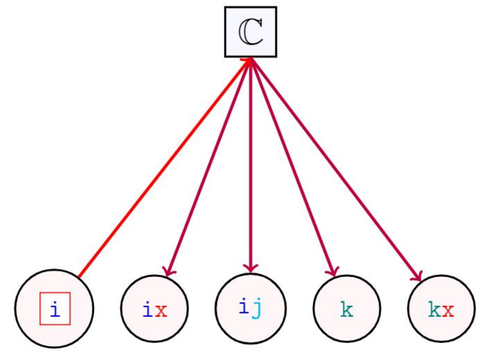
Original Multicast Tree: Coordinator must send O(n) messages per multicast.

K-Ary Tree Protocol: Each node sends / receives at most K + 1 messages per multicast

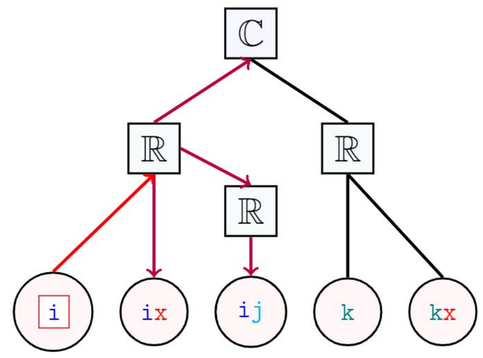
FAT-Tree Assumption: Nodes closer to the root have higher bandwidth than leaves.

Aliveness Signal *(In Progress)*: Absence of heartbeat signal prompts coordinator to reorganize tree.

Current Protocol



Sharded Multicast Tree



Sharded M-Cast Tree w/ Domains