Design space of computing metric distribution

Hang Shi, Zongpeng Du, Xinxin Yi, Tianle Yang
Huawei, China Mobile, China Unicom, China Broadcast Mobile Network Company
IETF 119
Recap of the CATS framework

• Core functional components:
  • C-SMA: CATS Service Metric Agent
  • C-PS: CATS Path Selector

• SMA collect the computing metric and distribute it to PS to make optimal path decision.

• Design choice regarding:
  • How to collect
  • How to distribute
Previously in this draft

Centralized C-PS + Centralized C-SMA

Distributed C-PS + Centralized C-SMA

Centralized C-PS + Distributed C-SMA

Distributed C-PS + Distributed C-SMA
What’s new: Metric distribution overhead

• Regardless of distributed/centralized. Two sides of the metric distribution: Producer(C-SMA) + Consumer(C-PS)

Metric Distribution Overhead =
No of Producer x No of Consumer x Distribution Frequency x Metric size

Metric distribution scope

Metric Distribution frequency

Metric Model

IETF 119 Brisbane
Reduce the scope of metric distribution

• Restrict the scope of metric distribution to ingresses that actually needs the metric
• Notification domain. [draft-fu-idr-computing-info-notification-domain-01 - Computing resource notification domain in network (ietf.org)]

Option 1: Pub/sub relationship

Option 2: Hierarchical. Organize the sites into different latency ring for each sites. If the distance is too far away(highly unlikely to be scheduled), distribution of detail metric can be omitted.

Figure 1: BGP Service Metric Route Process

draft-lin-idr-distribute-service-metric-00 - Distribute Service Metric By BGP (ietf.org)
draft-yi-cats-hierarchical-metric-distribution-00 - Hierarchical methods of computing metrics distribution (ietf.org)
Reduce Distribution Frequency: Push versus Pull

- The optimization depends on the Push versus Pull.
- Both optimization is a trade off between the metric freshness and distribution overhead.

<table>
<thead>
<tr>
<th>Distribution Frequency</th>
<th>Push</th>
<th>Pull</th>
</tr>
</thead>
<tbody>
<tr>
<td>Without optimization</td>
<td>When there is new metric</td>
<td>When there is new client join</td>
</tr>
<tr>
<td>Optimization</td>
<td>Threshold. Only push</td>
<td>Cache. Only fetch when cache expired</td>
</tr>
<tr>
<td></td>
<td>meaningful update</td>
<td></td>
</tr>
</tbody>
</table>
Next

• Intend to inform the discussion around the metric distribution protocol design
• Comments, suggestion, contribution is more than welcomed