Monitoring and accounting in the INFN-Cloud infrastructure

Vincenzo Spinoso for the INFN Cloud project

EGI Conference 2020
INFN Cloud monitoring

- INFN Cloud is born on top of an existing cloud (*backbone*) that federates ReCaS-BARI and CNAF
- The backbone is an autonomous *tightly coupled federation*
- Other sites can join the INFN Cloud (*loosely coupled*)
- The monitoring solution chosen for the INFN Cloud monitoring is Zabbix
Why Zabbix

• Robust, supported, mature, reliable, widely used, well documented
• A bit more complex than Nagios (if you only need alarms)
• Less expressive than Grafana
• Zabbix provides APIs
  • Grafana can behave as a Zabbix «dashboard/frontend»
  • Monitoring can be easily developed using simple JSON/RPC calls
Backbone monitoring architecture

- A Zabbix proxy at each site collects the metrics reported at low level
- Proxies send the metrics to the central server, exposing the Zabbix frontend
- Notifications via mailing list and messaging systems (Microsoft Teams as of now)
- Host metrics (CPU, memory, disks, network...) and service metrics (CEPH, DNS, other basic services)
- Some other metrics in progress (i.e. backbone object storage like Swift or S3)
INFN Cloud monitoring architecture

- A separated instance of Zabbix exposed on the Internet
- Provides monitoring for
  - All the INFN Cloud PaaS «building blocks»
  - Federated clouds (i.e. VM cycle)
- The INFN Cloud status page shows the users the overall status of the federation, *at a glance* [*]
  - [https://status.cloud.infn.it/](https://status.cloud.infn.it/)

[*] basata su [https://staytus.co/](https://staytus.co/)
Monitoring di INFN Cloud

1. INFN Cloud

<table>
<thead>
<tr>
<th>Service</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object Storage</td>
<td>Operational</td>
</tr>
<tr>
<td>Backbone - Cloud Compute (Bari)</td>
<td>Operational</td>
</tr>
<tr>
<td>Backbone - Cloud Compute (CNAF)</td>
<td>Operational</td>
</tr>
<tr>
<td>Authentication</td>
<td>Operational</td>
</tr>
<tr>
<td>Cloud Compute</td>
<td>Operational</td>
</tr>
</tbody>
</table>

5. PaaS services

<table>
<thead>
<tr>
<th>Service</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Infrastructure Manager</td>
<td>Operational</td>
</tr>
<tr>
<td>Orchestrator</td>
<td>Operational</td>
</tr>
<tr>
<td>CPR</td>
<td>Operational</td>
</tr>
<tr>
<td>CMDB</td>
<td>Operational</td>
</tr>
<tr>
<td>Dashboard</td>
<td>Operational</td>
</tr>
</tbody>
</table>

https://status.cloud.infn.it/
Monitoring di INFN Cloud

2. Federated Cloud - CloudVeneto

CloudVeneto - Cloud Compute  
Operational

3. Federated Cloud - ReCaS-Bari

RECAS-BARI - Cloud Compute  
Operational

4. Federated Cloud - Cloud@CNAF

Cloud@CNAF - Cloud Compute  
Operational

Status of a cloud that federates to INFN Cloud ➔ *basis for the calculation of the Availability/Reliability of the federated site*

https://status.cloud.infn.it/
Monitoring the user level services

• When requested by the particular use case, by design monitoring is included in the deployment
  • Some use cases have it already
  • Working groups on Mesos,
Accounting

• Server based on RabbitMQ, APEL and Grafana
• Client basato su InfluxDB, collectd, cASO e APEL-SSM
• Status
  • Just tested at INFN-PADOVA (September ‘20), fully working
  • Installation in production happening now at backbone/CNAF
Accounting building blocks

Metrics (wallclocktime, cputime, disk usage, etc.) based on the EGI Cloud Accounting Usage Record

cASO (for OpenStack), oneacct (for OpenNebula)
APEL tools *ssmsend*, *ssmreceive* (based on the STOMP protocol)
APEL server
Accounting original testbed
Accounting improved architecture

- APEL Server
  - STOMP Message broker
  - ssmreceive
  - ssmrecv

- cASO "patched"
  - query

- InfluxDB
  - query
  - CPUPerf metrics

Openstack site
- NOVA
- Compute
- Compute

Missing networking information
Improving some metrics
Getting the CPUPerf via collectd

INFN Cloud
04/11/20
Collectd on the CloudVeneto testbed
Collectd on the CloudVeneto testbed
Plans and objectives

- Improve the **robustness** of the service
  - Zabbix HA implementation relying on frontend/server/DB replication
- Evaluate the Operational Level Agreement fulfilment with the federated clouds
  - Add and improve probes to allow testing of the federated clouds and their A/R evaluation
- Focus on embedding monitoring into the available deployments
  - All requiring use cases to bring monitoring into the deployment
  - Grafana to be used as common dashboard for the users, on top of Zabbix or Nagios (or stg else)