NetMon a tool for multi-user network service monitoring and fault localization

Pavle Vuletić, (University of Belgrade)

5th SIG_PMV 2018, Manchester UK
• Virtualized infrastructures – multiple users traffic is multiplexed over the same physical links
• How to estimate the quality of service/experience of each user separately?
• Monitoring physical infrastructure is not sufficient and using separate tools for each virtual network is not scalable
• Various network technologies are used for multiplexing users traffic (different VPN flavours, L2, L3, e-circuits, etc.). Goal: create a single, scalable, vendor independent monitoring platform capable to monitor all these technologies
• Automated monitoring upon network service installation integrated with the provisioning
• Fault localization – where is performance degradation on the end-to-end path?
NetMon approach

• NetMon provides:
  • real-time feedback to network operations personnel or users,
  • determines whether services are performing to spec (SLA verification),
  • if not, it initiates an automated analysis to localise the fault, and notify the appropriate agent to take corrective action.

• Key performance indicators:
  • MEF (10.3) and ITU defined metrics: delay, jitter, loss, availability, etc.

• Getting the metrics – hybrid approach (RFC 7799).

• Key components:
  • Monitoring Controller
  • Multihomed Monitoring Agents
  • Monitoring Result Repository and portal
  • (Capturers and Correlators for fault localization)

• 3 modes of operation: active – end-to-end, active+fault localization, full traffic analysis
Workflow integrated with the rest of the OSS/BSS Architecture (Mode 1)

SPA
- Other OSS/BSS
- Service inventory

NetMon
- Monitoring controller
- Result repository/portal

Service 1
Service 2
Workflow integrated with the rest of the OSS/BSS Architecture (Mode 1)
Workflow integrated with the rest of the OSS/BSS Architecture (Mode 1)

SPA
- Other OSS/BSS
- Service inventory

NetMon
- Monitoring controller
- Result repository/portal

Service 1
Service 2
Workflow integrated with the rest of the OSS/BSS Architecture (Mode 1)
Workflow integrated with the rest of the OSS/BSS Architecture (Mode 1)
Workflow integrated with the rest of the OSS/BSS Architecture (Mode 1)
Workflow integrated with the rest of the OSS/BSS Architecture (Mode 1)
Compatibility and technology

• TMF Service Test API
• TMF Service inventory API
• TMF Trouble ticket API
• Monitoring @100G

• Integrate proven solutions:
  • Active probing - modified
  • Component configuration
  • Inter-component communication
  • Result database
  • Result display
Network setup for the demo
• Between MX routers (PODs): CCC L2VPN
• Between VMX: native IP, L2VPN, L3VPN
• Total on the wire:
• 100 – Multipoint L3VPN
• (200 – p2p L3VPN)
• 300 – Multipoint L2VPN
• 400 – point to point L2 VPN
• Native IP communication between the CPE/MA devices
• In the example, we turn on and off VPN 200 monitoring and change the delay on the selected network path in the network
Initiating Monitoring Session
## Monitoring devices

### Service Tests

### Specifications

### Devices

<table>
<thead>
<tr>
<th>Name</th>
<th>Location</th>
<th>Domain</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA1</td>
<td>London</td>
<td>GTS5</td>
<td>MA1 in GTS5</td>
</tr>
<tr>
<td>MA2</td>
<td>London</td>
<td>GTS5</td>
<td>MA2 in GTS5</td>
</tr>
<tr>
<td>MA3</td>
<td>Prague</td>
<td>GTS5</td>
<td>MA3 in GTS5</td>
</tr>
<tr>
<td>MA4</td>
<td>Prague</td>
<td>GTS5</td>
<td>MA4 in GTS5</td>
</tr>
</tbody>
</table>
### Device information

**Basic Information**
- **Name:** MA1
- **Location:** London
- **Domain:** GTS5
- **Description:** MA1 in GTS5
- **ID:** 08c05d80-416f-11e8-a40b-8d43421f649a
- **Href:** http://172.16.0.74:8081/control/device/08c05d80-416f-11e8-a40b-8d43421f649a

**Management interface**
- **Interface name:** eth0
- **IP address:** 172.16.0.138

**Operational interface**
- **Interface name:** eth1
- **Role:** Measurement Agent (MA)
- **Attachment element:** CPE1
- **Attachment port:** 1
## Test specifications

### Service Test Specifications

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STS-Inproduction (delay, jitter, loss)</td>
<td>Inproduction performance verification (delay, jitter, loss)</td>
</tr>
<tr>
<td>STS-Inproduction (delay, loss)</td>
<td>Inproduction performance verification (delay, loss)</td>
</tr>
<tr>
<td>STS-Preproduction</td>
<td>Preproduction test based on ping</td>
</tr>
</tbody>
</table>
### Measure definition 1

- **Name:** E2D Delay
- **Metric href:** e2e-delay
- **Metric name:** e2e delay
- **Metric description:** end to end delay
- **Unit of measure:** ms
- **Value type:**
- **Capture method:** inproduction-test-mode1
- **Capture frequency:** 60s
- **Threshold rules:** [Show](#)

### Measure definition 2

- **Name:** E2E Jitter
- **Metric href:** e2e-jitter
- **Metric name:** e2e jitter
- **Metric description:** end to end jitter
- **Unit of measure:** ms
- **Value type:**
- **Capture method:** inproduction-test-mode1
- **Capture frequency:** 60s
- **Threshold rules:** [Show](#)

### Measure definition 3

- **Name:** E2E Loss
- **Metric href:** e2e-loss
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Service</th>
<th>Status</th>
<th>Start Time</th>
<th>End Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inproduction test (v100)</td>
<td>Inroduction test on v100</td>
<td>v100</td>
<td>In Progress</td>
<td>2018-06-10T20:20:42.486Z</td>
<td>2018-06-10T20:51:52.874Z</td>
</tr>
<tr>
<td>Inproduction test (v100)</td>
<td>Inroduction test on v100</td>
<td>v100</td>
<td>Failed</td>
<td>2018-06-10T20:19:45.084Z</td>
<td>2018-06-10T20:21:28.668Z</td>
</tr>
<tr>
<td>Inproduction test (v100)</td>
<td>Inroduction test on v100</td>
<td>v100</td>
<td>Failed</td>
<td>2018-06-10T20:16:20:155Z</td>
<td>2018-06-10T20:18:03:941Z</td>
</tr>
</tbody>
</table>
### Create new Service Test

#### Properties

- **Name:** v200 - TNC
- **Description:** v200 test during the TNC'18

#### Service Test Specification

- **Name:** STS-Inproduction (delay, jitter, loss)
- **Href:** [http://172.16.0.74:8081/serviceTestManagement/serviceTestSpecification/8c3caf5b-4170-11e8-c](http://172.16.0.74:8081/serviceTestManagement/serviceTestSpecification/8c3caf5b-4170-11e8-c)

#### Related Service

- **Name:** v200
- **Href:** [http://172.16.0.74:8081/control/service/8a86b3e0-4a21-11e8-9762-1bd5819e338c](http://172.16.0.74:8081/control/service/8a86b3e0-4a21-11e8-9762-1bd5819e338c)
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Service</th>
<th>Status</th>
<th>▼Start Time</th>
<th>End Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>v200 - TNC</td>
<td>v200 test during the TNC’18</td>
<td>v200</td>
<td>In Progress</td>
<td>2018-06-11T11:53:03.530Z</td>
<td></td>
</tr>
<tr>
<td>Inproduction test (v300)</td>
<td>Inproduction test on v300</td>
<td>v300</td>
<td>In Progress</td>
<td>2018-06-11T07:33:27.807Z</td>
<td></td>
</tr>
<tr>
<td>Inproduction test (v100)</td>
<td>Inproduction test on v100</td>
<td>v100</td>
<td>In Progress</td>
<td>2018-06-10T20:20:42.486Z</td>
<td></td>
</tr>
<tr>
<td>Inproduction test (v100)</td>
<td>Inproduction test on v100</td>
<td>v100</td>
<td>Failed</td>
<td>2018-06-10T20:16:20.155Z</td>
<td>2018-06-10T20:18:03.941Z</td>
</tr>
<tr>
<td>Service ID 53</td>
<td>LondonCPE1</td>
<td>LondonCPE2</td>
<td>PragueCPE3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
<td>------------</td>
<td>------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LondonCPE1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| LondonCPE2   |            | Average Delay - 2.373
              |            | Average Jitter - 1.075 |
|              |            |            |            |
| PragueCPE3   |            | Average Delay - 62.886
<pre><code>          |            | Average Jitter - 2.255 |
</code></pre>
<p>|              |            |            |            |
| Service ID 56 | LondonCPE2 | PragueCPE3 |
|              |            |            |            |
| LondonCPE2   |            |            |            |
|              |            |            |            |
| PragueCPE3   |            |            |            |</p>
<table>
<thead>
<tr>
<th>Service ID</th>
<th>Location</th>
<th>Average Delay</th>
<th>Average Jitter</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>London CPE 1</td>
<td>2.371</td>
<td>1.074</td>
</tr>
<tr>
<td>56</td>
<td>London CPE 2</td>
<td>3.159</td>
<td>1.115</td>
</tr>
<tr>
<td>53</td>
<td>Prague CPE 3</td>
<td>62.883</td>
<td>2.255</td>
</tr>
<tr>
<td>56</td>
<td>London CPE 2</td>
<td>63.484</td>
<td>2.206</td>
</tr>
<tr>
<td>53</td>
<td>Prague CPE 3</td>
<td>62.725</td>
<td>2.155</td>
</tr>
<tr>
<td>56</td>
<td>Prague CPE 3</td>
<td>63.3</td>
<td>3.4</td>
</tr>
<tr>
<td>53</td>
<td>Prague CPE 3</td>
<td>62.5</td>
<td>1.6</td>
</tr>
</tbody>
</table>
End-to-end Monitoring
Fault localization
Fault localization (Mode 2 and 3)

- It is necessary to get the information from the intermediate points in the network
- Similar approaches:
  - Single technology (CFM) or vendor/proprietary solutions
Fault localization (Mode 2 and 3)

- It is necessary to get the information from the intermediate points in the network
- Similar approaches:
  - Single technology (CFM) or vendor/proprietary solutions
  - Concept of the monitoring zone

Taken from: Ericsson Diamond: https://pdfs.semanticscholar.org/0119/099638d68a0836d55d7de0dfc00891571876.pdf
Fault localization (Mode 2 and 3)

• It is necessary to get the information from the intermediate points in the network

• Similar approaches:
  • Single technology (CFM) or vendor/proprietary solutions
  • Concept of the monitoring zone
  • Flow Broker
Fault localization (Mode 2 and 3)

- It is necessary to get the information from the intermediate points in the network
- Similar approaches:
  - Single technology (CFM) or vendor/proprietary solutions
  - Concept of the monitoring zone
  - Flow Broker
  - IETF RFC 8321 (Jan 2018) – Alternate marking (requires changes in the network elements)
Fault localization (Mode 2 and 3)

- It is necessary to get the information from the intermediate points in the network
- Similar approaches:
  - Single technology (CFM) or vendor/proprietary solutions
  - Concept of the monitoring zone
  - Flow Broker
  - IETF RFC 8321 (Jan 2018) – Alternate marking (requires changes in the network elements)
  - IETF RFC 8372 (May 2018) – MPLS Flow identification
Fault localization (Mode 2 and 3)

• It is necessary to get the information from the intermediate points in the network

• Similar approaches:
  • Single technology (CFM) or vendor/proprietary solutions
  • Concept of the monitoring zone
  • Flow Broker
  • IETF RFC 8321 (Jan 2018) – Alternate marking (requires changes in the network elements)
  • IETF RFC 8372 (May 2018) – MPLS Flow identification

• NetMon approach
  • Specially crafted OWAMP packets (serviceID)
  • Captured at various points in the network
  • Matched based on the packet hash and service ID
  • Packet digest sent to the Correlator and from there to the Result repository

```
owping -s 30 -x 04000064AC100016 192.168.100.2:8765
```
Can NetMon be merged with perfSONAR?

• NetMon uses active monitoring approach (but no BW tests)
• NetMon uses the same key monitoring tool (owamp/twamp)
• perfSONAR recently adopted the work in netnamespaces (multihoming – multi-tenant operation)
• perfSONAR has well organized development process and a long history of successful deployments
• Key gaps:
  • Service awareness (use case: Service X operating between CPE A, B, C over VLANs 100, 200, 300 on interfaces eth2, eth1, eth2 respectively. KPI for Service X: delay, jitter and loss. SLA specification, RAG alarm thresholds, signaling towards other components)
  • Integration with the other OSS/BSS components (extracting the required data from other inventories, receiving monitoring orders, sending alarms)
  • Fault localization
  • perfSONAR plans
Thank you
Any Questions?