CISCO IOS SOFTWARE RELEASE
12.2SX6D ROUTING ENHANCEMENTS
INTERNET TECHNOLOGIES DIVISION
JULY 2004
Agenda

• Introduction
• Border Gateway Protocol (BGP) Convergence Optimization
• BGP Dynamic Peer Groups
• Incremental Shortest Path First (iSPF)
• Intermediate System-to-Intermediate System (IS-IS) Exclude Connect IP Prefix From Label Switched Path (LSP)
• Open Shortest Path First (OSPF) Fast Hellos
• OSPF LSP Throttling
• Conclusion
Introduction

- Cisco IOS® Software Release 12.2(18)SXD consolidates recent routing enhancements previously available in Releases 12.0S and 12T
- Enhancements are mainly concerned with improving scalability and convergence time
- Permit a higher degrees of routing protocol customization, enabling customers to adjust those parameters applicable for their deployment
Agenda

- Introduction
- **BGP Convergence Optimization**
- BGP Dynamic Peer Groups
- Incremental SPF
- IS-IS Exclude Connect IP Prefix From LSP
- OSPF Fast Hellos
- OSPF LSP Throttling
- Conclusion
BGP Convergence Optimization

- Refers to a series of BGP enhancements
- Cisco Routing Scalability Team analyzed the roadblocks in BGP convergence and addressed them individually
- Combination of code optimizations and deployment / configuration recommendations
- Results in this section are based on tests with 12.0S (where functionality was first released)
  - 12.2S benefits from this functionality; results should be comparable
BGP Initial Convergence

• Involves advertising 120,000 routes to hundreds of peers
  
  A vendor’s implementation of BGP plays a major role in how fast a router can converge initially

• Cisco IOS Software recently introduced a series of enhancements and fixes
  
  NOTE: all graphs show the percentage improvement in the number of BGP peers which can be supported while still converging in less than 10 minutes
BGP Initial Convergence – TCP Interaction

- Conservative interaction between BGP and TCP resulted in slow UPDATE propagation.
  - TCP frames were not being filled properly for maximum capacity.
- Solution: alter BGP/TCP interaction to fill frames completely.
- Simple solution provided a 133% increase in number of peers supported.
BGP Initial Convergence – Peer Groups

- Problem: advertise 120,000 routes to hundreds of peers. BGP will need to send a few hundred megs of data in order to converge all peers.

- Solution: use peer-groups
  - UPDATE generation is done once per peer-group
  - The UPDATEs are then replicated for all peer-group member

- Scalability and convergence is enhanced because more peers can be supported
BGP Initial Convergence – Peer Groups

- **UPDATE generation without peer-groups**
  
  The BGP table is walked once, prefixes are filtered through outbound policies, UPDATEs are generated and sent...per peer!

- **UPDATE generation with peer-groups**
  
  A peer-group leader is elected for each peer-group. The BGP table is walked once (for the leader only), prefixes are filtered through outbound policies, UPDATEs are generated and sent to the peer-group leader and replicated for peer-group members that are synchronized with the leader.

  Replicating an UPDATE is much easier/faster than formatting an UPDATE, which (unlike replication) requires a table walk and policy evaluation.
BGP Initial Convergence – Peer Groups

Synchronization

• A peer-group member is *synchronized* with the leader if all UPDATEs sent to the leader have also been sent to the peer-group member.
  
The more peer-group members stay in sync the more UPDATEs BGP can replicate.

• A peer-group member can fall out of sync for several reasons:
  
  Slow TCP throughput
  
  Rush of TCP Acks fill input queues resulting in drops
  
  Peer is busy doing other tasks
  
  Peer has a slower CPU than the peer-group leader
BGP Initial Convergence – Peer Groups

Peer-groups provide a significant increase in scalability

![Advantage of Peer Groups](image)
BGP Initial Convergence – Input Queues

• If a BGP speaker is pushing a full Internet table to a large number of peers, convergence is degraded due to enormous numbers of dropped TCP Acks (100k+) on the interface input queue

  Typical ISP gets ~½ million drops in fifteen minutes on an average route reflector

• Increasing the size of the input queue, thus reducing the number of dropped TCP Acks, improves BGP scalability, and reduces convergence
BGP Initial Convergence – Input Queues

- Rush of TCP Acks from peers can quickly fill the seventy-five spots in process level input queues
- Increasing queue depths (4096) improves BGP scalability

```
hold-queue <1-4096> in
```

<table>
<thead>
<tr>
<th>Percentage Improvement</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (zero)</td>
<td></td>
</tr>
<tr>
<td>TCP Improvements</td>
<td></td>
</tr>
<tr>
<td>Peer Groups</td>
<td></td>
</tr>
<tr>
<td>Larger Input Queues</td>
<td></td>
</tr>
</tbody>
</table>
BGP Initial Convergence – MTU Size

- Default MSS (Max Segment Size) is 536 bytes
- Inefficient for today’s POS/Ethernet networks
- Using “ip tcp path-mtu-discovery” improves convergence

```
ip tcp path-mtu-discovery
```

MTU Discovery

- Baseline (zero)
- TCP Improvements
- Peer Groups
- Larger Input Queues
- MTU Discovery

Results
BGP Initial Convergence – MTU Size

Simple config changes can give significant improvement

MTU Discovery + Larger Queues

Results

Percentage Improvements

Baseline (zero)
TCP Improvements
Peer Groups
Larger Input Queues
MTU Discovery
MTU Discovery + Larger Input Queues
**UPDATE Packing**

- A BGP UPDATE contains a group of attributes that characterize one (or more) prefixes

  Ideally, all the prefixes that have the same attributes should be advertised in the same UPDATE message (use as few messages as possible)

  For example:

  BGP tables contain 100,000 routes and 15,000 attribute combinations: user can advertise all routes with 15,000 updates if prefixes can be packed 100%

  100,000 updates indicate that the user achieves 0% update packing

- Convergence times vary greatly depending on the number of attribute combinations used in the table and on how well BGP packs updates
BGP Initial Convergence – Update Packing

• Improved update generation algorithm
  100% update packing – attribute distribution no longer makes a significant impact
  100% peer-group replication – no longer have to worry about peers staying “in sync”
BGP Initial Convergence – Update Packing

Improvement of almost 2000% for 120K routes

![Bar chart showing percentage improvement for various methods]

- Baseline (zero)
- TCP Improvements
- Peer Groups
- Larger Input Queues
- MTU Discovery
- MTU Discovery + Larger Input Queues
- Update Packing
BGP Initial Convergence – Putting It All Together

- Update packing + Peer Groups + MTU discovery + Larger input queues = > 4500% Improvement

![Bar chart showing percentage improvement for different BGP enhancements](chart.png)
BGP Initial Convergence – Summary

- Significant improvements gained just by using configuration options
  - Use peer-groups
  - Adjust input queues
  - Use path MTU discovery
- No need for network upgrades; enhancements are router specific (internal)
  - No interoperability issues
Agenda

• Introduction
• BGP Convergence Optimization
• BGP Dynamic Peer Groups
• Incremental SPF
• IS-IS Exclude Connect IP Prefix From LSP
• OSPF Fast Hellos
• OSPF LSP Throttling
• Conclusion
BGP Peer Groups

• The main **benefits** of peer-groups are:
  UPDATE replication: only one UPDATE message is created per peer-group – it is then sent to each individual member.
  Configuration grouping: all the members of a peer-group MUST have the same outgoing policy.
• Any deviation from the peer-group’s outgoing policy causes the peer not to be able to be a part of the peer-group
  Results in longer configuration files.
• Peer groups have been shown to **significantly improve convergence**
• The configuration must be **simplified** in order to encourage wide deployment of peer groups
BGP Dynamic Peer Groups

- Peer-group members must have the same outgoing policy
- Dynamic peer-groups eases the configuration by internally determining which peers have the same outgoing policy and then generating only one UPDATE for such peers
  
  No configuration needed

- Updates are replicated for each member of the group
  
  Reduced CPU and memory requirements
  
  Faster convergence
Agenda

- Introduction
- BGP Convergence Optimization
- BGP Dynamic Peer Groups
- Incremental SPF
- IS-IS Exclude Connect IP Prefix From LSP
- OSPF Fast Hellos
- OSPF LSP Throttling
- Conclusion
SPF Computation Review

• Dijkstra algorithm runs by examining each node’s LSPs in LSDB
  - Build TENT database and Path database (SPT)
  - Insert routes into routing tables
• SPF computation is triggered when receiving a new LSA
  - A new LSA can be received as a result of a link cost change or adding a stub network
• The computation usually involves all routers in the same routing area/domain
SPF Computation

• Some changes affect only a small part of the SPT, and some do not affect it at all

• Thus, it maybe unnecessary to run a “full” SPF computation when there is a topology change, or to run SPF at all when receiving a new LSA
Shortest Path Tree

Routing Topology

Cost b/w A and D change from 2 to 5

If there is a stub link, changes of the stub link will not have impact on the SPT, but SPF will run anyway
Incremental SPF – Concept

- Incremental SPF (iSPF) allows routers to intelligently determine where the impact of the change is in the SPT and then only re-computes the effected nodes to update the SPT.
- As a result, it reduces convergence time by reducing SPF processing time.
- Amount of convergence time and CPU cycles saved depend on how many nodes that Dijkstra algorithm would need to examine with and without iSPF.

  The amount of convergence time saved tends to increase as the user moves further from the change.
Incremental SPF – Configuration and Deployment

**OSPF Configuration**

```
router ospf 1
[no] incremental-spf
```

**ISIS Configuration**

```
router isis
    incremental-spf [level-1|level-2|level-1-2] [<1-100>]
```

Final parameter [<1-100>] is number of full Dijkstra runs which will be performed before incremental runs begin

Ideal for routing area/domain with large number of nodes and/or stub links
Incremental SPF – OSPF Debug Output

debug ip ospf spf statistic

Without iSPF

• OSPF: Begin SPF at 188927.520ms, process time 149760ms
• OSPF: End SPF at 188927.572ms, Total elapsed time 52ms
• Intra: 48ms, Inter: 0ms, External: 0ms
• R: 488, N: 758, Stubs: 598
• SN: 0, SA: 0, X5: 0, X7: 0

With iSPF

• OSPF: Begin SPF at 188687.524ms, process time 149612ms
• OSPF: End SPF at 188687.536ms, Total elapsed time 12ms
• Incremental-SPF: 0ms
• Intra: 8ms, Inter: 0ms, External: 0ms
• R: 18, N: 29, Stubs: 22
• SN: 0, SA: 0, X5: 0, X7: 0
Agenda

- Introduction
- BGP Convergence Optimization
- BGP Dynamic Peer Groups
- Incremental SPF
- **IS-IS Exclude Connect IP Prefix From LSP**
- OSPF Fast Hellos
- OSPF LSP Throttling
- Conclusion
Why Exclude Connected Prefixes?

- In large Internet Service Provider (ISP) networks, IS-IS may be used *solely* to get the next-hop address for BGP prefixes.
- Only the *loopback address* of the router needs to be in IS-IS.
- By default, IS-IS will advertise all connected interfaces, eases configuration for full IS-IS networks.
- This results in large IS-IS *link-state databases*.
- Cisco IOS Software Release 12.2(18)SXD adds configuration option to suppress this default behavior.
Configuration of IS-IS Excluded Prefixes

- On a per-interface basis:
  ```
  interface ethernet 1/0
  no isis advertise prefix
  ```

- On a per-router basis:
  ```
  router isis
  advertise passive-only
  ```

Note: although the same effect can be achieved by using unnumbered interfaces, ISPs prefer numbered interfaces for management purposes.
Agenda

• Introduction
• BGP Convergence Optimization
• BGP Dynamic Peer Groups
• Incremental SPF
• IS-IS Exclude Connect IP Prefix From LSP
• OSPF Fast Hellos
• OSPF LSP Throttling
• Conclusion
Fast Hellos – The Problem

• As customers converge more mission-critical applications onto their IP infrastructure, the ability to quickly reroute around failures is critical

• OSPF uses a “HELLO” mechanism to detect failure

• HELLOs are sent at <hello-interval time>; If no HELLO seen in <dead-interval time>, traffic reroute begins

• Default timers are acceptable for most applications

• However, some specialized applications (ie: voice, financial trading, military) may require very aggressive timers
OSPF Fast Hellos

- Allows the dead-interval to be set at one second, allowing near instantaneous failure detection

```plaintext
int ethernet 1/0
  ip ospf dead-interval minimal hello-multiplier <3-20>
```

- **Warning:** lowering the dead-interval to one second also raises the risk of “false positives”
- Customers should verify behavior in a lab that accurately emulates their production environment before deploying
Agenda

- Introduction
- BGP Convergence Optimization
- BGP Dynamic Peer Groups
- Incremental SPF
- IS-IS Exclude Connect IP Prefix From LSP
- OSPF Fast Hellos
- **OSPF LSP Throttling**
- Conclusion
OSPF Event Propagation

- On an OSPF network, after a network event has been detected, an LSA is generated to reflect the change.

- LSA is not generated immediately.

**OSPF_LSA_DELAY_INTERVAL** – 500ms delay (fixed) used when generating Router and Network LSA.

**MinLSInterval** – minimum time between distinct originations of any particular LSA; value of MinLSInterval is set to 5 seconds.
OSPF Event propagation (Cont.)

- The reason for this delay is to collect any changes that occur during the delay interval and include them all in the new LSA
- This protects routers from generating LSAs too frequently if the interface(s) keeps flapping
- While this timer promotes network stability, it can also delay convergence
Delay in Event Propagation Example

Delay in convergence

R1 R2 R3 R4

Event detect

LSA Generation Attempt

LSA Build

500 msec 1 sec 500 msec

1st

2nd

5 sec

3.5 sec
OSPF LSA Throttling

• Enables fast LSA propagation while maintaining stability
• Uses back-off algorithm to generate all LSA as opposed to a constant 5 sec delay
• Introduces three timers (unit: msec)
  <initial>: initial delay for generating the first LSA (1-5000)
  <start>: minimum delay while generating LSAs (1-10000); used as a multiplier for consecutive LSA generations
  <max>: maximum wait time while generating LSAs (1-100000)
Throttling Back-off Algorithm and Stability

- timers lsa throttle all <initial> <start> <max>

- LSA Build
  - N*b is less than or equal to max - min[N*b, c]
  - Algorithm restart if no SPF event within 2 * c

- LSA throttling back-off algorithm absorbs routing-churn effect
OSPF LSA Throttling and Convergence

- LSA throttling allows traffic to switch to the alternative path faster, and
- Dampens route-churning during rapid network changes
Agenda

- Introduction
- BGP Convergence Optimization
- BGP Dynamic Peer Groups
- Incremental SPF
- IS-IS Exclude Connect IP Prefix From LSP
- OSPF Fast Hellos
- OSPF LSP Throttling
- Conclusion
Conclusion

• Cisco IOS Software Release 12.2(18)SXD incorporates significant routing enhancements from other Cisco IOS Software releases

• Enhancements designed to provide the end-user with better:
  - Convergence optimization
  - Flexibility
  - Ease of deployment