OSM-MR#9 Remote
HD4.3 Closed-Loop Operations
Adding Auto-Scaling & Alerting to VNFs

Subhankar Pal
(Altran)
Current Auto Scaling & Alarms Feature
OSM Service Assurance
Revisiting Service Assurance MDG

**Main components**

**MON**
- Covers the basic use cases, with a solid architecture to expand them easily.
- Opportunities to enhance usability.

**POL**
- Designed around the autoscaling use case.
- Starting to cover VNF alarms.
- Architecture needs a revisit based on expected use cases.

**Prometheus**
- OSM’s TSDB for metrics since REL5
- Opportunities to enhance multi-tenancy to match new RBAC capabilities.

**Grafana**
- Integrates seamlessly with Prometheus.
- Great tool for enhancing usability of the system’s Service Assurance.

**ELK**
- Proved seamless integration with OSM.
- Main use case remains at log processing where stack is used.

**Auxiliary/Optional**

- And an upcoming Placement module!
## Auto Scaling & Alarms Features

### Auto Scaling
- Auto scaling allows to automatically scale VNFs with a VDU granularity and based on any available metric.
- Scaling descriptors can be included and be tied to automatic reaction to VIM/VNF metric thresholds.
- Supported metrics are both VIM and VNF metrics.

### Alarms
- An internal alarm manager has been added to MON through the 'mon-evaluator' module, so that both VIM and VNF metrics can also trigger threshold-violation alarms and scaling actions.
Revisiting MON Architecture

Formal documentation: https://osm.etsi.org/gitlab/osm-architecture/osm-arch-doc/blob/master/04-mon.md
POL Architecture

Formal documentation: https://osm.etsi.org/gitlab/osm-architecture/osm-arch-doc/blob/master/05-pol.md
When configuring alarms associated to scaling actions or just webhook notifications (through the VNFD), the following components interact.

- `lcm`: configured alarms can be queried
- `pol`: policy manager module (POL) takes actions such as auto-scaling.
- `mon-evaluator`: evaluates thresholds related to metrics
- `tsdb (prometheus)`: when triggered, puts alarm in bus for POL to take actions
- `webhook service (external)`: if action is to notify, sends notification to webhook service

- Whenever a threshold is crossed and an alarm is triggered, the notification is generated by MON and put in the Kafka bus so other components, like POL can consume them.
The scaling descriptor is part of a VNFD. Like the example shows, it mainly specifies:

- An existing metric to be monitored, which should be pre-defined in the monitoring-param list (vnf-monitoring-param-ref).
- The thresholds to monitor (scale-in/out-threshold)
- The minimum and maximum amount of scaled instances to produce.
- The minimum time it should pass between scaling operations (cooldown-time)
- The VDU to be scaled (vdu-id-ref) and the amount of instances to scale per event (count)
Alarm Descriptors

Alarms based on metric thresholds can be sent to webhooks. The alarm descriptor is also part of a VNFD. Like the example shows, it mainly specifies:

• An existing metric to be monitored, which should be pre-defined in the monitoring-param list (vnf-monitoring-param-ref).

• The thresholds to monitor (alarm-threshold)

• The web hook to be invoked (url)

```
- alarm:
  - actions:
    alarm:
      - url: https://webhook.site/5706da10-04a0-4ab0-819b-cb524f71a367
        alarm-id: cpu-above-threshold
        operation: GT
        value: 80
        vnf-monitoring-param-ref: agw_cpu_util
```
New Proposals

OSM Service Assurance
Closed-loop automation powers autonomous networks.

1. Observe
   Collect network metrics through different telemetry interfaces.

2. Decide
   Processes collected metrics to determine the network status, decides action to be taken based on network policies. This phase is not responsible for executing the action.

3. Act
   Acts upon orchestrated object and implements given lifecycle action.
Auto-Scaling & Alarms – New Architecture

VIM Alarms (aodh)

1a) Aodh notification

VNF metrics (prometheus)

1b) AlertManager notification

OSM VNF Metric Exporters

VNF Alarms (ex. snmp traps → webhook)

1c) Custom webhooks

In the future, POL could act upon combined alarms

(2c) if action is to scale: send to bus for LCM to proceed and store action to commonDB

lcm

(2d) if action is to notify, send notification to webhook service

webhook service (external)
Other Features in Roadmap

- Move away from threshold to ML based anomaly detection
- Improved Closed Loop Operation
  - Dynamic Thresholds
  - Predictive Alerts/ Actions
  - Auto Healing
  - Setting threshold on correlated metrics (multiple metrics)
Hands-on!
VNF Monitoring
Let’s play with metrics and (auto)dashboards!

- We will use slice created in previous session and stress the VDU of AGW VNF
Let’s play with metrics and (auto)dashboards!

Let’s play with metrics and (auto)dashboards!

• Login to AGW VM from management VM (172.21.248.4) command line.

  $ ssh magma@172.21.248.14
  Note- Password is same as the user name i.e. magma

• Increase CPU load with this command. Not down the process id.

  $ yes > /dev/null &

• Observe increase in CPU load and eventually a new VDU is created through auto scaling.
Let’s play with metrics and (auto)dashboards!

- Increase in load visible in Grafana
Let’s play with metrics and (auto)dashboards!

- Metrics collection from scaled out VDU is also visible after sometime.
Let’s play with metrics and (auto)dashboards!

• Check webhook invoked at [https://webhook.site/](https://webhook.site/) when alarm is generated.
Let’s play with metrics and (auto)dashboards!

• Now locate the IP of the process and kill it to reduce the extra CPU load

```bash
$ kill <process-id>
```

• Observe decrease in CPU load and eventually a additional VDU is deleted.