DOTS telemetry related Hackathon activity report

IETF 106 Hackathon
Kaname Nishizuka/NTT Communications
Hackathon Plan

- Draft

- Preliminary implementation and PoC of DOTS telemetry

- Design Review of DOTS telemetry
Design of DOTS telemetry

Purpose
• Giving a maximum capability of conveying normal/attack traffic related metrics as hints from a DOTS client to a DOTS server and vice versa.

Timing
• pre-mitigation
• post-mitigation
• pre-mitigation
pre-mitigation telemetry

Pre-mitigation resources are not bound to any mitigation request.

**URI-Path: “telemetry”**

<table>
<thead>
<tr>
<th>module</th>
<th>explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>telemetry-config</td>
<td>configuration of telemetry</td>
</tr>
<tr>
<td>total-*</td>
<td>baseline/capacity</td>
</tr>
<tr>
<td>attack-detail</td>
<td>attack information</td>
</tr>
</tbody>
</table>

**Proposal**

- Separate the URI-path of them into 2
  - update of attack-detail doesn’t always require configuration update
DOTS client to server telemetry

URI-Path: “telemetry-config” (proposal)

<table>
<thead>
<tr>
<th>module</th>
<th>explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>telemetry-config</td>
<td>configuration of telemetry</td>
</tr>
<tr>
<td>total-*</td>
<td>baseline/capacity</td>
</tr>
<tr>
<td>attack-detail</td>
<td>attack information</td>
</tr>
</tbody>
</table>

PUT: convey the telemetry configuration
GET: retrieve the negotiated configuration
DELETE: delete and set the parameters to default values

Theoretically it works well with “tcid” (=Telemetry Configuration Identifier)
Machine Learning approach
consideration

1. Sending “normal traffic baseline” calculated at a DOTS client
2. Sending traffic metrics periodically, then “normal traffic learning” at a DOTS server
DOTS client to server telemetry

URI-Path: "pre-mitigation" (proposal)

<table>
<thead>
<tr>
<th>module</th>
<th>explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>attack-detail</td>
<td>attack information</td>
</tr>
<tr>
<td>normal-traffic</td>
<td>normal traffic related information</td>
</tr>
</tbody>
</table>

PUT: convey the current information of attack/normal traffic **from a DOTS client**

GET: retrieve the (historical) traffic information

DELETE: delete all the traffic information

(timestamp will be needed for normal-traffic)

NOTE: If DOTS agents send traffic metrics, it needs to be compared with other approaches like IPFIX
DOTS server to client telemetry

URI-Path: “pre-mitigation-attackinfo” (proposal)

<table>
<thead>
<tr>
<th>module</th>
<th>explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>attack-detail</td>
<td>attack information</td>
</tr>
<tr>
<td>normal-traffic</td>
<td>normal traffic related information</td>
</tr>
</tbody>
</table>

- Observe Option set to '0' in the GET request
- receive asynchronous notifications of attack-detail from the DOTS server.
Why S-to-C attack info in pre-mitigation stage

Inconsistency in attack knowledge

Scenario

• When a DDoS attack happened, SOC at the DOTS client side can notice something going wrong but cannot figure out which IP address is exactly attacked

• What if SOC at the DOTS server can convey attack-detail to the DOTS client?

• The DOTS client can finally trigger a mitigation request based upon the hint gave via telemetry
• post-mitigation
post-mitigation telemetry

Post-mitigation resources are bound to existing mitigation-scope.

<table>
<thead>
<tr>
<th>module</th>
<th>explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>attack-detail</td>
<td>attack information</td>
</tr>
</tbody>
</table>

Client to Server

• Sent in initial mitigation request (PUT)
• Sent as a part of efficacy update (PUT)

Server to Client

• Sent as a part of mitigation status update

• No new URI-path will be need
• Considerations
percentile calculation

Context of percentile calculation
• period of time (1 hour, 1 day ... 1 month)
• time granularity (1 sec, 1 min, 5 min ...)

Inconsistency of them between the DOTS client and server will lead to misunderstanding of attack characteristics
Inconsistency of definition of “request” and “partial-request” will also lead to misunderstanding of attack characteristics (couldn’t find exact definition of what is “request” here)
• Interop status updates
Interop status updates

• Continuous interop testing with Jon (after the last IETF)
• Found several bugs on both sides. There is no significant issue on signal-channel(-38) and data-channel(-31) except one(*)
• Both are willing to test the new DOTS heartbeat spec introduced from v39 draft ASAP
  • will be reported back to WG
• go-dots: https://github.com/nttdots/go-dots
  • kubernetes deployment will be available soon

* usage of RST to cancel Observe requests will not work with DOTS gateway.
Thank You