Bidirectional Forwarding Detection (BFD)

Matt Kolon

APRICOT2004

KualaLumpur

Juniper NETWORKS

Overview

- Goals
- Protocol Overview
- Operation
- Status
- Implementation Details

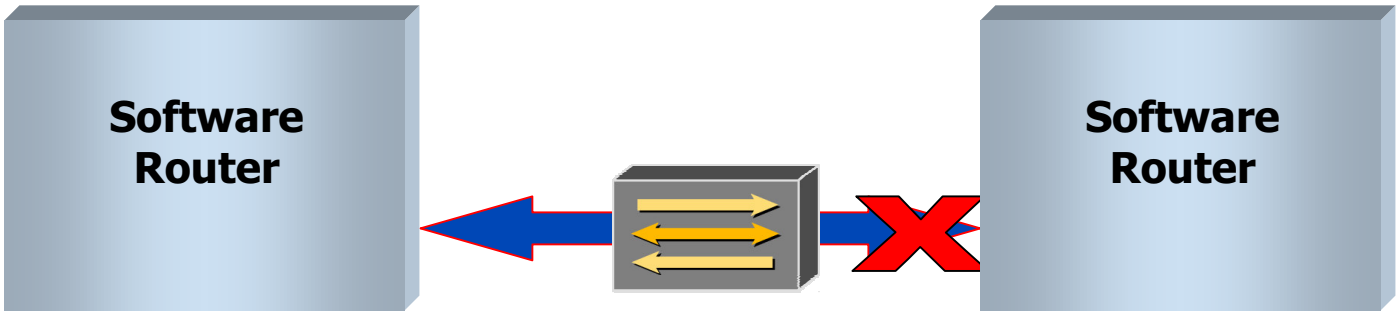
Juniper NETWORKS

Detecting Forwarding Failures (I)

- For IP, classically a function of the routing protocol
  - Because formerly, routing = forwarding
  - Fault resolution in perhaps tens of seconds
  - Perhaps too slow for anything but best-effort IP
  - Sometimes there is no routing protocol!

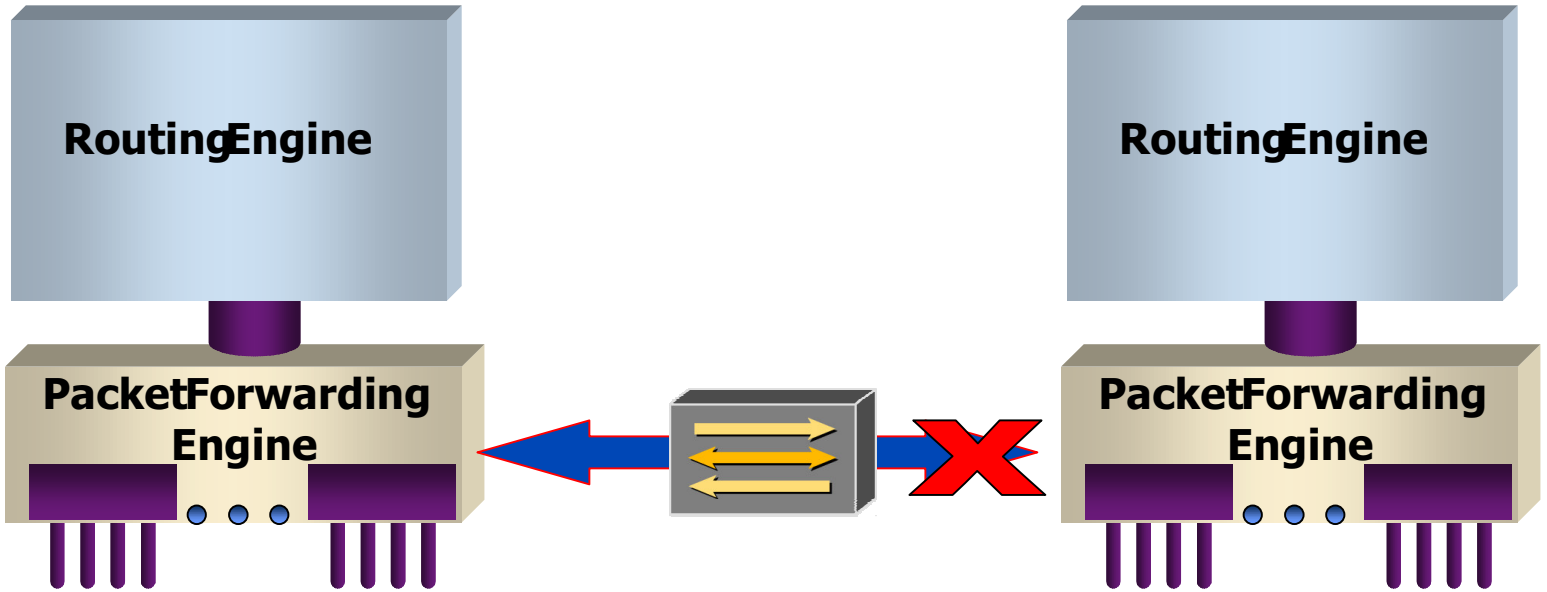
Detecting Forwarding Failures (II)

- Interface state can be used, but:
  - Transient failures are problematic
  - So are switched environments



Detecting Forwarding Failures (III)

- In modern architectures: routing ≠ forwarding
- Would be nice to have a way to detect forwarding without relying on interface state



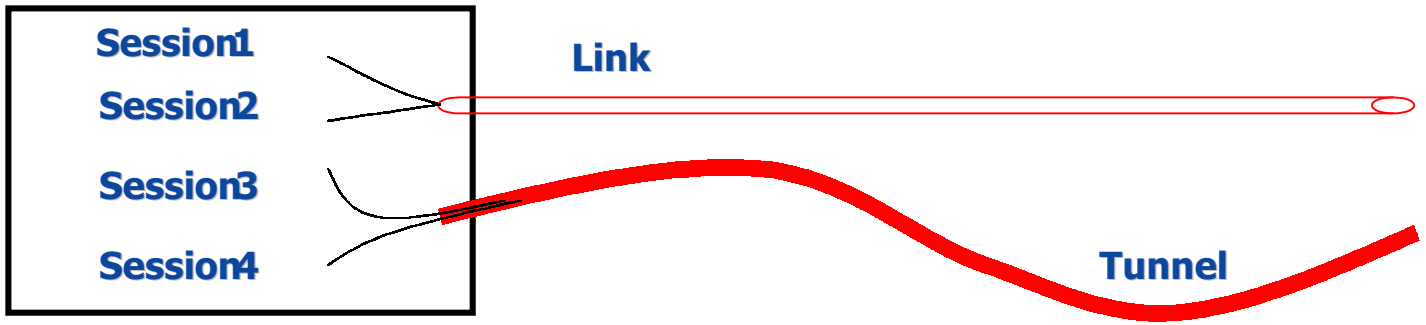
Goals for BFD

- Detection of forwarding plane-to-forwarding plane connectivity (including links, interfaces, etc.)
- Faster convergence of routing protocols, particularly on shared media like Ethernet
- Semantic separation of forwarding plane connectivity vs. control plane connectivity (works well in concert with Graceful Restart)
- A single mechanism that is independent of media, routing protocol, and data protocol
- No changes to existing protocols

BFD Protocol Overview

- At its heart, Yet Another Hello Protocol
- Packets sent at regular intervals; neighbor failure detected when packets stop showing up
- Intended to be implemented in the forwarding plane to the extent possible (avoiding fate sharing with the control plane)
- Context defined by encapsulating protocol (sending inside IPv4 packets signals IPv4 connectivity; also could be sent over IPv6, directly over the datalink, or whatever)
- Always unicast, even on shared media

BFD Protocol Overview



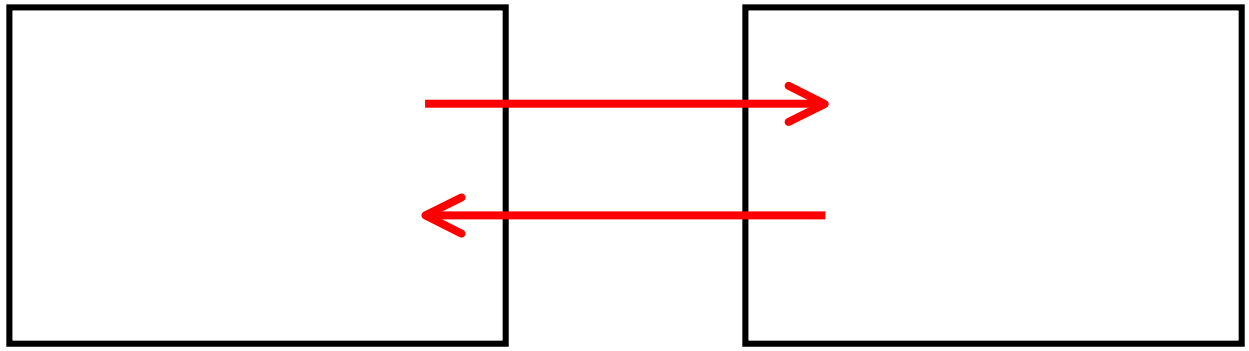
- Not just for direct links; can also be used over MPLS LSPs, other tunnels
- Sufficient context in the protocol to keep track of multiple parallel paths between systems
- Timing is adjustable on-the-fly, allowing for adaptivity to avoid catastrophic collapse due to false failure detection

BFD Operation

- BFD Control Packets sent in both directions, providing basic connectivity check and continuous parameter negotiation
- Optional Echo Mode can be negotiated
  - BFD Echo Packets transmitted addressed to originating system; other system forwards them back through regular forwarding path
  - Exercises entire forwarding path in destination system
  - May not always be possible or desirable, thus it is negotiated

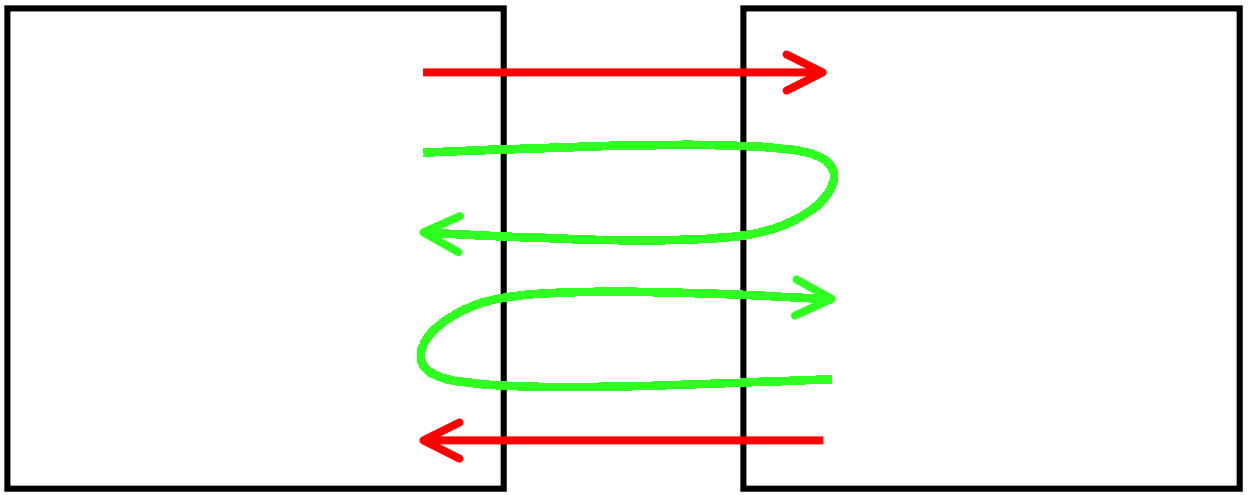
BFD Async Mode

- Control packets flow in each direction



BFD Echo Mode

- Control packets flow in each direction
- Echo packets loop through remote system
- Control packet flow is slower



BFD Operation

- Combination of continuous negotiation and unicast allows high adaptivity
  - Less capable systems can negotiate slower detection without penalizing faster systems
  - Transmission rate and detection time can be changed on-the-fly, allowing for automatic mechanisms for recovery from overly aggressive timers or unusual network events
  - Echo mode can be enabled or disabled without taking down BFD session
- Three-way handshake on both up and down transitions
- Each system in full control of its own fate

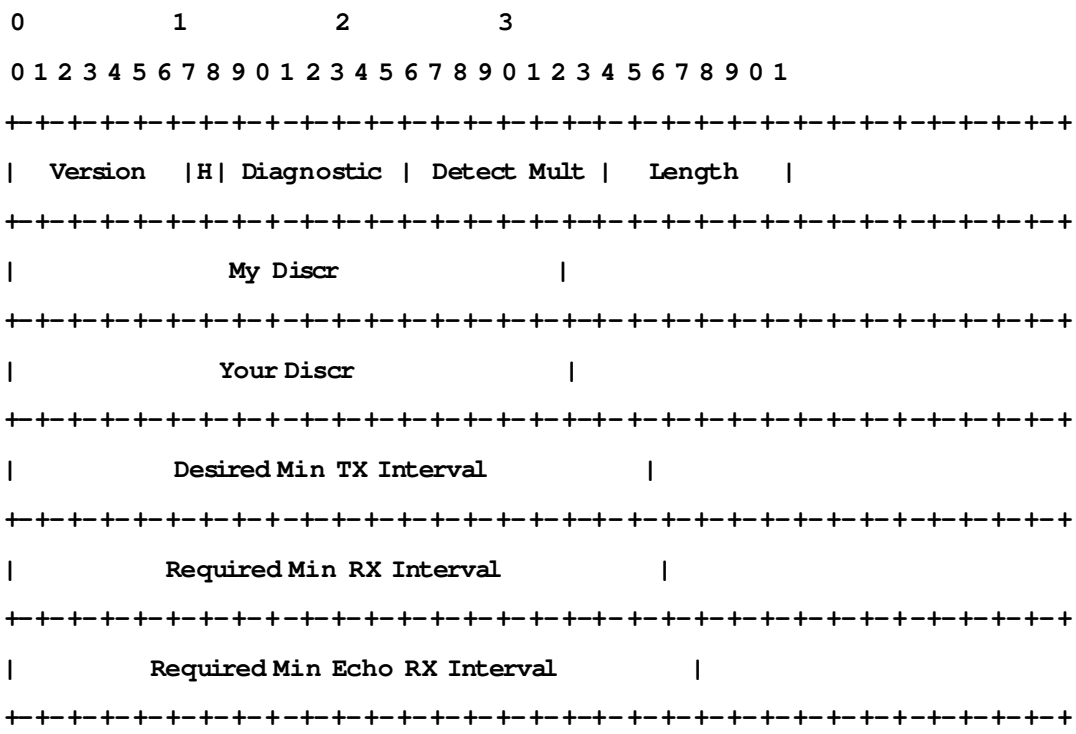
BFD Operation

- No discovery protocol, handled by control protocol
- Semantics of BFD session failure are contextual
  - BFD-over-IP implies neighbor failure; IGP neighbor should be torn down
  - BFD-over-Ethernet implies switch failure; subnet should be withdrawn from routing protocols
- Works over a wide range of time constants
  - Timers specified in microseconds, allowing very fast or very slow detection
  - Systems specify how quickly they can receive BFD packets so that boxes of differing abilities can interoperate

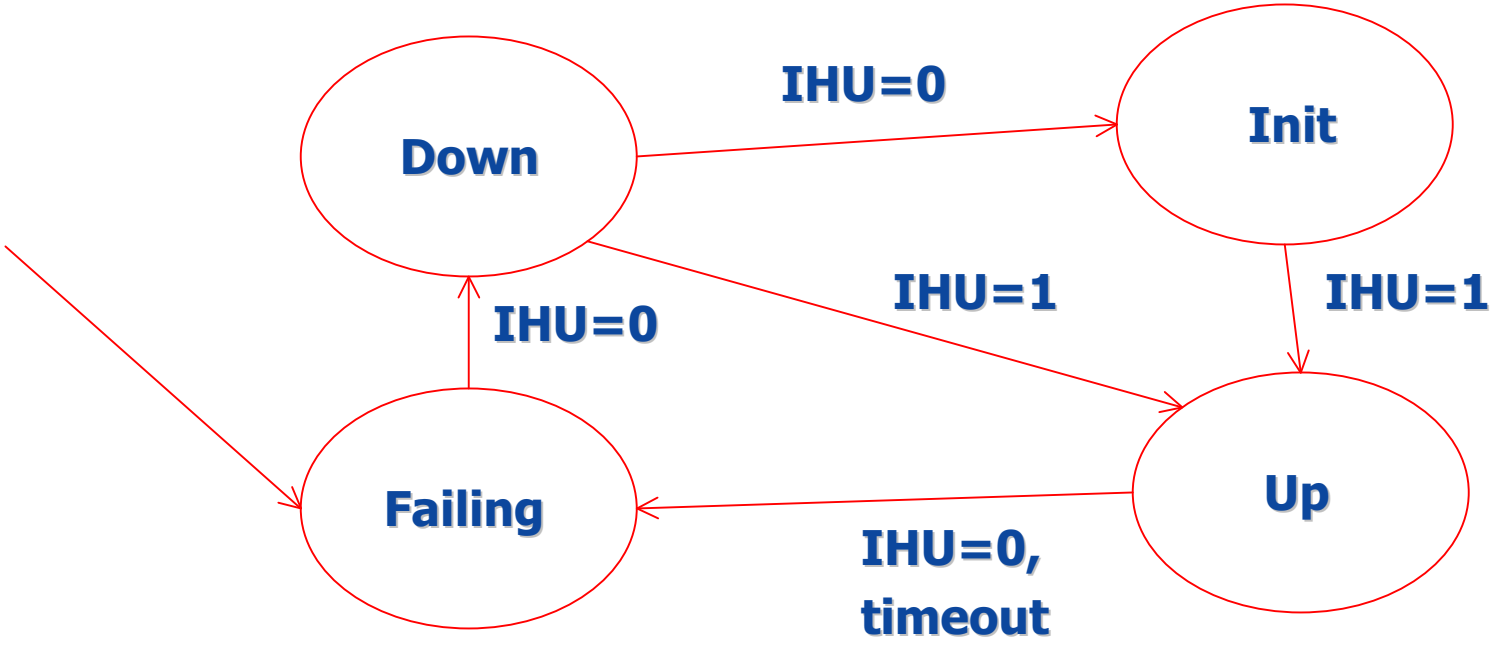
BFD Operation

- Discriminator field used to demultiplex incoming packets once session is up
  - Each system chooses 32 bit value that is unique among all sessions, puts it in "My Discr" field
  - Remote system's discriminator is looped back in "Your Discr" field
  - "Your Discr" used to choose session for incoming packets
- Initialization is slightly tricky
  - First packet has to be demuxed on other info—source address and interface index, for example

BFD Control Message



BFD State Machine



BFD Applications

- Routing protocol liveness detection (obviously)
- Switch-to-router liveness detection
- Router-to-host liveness detection (a real answer for the web farms)
- Virtual link liveness (MPLS, GRE tunnels, etc.—no need to keep reinventing these)
- Alarm passthrough (receive alarm from concatenated link; take session admin-down with diagnostic "concatenated link down")

BFD Status

- Protocol jointly developed by Juniper and Cisco
- Internet draft published (draft-katz-ward-bfd-0x.txt)
- Published as an Informational document (so that progress can be made in a reasonable amount of time
- Rumors of "fast track" standardization being pursued

Thank You!

http://www.juniper.net