Fault Isolation in a Multicast Tree using DYSWIS

COMS 6181 Fall 2011

Phil Sphicas

Goal

- Correlate faults between multicast receivers of the same stream, and pinpoint where in the network the loss occurred

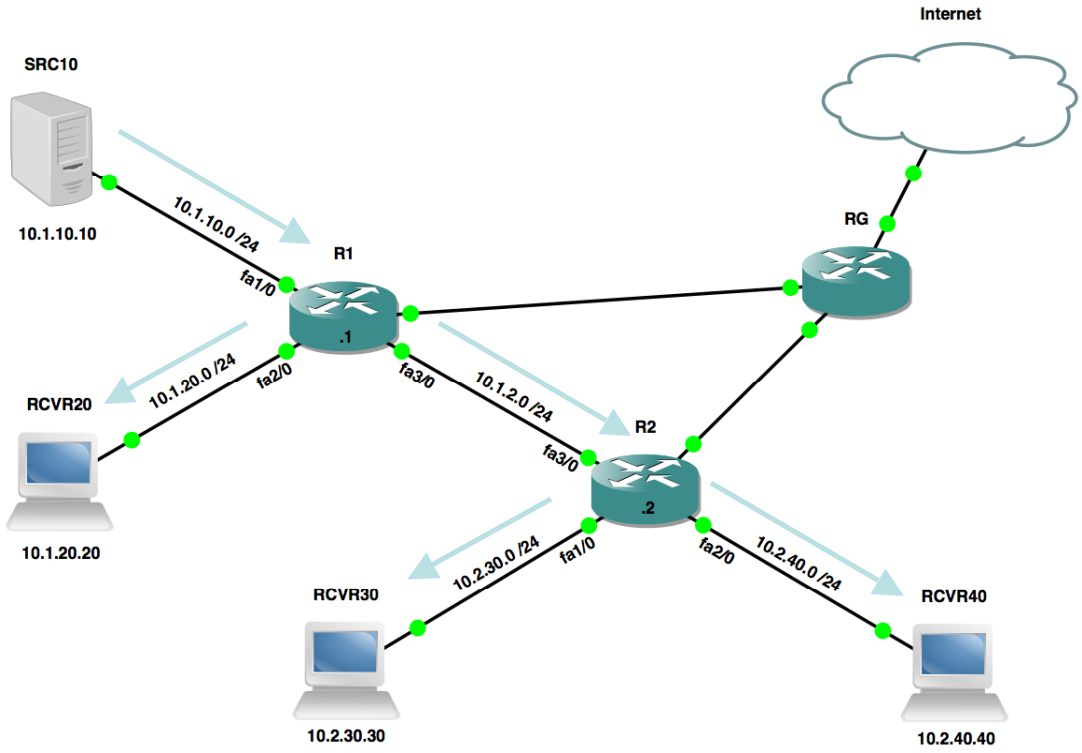
How do we detect faults?

- RTP has sequence numbers
- When a packet is received with an unexpected sequence number, we know that a fault occurred
- This could be a loss or a packet reordering - for the purposes of this project, we don’t distinguish

How can we isolate faults?

- Nodes can determine their own path to the multicast source
- If an end node experienced a fault, all the hops between it and the source are suspect
- If an end node was joined to the stream but did not experience the fault, all the hops between it and the source are good
- By combining the sets of known good and possibly bad hops, we can come up with a smaller set of suspect hops

Multicast Topology



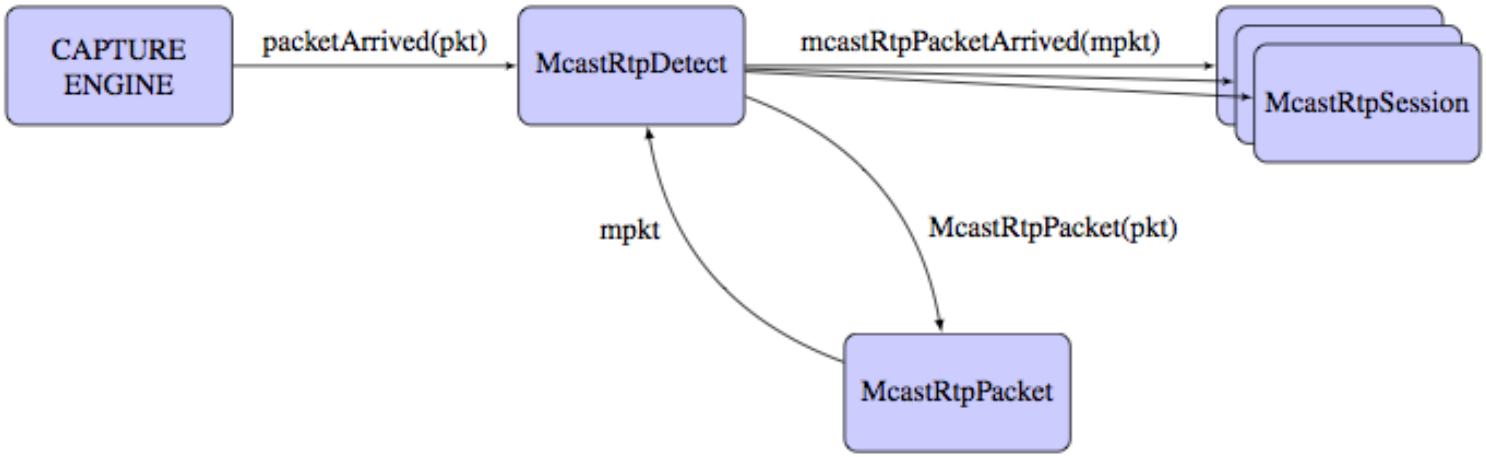
Fault Isolation Algorithm

- Let *H(n)* be the set of hops between node n and the source
- Choose a node *a* that experienced a fault
- Let *B* represent the set of possible bad hops
- Initialize $B = H(a)$
- For all other nodes b that experienced the fault, $B = B \cap H(b)$
- For all nodes c that did not experience the fault, $B = B \setminus H(c)$

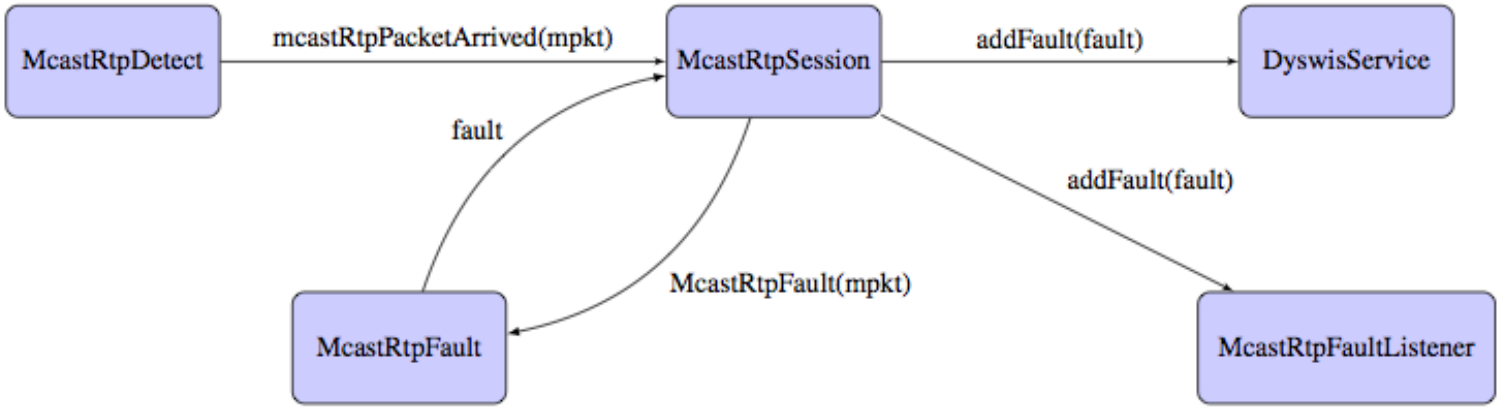
Shifting gears … DYSWIS

- DYSWIS is a distributed automatic fault detection and diagnosis system
- Provides a framework for detecting faults, querying other nodes for information about faults, and analyzing the results

Monitoring multicast RTP streams



Detecting multicast RTP faults



DYSWIS Probes

- Probes are used to query remote DYSWIS nodes for information
- We use a MultiProbe to query multiple nodes at once
- We ask each remote node:
  - Was it joined to the same stream at the same time?
  - Did it experience the same fault?
  - What is its path back to the source?

Diagnosing multicast RTP faults