Building Highly Available Networks: Features, Architectures, and Directions

TutoJRES #9 - 2009/02/24
Haute disponibilité des réseaux
Agenda

- Key features on hardware and software
- High availability on Routing solutions
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
Agenda

- Key features on hardware and software
  - High availability on Routing solutions
  - High availability on Switching solutions
  - High availability on Security solutions
  - Conclusion
High Availability Solution Architecture

IP Carrier-Class Availability Is a Culture, Not a Single Feature or Product

- Reliable Services
- Reliable Networks
- Reliable Platforms
  - Hardware
  - Software
  - Process
- ... n

Security

Performance
Reliable Hardware

- **Hard Fault Tolerance**
  - Component redundancy
  - Redundant boot devices
  - Shared memory architecture
  - Use of ASICs

- **Soft Fault Tolerance**
  - Extensive internal diagnostics
  - Error detection - protected internal data paths

- **MTTR Reduction**
  - Hot swappable components
  - Field replaceable components

MTTR= Mean Time To Repair
One Modular Software Architecture

Key architecture features and benefits:
- Firewalls control plane
- Separation of Interfaces and Forwarding planes
  - Enhances resiliency and stability
  - Enables redundancy
  - Enables flexible innovation
- In-Service Software Upgrades (ISSU)
- Non-Stop Routing (NSR)
- Graceful Restart (GR)
- Graceful RE Switchover (GRES)
- Scales performance

High-Level Architecture:
- Control Plane / Routing Engine
- Forwarding Plane / Forwarding Engine
- Kernel
- Hardware Abstraction Layer
- Physical Interfaces

Packet Forwarding
Forwarding Services
Reliable Software

- **Hard Fault Tolerance**
  - Redundant Control Plane (Routing Engines)
  - Different software versions

- **Soft Fault Tolerance**
  - Separate control and forwarding
  - Modular processes can be restarted independently
  - Processes protected in own memory space

- **MTTR Reduction**
  - Incremental software upgrades
  - Modularity to speed up testing
One Modular Software Architecture

- **Stand-alone modules**
  - Protected memory for stability
  - Contains faults, enables rapid fault isolation
  - Restart independently
  - Enables flexible innovation

- **Separation of control and forwarding planes**
  - Scales performance
  - Enhances resiliency
  - Enables redundancy
  - Firewalls control plane
One OS: Consistent Feature Implementation

- Single repository of source code
- One implementation of control plane features
  - Streamlines testing, qualification and deployment
- Consistent user experience
  - Single common management interface and tools
- Single release train
  - Releases are a superset of the previous
  - Achieve zero critical regression errors in each release
- Streamlines upgrades and reduces upgrade issues
HA Problem Space

- Replicate configs
- Replicate FIB
- Replicate RIB
- Keep sessions up on RE switchover

- Without dropping keepalives
- Without resetting hardware
- Across different versions

- Replicate configs
- Replicate FIB
- Replicate RIB
- Keep sessions up on RE switchover

- Without dropping keepalives
- Without resetting hardware
- Across different versions
Agenda

- Key features on hardware and software
- **High availability on Routing solutions**
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
Reliable Networks

Protection and Recovery from failures

- Use of standards is the key for interoperability
- Unique root code is the key for consistency
Agenda

- Key features on hardware and software
- High availability on Routing solutions
  - Bidirectional Forwarding Detection (BFD)
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
Bidirectional Forwarding Detection (BFD)

- The bidirectional forwarding detection (BFD) protocol is a simple hello mechanism that detects failures in a network.
- BFD works with a wide variety of network environments and topologies.
- The failure detection timers for BFD have shorter time limits than the standard failure detection mechanisms, providing faster detection.
- These timers are also adaptive and can be adjusted to be more or less aggressive.
Agenda

- Key features on hardware and software
- **High availability on Routing solutions**
  - Graceful RE Switchover (GRES)
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
GRES High Level Overview (aka Non Stop Fwdg)

- State replication for interfaces and forwarding resources.
  - **GRES** (Graceful RE Switchover) in Juniper terminology.

- **No Route Replication**
  - Backup rpd is *not running*

- **RE keepalive process**
  - Automatic failover to standby
  - Configurable keepalive timer
  - Default value set to 2 seconds

- **RE configurations are synchronized**

- **RE failure does not reset PFE**
  - No effect upon traffic forwarding
  - Alarms, SNMP traps on failover

- **What is replicated**
  - Interfaces
  - Routes
  - *Firewall, sampling, monitoring*
  - *Class of service*
  - *PPP, Frame-Relay, ATM, layer-2 state*
  - ARP Cache

- **What is not replicated**
  - *Routing Table*
  - *TCP state*
GRES Goals

- Switchover from primary to backup (standby) RE
  - Planned restarts
    - Usually for software upgrades
  - Unplanned restarts
    - Hardware failure
    - Software failure

- GRACEFUL: Switchover with as little disruption to control plane as possible

- Key building block for NSR and ISSU
Agenda

- Key features on hardware and software
- **High availability on Routing solutions**
  - Graceful Restart (GR)
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
Graceful-Restart High Level Overview

Graceful Restart
1. Routing goes down, forwarding continues.
2. Re-establish Adjacencies & Sessions.
3. Tell Neighbors to not reset Adjacencies & Sessions
   - requires Protocol Extensions
4. Re-learn inputs (advertisements from neighbors).
5. Advertise out all learned information.
Before NSR: Graceful Restart and NSF

If router’s control plane fails, data plane will keep forwarding packets.

Neighbors must know GR procedures to hide failure from all others routers in the network.

When router recovers, neighbors sync up – provided no topology changes occurred!

Other routers are never made aware of failure.
Graceful Restart – Benefits & Limitations

**BENEFITS**

- No disruption to routing and forwarding
  - On the restarting node, the peers or any other routers in the entire network
- No change in the traffic patterns
  - No impact on jitter, latency, packet ordering, route optimality
- By limiting the scope of the nodes that are aware of a control plane restart to only the routing peers of the restarting node we improve control plane:
  - Scalability across the network
  - Computation capacity and adaptability

**LIMITATIONS**

- Neighboring routers must understand GR procedures and messages
  - Inhibits full GR implementation
- GR must stop if topology changes during grace period
- In some cases, cannot distinguish between link failure and control plane failure
- In some cases, routing reconvergence might exceed grace period
  - For example, if there are hundreds of BGP peers
- Protocol interdependencies can slow reconvergence beyond grace period
  - For example, if BGP depends on LDP restart completion, which depends on RSVP restart completion
- Industry acceptance of GR is not widespread
Agenda

- Key features on hardware and software
- High availability on Routing solutions
  - Non-stop Routing (NSR)
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
NSR High Level overview

- **Switchover from primary to secondary RE transparent to neighboring routers**
  - No interruption of protocol peering
  - Therefore 100% availability during switchover

- **Self-contained mechanism**
  - No required protocol extensions
  - No required cooperation from remote peers

- **Replacement for GR protocol extensions**
- **Key enabler for ISSU**

- State replication for interfaces and forwarding resources provided by **GRES**
- State Replication for Protocols provided by the **mirror system**
- Backup rpd is **running**
NSR: Operations

State maintained
In backup RE

Neighbors – and other routers – do not need to know the Hot RE went down

If router’s control plane fails, backup RE goes into service with full state
Agenda

- Key features on hardware and software
- **High availability on Routing solutions**
  - In-service software upgrade (ISSU)
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
ISSU Goals

- **Upgrade complete OS package**
  - While maintaining sessions
    - L3 routing
    - L2 keepalives
    - Link management

- **Prevent software upgrade from affecting SLA**
  - Service interruption
  - Packet loss

- **Minimize upgrade risk**
  - ISSU upgrade the same as normal upgrade

- **Provide ISSU on existing hardware platforms**
ISSU the Wrong Way

- ISSU must not be used for mix-and-match feature acquisition
- A “unique” code set sounds attractive
- Code patching for feature upgrades is bad engineering practice
  - Can result in uncontrolled, unmanageable code diversity
  - Regression testing of all possible code combinations becomes impossible
- Impossible for customer to test
ISSU Approach

- Add versioning capability to GRES and NSR replication messages
- Non-Stop Routing and Graceful RE Switchover are prerequisites
  - By extension, dual REs are prerequisite
- Both “from” and “to” code must be ISSU capable
Agenda

- Key features on hardware and software
- High availability on Routing solutions
- **High availability on Switching solutions**
- High availability on Security solutions
- Conclusion
What is a Virtual Chassis?

- Two or more switches interconnected via virtual-chassis backplane cables or extended via GbE and 10GbE uplinks operating as a single Juniper chassis system
- Simplified Management: Single management interface, single OS software version, single copy of configuration, and chassis-like slot/module/port numbering scheme
- Simplified Network Design: Single network entity, single control plane, link aggregation across VC members
- Superior Resiliency: Redundant master and backup Route Engines (REs), redundant switch backplane and power/fan modules
- Flexibility: Add more VC elements as port density grows, add more 10GbE uplinks, mix and match switch types
Chassis-Class Maintenance

1) Issue recycle command
2) Attach new switch
3) RE downloads software & config
Virtual Chassis™ Backplane Resiliency

Design for a resilient ring

1. Packets going from ge0/0/10 to ge3/0/14 originally pass through Member 4 to reach Member 3 following the shortest path
2. VC cable breaks between switch Member 0 and Member 4
3. Traffic between ge0/0/10 and ge3/0/14 is automatically re-routed through switch Members 1 and 2
Best Practice – Link Aggregation Group (LAG)

- Multiple physical interfaces configured as a single logical trunk group
- Increased bandwidth between devices
- Traffic distributed across active group ports and links
- Link and port level redundancy
- Fast failover in the event of a link or port failure
Virtual Chassis and LAG

- **Configure multiple Virtual Chassis (VC) groups**
  - Within single wiring closet
  - Across multiple wiring closets
  - Within Aggregation Layer

- **Uplinks from closet VC groups extend across multiple switches units in Aggregation Layer**
  - Simplified network design as Spanning Tree is not required
  - Increased redundancy when uplinks are distributed across multiple switches units within a single VC
  - Cost and operational savings as all uplinks are load-sharing and redundant
Redundant Trunk Group (RTG)

- **Redundant Trunk Group (RTG)** provide a fast and simplified Layer 2 link failover mechanism **without** the complexity of configuring and running Spanning-Tree Protocol

- **RTG** feature allows users to configure a L2 interface (Physical or LAG) to backup another L2 interface. These two interfaces provide mutual backup to each other.
Usage of RTG – Normal State

Aggregate Layer

Switch A  Virtual Chassis  Switch B

Active Link

Non-active Link

Access Layer

Switch C
Usage of RTG – After Primary Link Failure

Aggregate Layer

Switch A

Virtual Chassis

Switch B

Virtual Chassis

Non-active Link

Active Link

Link Failure

Access Layer

Switch C

Virtual Chassis
Agenda

- Key features on hardware and software
- High availability on Routing solutions
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
High Availability Reliability

**Active/Passive**
- Secondary device configuration mirrors primary
- Stateful failover for both firewall AND VPN
- Active sessions, NAT, VPN tunnels, security associations maintained

**Active/Active**
- Traffic split between devices – both devices are active with ALL network and VPN traffic flowing through
- Stateful failover for both firewall AND VPN
- Active sessions, NAT, VPN tunnels, security associations maintained

**Active/Active Full Mesh**
Agenda

- Key features on hardware and software
- High availability on Routing solutions
- High availability on Switching solutions
- High availability on Security solutions
- Conclusion
High Availability - Summary

- Juniper Networks address system reliability with features like:
  - Modular operating system
  - Bidirectional Forwarding Detection (BFD)
  - Graceful Restart (GR)
  - Graceful Routing Engine Switchover (GRES / NSF)
  - Non-Stop Routing (NSR)
  - In-Service Software Upgrade (ISSU)
  - Virtual Chassis
  - Redundant Trunk Group (RTG)
  - Link Aggregation Group (LAG)
  - Clustering
High Availability – Summary – cont’d

- **Reliability must be addressed at several levels**
  - System architecture
  - Node level
  - Link level
  - Upper level

- **Reliability must address both unplanned downtime…**

- **And planned downtime**
Contacts

- Dhahbyah CHAREF
  Systems Engineer
  dcharef@juniper.net
  06 75 50 38 17

- Yann Mignot
  Account Manager – Government & Institutions
  ymignot@juniper.net
  06 35 34 85 52