BGP update profiles and the implications for secure BGP update validation processing

Geoff Huston

PAM2007

5 April 2007
Why?

- Secure BGP proposals all rely on some form of validation of BGP update messages
- Validation typically involves cryptographic validation, and may refer to further validation via a number resource PKI
- This validation may take considerable resources to complete.
- This implies that the overheads securing BGP updates in terms of validity of payload may contribute to:
  - Slower BGP processing
  - Slower propagation of BGP updates
  - Slower BGP convergence following withdrawal
  - Greater route instability
  - Potential implications in the stability of the forwarding plane
What is the question here?

- Validation information has some time span
  - Validation outcomes can be assumed to be valid for a period of hours

- Should BGP-related validation outcomes be locally cached?

- What size and cache lifetime would yield high hit rates for BGP update validation processing?
Method

- Use a BGP update log from a single eBGP peering session with AS 4637 over a 14 day period
- Examine time and space distributions of BGP Updates that have similar properties in terms of validation tasks
Update Statistics for the session

<table>
<thead>
<tr>
<th>Day</th>
<th>Prefix Updates</th>
<th>Duplicates: Prefix</th>
<th>Duplicates: Prefix + Origin AS</th>
<th>Duplicates: Prefix + AS Path</th>
<th>Duplicates: Prefix + Comp-Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>72,934</td>
<td>60,105 (82%)</td>
<td>54,924 (75%)</td>
<td>34,822 (48%)</td>
<td>35,312 (48%)</td>
</tr>
<tr>
<td>2</td>
<td>79,361</td>
<td>71,714 (90%)</td>
<td>67,942 (86%)</td>
<td>49,290 (62%)</td>
<td>50,974 (64%)</td>
</tr>
<tr>
<td>3</td>
<td>104,764</td>
<td>93,708 (89%)</td>
<td>87,835 (84%)</td>
<td>65,510 (63%)</td>
<td>66,789 (64%)</td>
</tr>
<tr>
<td>4</td>
<td>107,576</td>
<td>94,127 (87%)</td>
<td>87,275 (81%)</td>
<td>64,335 (60%)</td>
<td>66,487 (62%)</td>
</tr>
<tr>
<td>5</td>
<td>139,483</td>
<td>110,994 (80%)</td>
<td>99,171 (71%)</td>
<td>68,096 (49%)</td>
<td>69,886 (50%)</td>
</tr>
<tr>
<td>6</td>
<td>100,444</td>
<td>92,944 (92%)</td>
<td>88,765 (88%)</td>
<td>70,759 (70%)</td>
<td>72,108 (72%)</td>
</tr>
<tr>
<td>7</td>
<td>75,519</td>
<td>71,935 (95%)</td>
<td>69,383 (92%)</td>
<td>56,743 (75%)</td>
<td>58,212 (77%)</td>
</tr>
<tr>
<td>8</td>
<td>64,010</td>
<td>60,642 (95%)</td>
<td>57,767 (90%)</td>
<td>49,151 (77%)</td>
<td>49,807 (78%)</td>
</tr>
<tr>
<td>9</td>
<td>94,944</td>
<td>89,777 (95%)</td>
<td>86,517 (91%)</td>
<td>71,118 (75%)</td>
<td>72,087 (76%)</td>
</tr>
<tr>
<td>10</td>
<td>81,576</td>
<td>78,245 (96%)</td>
<td>75,529 (93%)</td>
<td>63,607 (78%)</td>
<td>64,696 (79%)</td>
</tr>
<tr>
<td>11</td>
<td>95,062</td>
<td>91,144 (96%)</td>
<td>87,486 (92%)</td>
<td>72,678 (76%)</td>
<td>74,226 (78%)</td>
</tr>
<tr>
<td>12</td>
<td>108,987</td>
<td>103,463 (95%)</td>
<td>99,662 (91%)</td>
<td>80,720 (74%)</td>
<td>82,290 (76%)</td>
</tr>
<tr>
<td>13</td>
<td>91,732</td>
<td>87,998 (96%)</td>
<td>85,030 (93%)</td>
<td>72,660 (79%)</td>
<td>74,116 (81%)</td>
</tr>
<tr>
<td>14</td>
<td>78,407</td>
<td>76,174 (97%)</td>
<td>74,035 (94%)</td>
<td>64,994 (83%)</td>
<td>65,509 (84%)</td>
</tr>
</tbody>
</table>
CDF by Prefix and Originating AS

BGP Prefix Update Cumulative Distribution

BGP Origin AS Update Cumulative Distribution
Time Distribution

Cumulative Total of Recurring Updates

Update Recurrence Interval (Hours)

Prefixes
Prefix + Origins
Prefix + Path
Prefix + Compressed Path
Space Distribution

- Use a variable size cache simulator
- Assume 36 hour cache lifetime
- Want to know the hit rate of validation queries against cache size
Prefix Similarity

![Graph showing validation cache hit percentage against cache size]
Prefix + Origin Similarity

Validation Cache Hit % vs Cache Size
Prefix + Path Similarity

![Graph showing validation cache hit percentage vs cache size]

- Validation Cache Hit %
- Cache Size

The graph illustrates the relationship between the cache size and the validation cache hit percentage. The data points are represented by different colored lines, indicating variations in the dataset or conditions under which the cache performance is measured. The horizontal shaded area highlights a specific range of cache sizes where the cache hit percentage stabilizes, suggesting an optimal cache size for efficient performance.
Observations

- A large majority of BGP updates explore diverse paths for the same origination
- True origination instability occurs relatively infrequently (1:4)?
- Validation workloads can be reduced by considering origination (prefix plus origin) and the path vector as separable validation tasks
- Further processing reduction can be achieved by treating a AS path vector as a sequence of AS paired adjacencies
AS Pair Similarity

Validation Cache Hit %

Cache Size
Observations

• Validation caching appears to be a useful approach to addressing some of the potential overheads of validation of BGP updates
• Separating origination from path processing, using a 36 hour validation cache can achieve 80% validation hit rate using a cache of 10,000 Prefix + AS originations and a cache of 1,000 AS pairs
What do we want from secure BGP?

• Validation that the received BGP Update has been processed by the ASs in the AS Path, in the same order as the AS Path, and reflects a valid prefix, valid origination and valid propagation along the AS Path?

or

• Validation that the received Update reflects a valid prefix and valid origination, and that the AS Path represents a plausible sequence of validated AS peerings?
Thank You