Suricata - Feature #548
Use bloomfilter for filemd5
09/10/2012 03:09 AM - David André

<table>
<thead>
<tr>
<th>Status:</th>
<th>New</th>
<th>Priority:</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignee:</td>
<td>Community Ticket</td>
<td>Category:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td>TBD</td>
<td>Effort:</td>
<td>low</td>
</tr>
<tr>
<td>Difficulty:</td>
<td>medium</td>
<td>Label:</td>
<td></td>
</tr>
</tbody>
</table>

Description
To reduce memory usage, use bloom filters.

Background:
Bloom filters are very memory efficient probabilistic data-structures that don't have false negatives but have false positives.

Pros:
There is already code implemented in suricata source
It is very efficient for blacklists.

Cons:
It might not be efficient for whitelists.

Notes:
Since it has false positives, it would probably be necessary to do a second level validation lookup from data on disk and it will be more expensive.
Implementing through a different keyword (filemd5bloom?) will help avoiding misuse by users.

History
#1 - 09/10/2012 04:45 AM - Victor Julien
I think it could make sense for performance, but not for reducing memory. As you mentioned, after a bloom match you will still need to do a validation step. We can't do that from disk, it would be too slow.

#2 - 09/28/2012 04:41 AM - Victor Julien
- Assignee set to Anonymous
- Target version set to TBD

Performance seems to be OK with the current code as well btw.

#3 - 12/22/2015 04:06 PM - Andreas Herz
Victor Julien wrote:

Performance seems to be OK with the current code as well btw.

Worth closing the ticket?

#4 - 01/17/2016 06:45 AM - Victor Julien
No I changed my mind a bit, think it would be interesting to look into this. Still low priority though.

#5 - 07/13/2018 08:16 PM - Victor Julien
- Effort set to low
- Difficulty set to medium

#6 - 02/23/2019 10:10 PM - Andreas Herz
- Assignee set to Community Ticket