

D3.6.1	
Version	1.0
Author	USV
Dissemination	PU
Date	27/01/2014
Status	Final



D3.6.1: Management Tools V1

Project acronym:	NUBOMEDIA
Project title:	NUBOMEDIA: an elastic Platform as a Service (PaaS) cloud for interactive social multimedia
Project duration:	2014-02-01 to 2016-09-30
Project type:	STREP
Project reference:	610576
Project web page:	http://www.nubomedia.eu
Work package	WP3: NUBOMEDIA cloud platform
WP leader	Giuseppe Carella (TUB)
Deliverable nature:	Prototype
Lead editor:	Cristian Spoiala (USV)
Planned delivery date	01/2015
Actual delivery date	27/01/2015
Keywords	Virtual infrastructure, OpenStack, Management Console

The research leading to these results has been funded by the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 610576



FP7 ICT-2013.1.6. Connected and Social Media



This is a public deliverable that is provided to the community under a **Creative Commons Attribution-ShareAlike 4.0 International License**
<http://creativecommons.org/licenses/by-sa/4.0/>

You are free to:

- Share** — copy and redistribute the material in any medium or format
- Adapt** — remix, transform, and build upon the material
for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Notices:

You do not have to comply with the license for elements of the material in the public domain or where your use is permitted by an applicable exception or limitation.

No warranties are given. The license may not give you all of the permissions necessary for your intended use. For example, other rights such as publicity, privacy, or moral rights may limit how you use the material.

For a full description of the license legal terms, please refer to:
<http://creativecommons.org/licenses/by-sa/4.0/legalcode>

Contributors:

Cristian Spoiala (USV)
Alin Calinciu (USV)
Constantin Filote (USV)

Internal Reviewer(s):

Giuseppe Carella (TUB)
Lorenzo Tomasini (TUB)
Luis Lopez (URJC)

Version History

Version	Date	Authors	Comments
0.1	16/07/2014	Cristian Spoiala	Initial version
0.2	30/08/2014	Cristian Spoiala	Added conclusion to the monitoring tools analysis
0.3	22/12/2014	Cristian Spoiala	Added version history, current NUBOMEDIA console
1.0	07/01/2014	Cristian Spoiala	Added list of figures, removed SaltStack section, Installation section

Table of contents

1 Analysis of monitoring tools.....	8
1.1 OpenStack Horizon	8
1.1.1 <i>Daily Report</i>	8
1.1.2 <i>Stats</i>	9
1.1.3 <i>Conclusion</i>	9
1.2 Nagios	10
1.2.1 <i>Features</i>	10
1.2.2 <i>Dashboard</i>	11
1.2.3 <i>Architecture</i>	11
1.2.4 <i>Conclusion</i>	12
1.3 Icinga	12
1.3.1 <i>Dashboard</i>	12
1.3.2 <i>Architecture</i>	13
1.3.3 <i>Conclusion</i>	14
1.4 Graphite	14
1.4.1 <i>Architecture</i>	14
1.4.2 <i>Conclusion</i>	15
1.5 Zabbix	15
1.5.1 <i>Dashboard</i>	16
1.5.2 <i>Features</i>	16
1.5.3 <i>Architecture</i>	16
1.5.4 <i>Conclusion</i>	17
1.6 Ganglia.....	17
1.6.1 <i>Dashboard</i>	17
1.6.2 <i>Conclusion</i>	18
1.7 Zenoss	18
1.7.1 <i>Dashboard</i>	18
1.7.2 <i>Features</i>	18
1.7.3 <i>Architecture</i>	19
1.8 Conclusion	19
2 NUBOMEDIA Management Console	19
2.1 Why we choose Horizon	19
2.2 Architecture	20
2.3 Presentation.....	20
2.3.1 <i>Instance management</i>	20
2.3.2 <i>Images management</i>	21
2.3.3 <i>Security management</i>	22
2.3.4 <i>Monitoring</i>	23
3 Installation.....	23
3.1 Install OpenStack with RDO	23
3.1.1 <i>Software prerequisites:</i>	23
3.1.2 <i>Hardware prerequisites:</i>	23
3.1.3 <i>Operating system preparation</i>	24
3.1.4 <i>Install with Packstack</i>	24
3.2 Icinga Installation	25
3.2.1 <i>How to install</i>	25
References.....	26

List of Figures:

<i>Figure 1 Horizon Resource Usage Daily Report</i>	8
<i>Figure 2 Horizon Resource Usage Daily report</i>	9
<i>Figure 3 Horizon Resource Usage Stats</i>	9
<i>Figure 4 Nagios Service Status.....</i>	11
<i>Figure 5 Nagios Architecture.....</i>	12
<i>Figure 6 Icinga Dashboard</i>	13
<i>Figure 7 Icinga Services Dashboard</i>	13
<i>Figure 8 Icinga Architecture</i>	14
<i>Figure 9 Graphite Architecture</i>	15
<i>Figure 10 Zabbix Main Dashboard.....</i>	16
<i>Figure 11 Ganglia Chart for Wikimedia.....</i>	18
<i>Figure 12 Zenoss Dashboard.....</i>	18
<i>Figure 13 Horizon VM Management Dashboard</i>	20
<i>Figure 14 Actions from Horizon VM Management.....</i>	21
<i>Figure 15 Horizon Logs Interface.....</i>	21
<i>Figure 16 Horizon Images Management.....</i>	22
<i>Figure 17 Horizon Security Management.....</i>	22
<i>Figure 18 Icinga added to NUBOMEDIA Console.....</i>	25

Acronyms and abbreviations:

API	Application Programming Interface
SLA	Service-level Agreement
WSGI	Web Server Gateway Interface
GUI	Graphical User Interface

1 Analysis of monitoring tools

This section aims to give an overview of current open source monitoring tools that can be used for monitoring NUBOMEDIA instances.

1.1 OpenStack Horizon

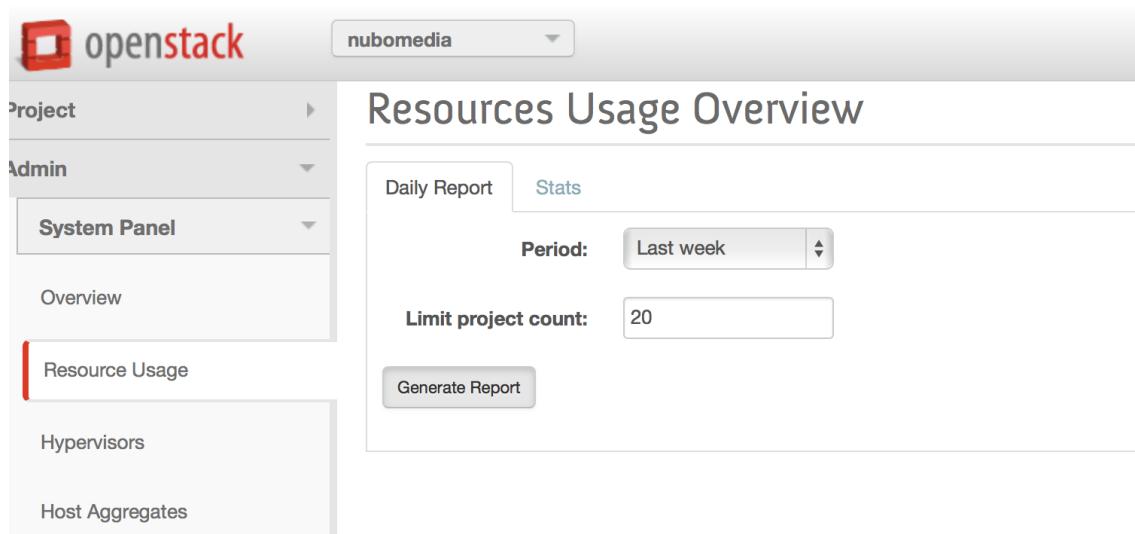
Horizon is a web-based dashboard for OpenStack and provides a web user interface for services like Nova, Keystone, Telemetry, etc.

Relevant to this section is the Resource Usage tab that can provide daily reports and stats.

1.1.1 Daily Report

Daily report tab generates reports for specific periods.

First select a period:



The screenshot shows the OpenStack Horizon dashboard with the "Resource Usage" tab selected. The main panel displays the "Resources Usage Overview" with a "Daily Report" tab selected. The "Period" dropdown is set to "Last week". A "Limit project count" input field contains the value "20". A "Generate Report" button is visible at the bottom of the form.

Figure 1 Horizon Resource Usage Daily Report

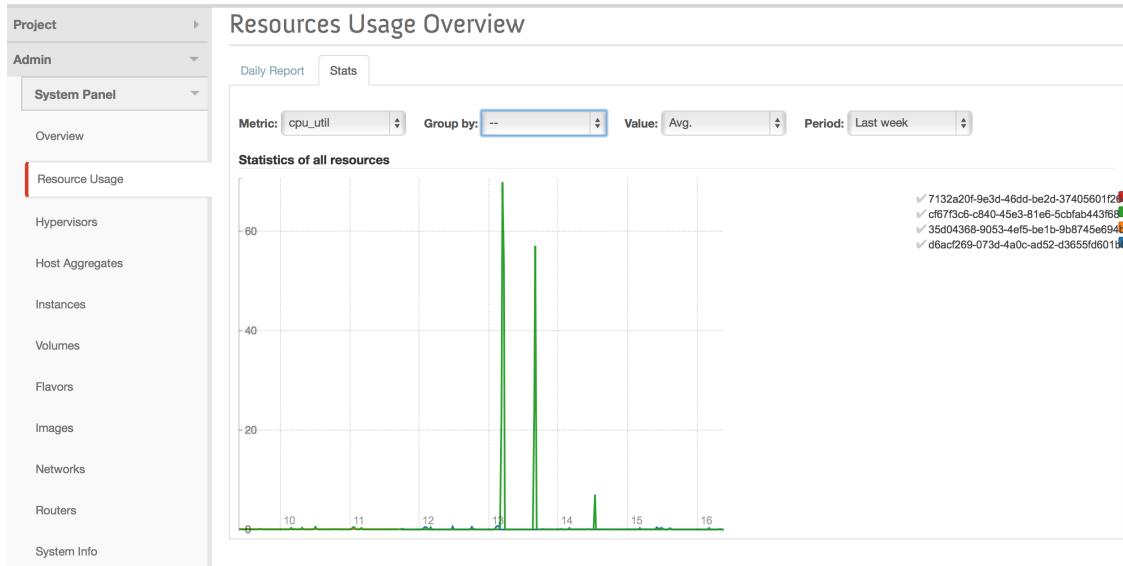
After selecting the period will be generated the report:

nubomedia

Service	Meter	Description	Day	Value (Avg)
Nova	disk.ephemeral.size	Size of ephemeral disk in GB	2014-07-11	0.0
Nova	disk.ephemeral.size	Size of ephemeral disk in GB	2014-07-12	0.0
Nova	disk.ephemeral.size	Size of ephemeral disk in GB	2014-07-16	0.0
Nova	disk.read.bytes	Volume of reads in B	2014-07-10	114061312.0
Nova	disk.read.bytes	Volume of reads in B	2014-07-11	114061312.0
Nova	disk.read.bytes	Volume of reads in B	2014-07-12	95878236.4444
Nova	disk.read.bytes	Volume of reads in B	2014-07-13	118950769.778
Nova	disk.read.bytes	Volume of reads in B	2014-07-14	121436672.0
Nova	disk.read.bytes	Volume of reads in B	2014-07-15	121436672.0
Nova	disk.read.bytes	Volume of reads in B	2014-07-16	17893869.844
Neutron	port	Duration of port	2014-07-12	1.0
Nova	disk.write.bytes	Volume of writes in B	2014-07-10	2123716807.11
Nova	disk.write.bytes	Volume of writes in B	2014-07-11	2130294670.22
Nova	disk.write.bytes	Volume of writes in B	2014-07-12	690342712.889
Nova	disk.write.bytes	Volume of writes in B	2014-07-13	275098168.889

Figure 2 Horizon Resource Usage Daily report**1.1.2 Stats**

Stats tab is displaying data from telemetry API for various metrics. Can be selected a metric and will be show a list of resources utilization for that metric.

**Figure 3 Horizon Resource Usage Stats****1.1.3 Conclusion**

Horizon seems good at displaying basic information about instances load, but detailed information could not be gathered. Also it lacks advanced filtering on graphs, reports availability, alerting system.

1.2 Nagios

Nagios is an open source monitoring and alerting tool for systems, network devices and services.

Is distributed under a GPL v2 license.

Nagios monitors your entire infrastructure to ensure systems, applications, services, and business processes are functioning properly. In the event of a failure, Nagios can alert technical staff of the problem, allowing them to begin remediation processes before outages affect business processes, end-users, or customers. Also can generate SLA reports for billing purposes.

1.2.1 Features

- Capabilities to monitor applications, services, operating systems, network protocols, system metrics and infrastructure components with a single tool. Powerful script APIs allow easy monitoring of in-house and custom applications, services, and systems
- Centralized view of entire monitored IT infrastructure. Detailed status information available through web interface
- Fast detection of infrastructure outages. Alerts can be delivered to technical staff via email or SMS. Escalation capabilities ensure alert notifications reach the right people
- Alert acknowledgments provide communication on known issues and problem response. Event handlers allow automatic restart of failed applications and services
- Trending and capacity planning addons ensure you're aware of aging infrastructure. Scheduled downtime allows for alert suppression during infrastructure upgrades
- Availability reports ensure SLAs are being met. Historical reports provide record of alerts, notifications, outages, and alert response. Third-party addons extend reporting capabilities
- Multi-user access to web interface allows stake holders to view infrastructure status. User-specific views ensures clients see only their infrastructure components
- Integration with in-house and third-party applications is easy with multiple APIs. Hundreds of community-developed addons extend core Nagios functionality
- Over 10 years of active development. Scales to monitor thousands of nodes. Failover capabilities ensure non-stop monitoring of critical IT infrastructure components.

1.2.2 Dashboard

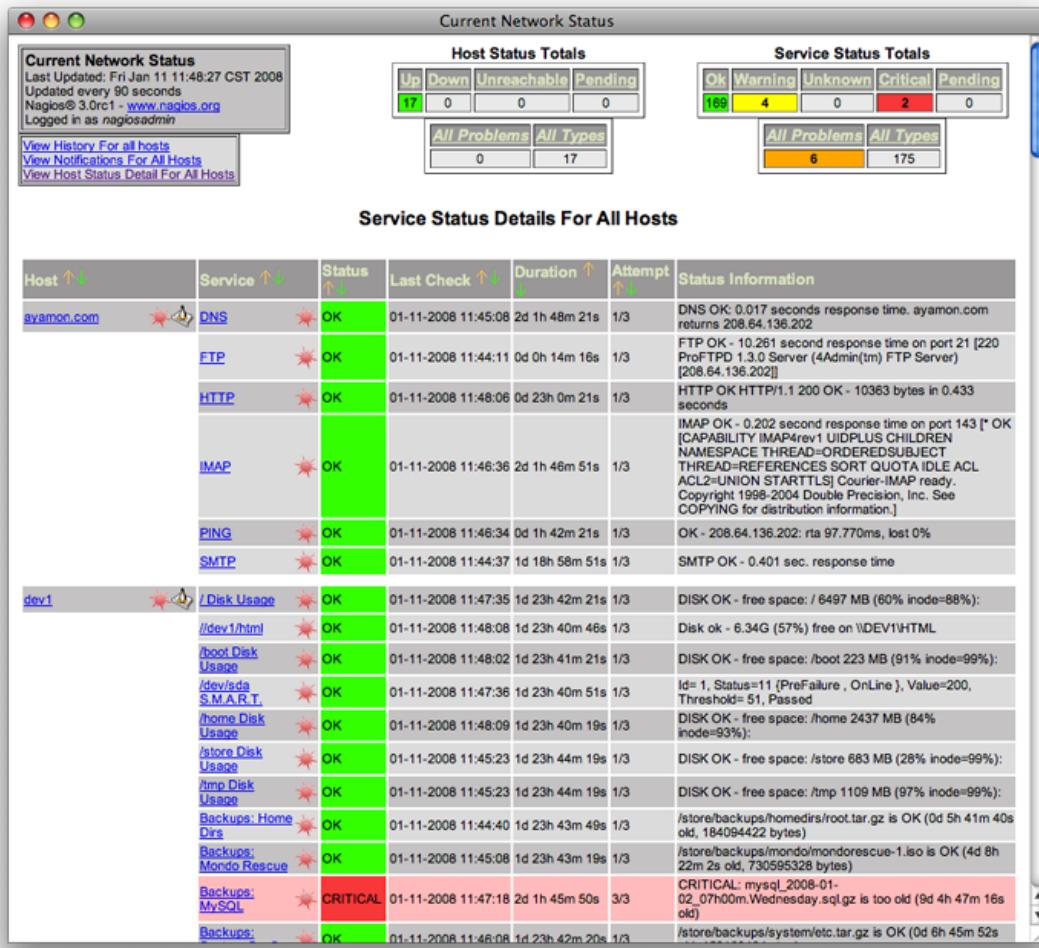


Figure 4 Nagios Service Status

1.2.3 Architecture

Using builtin plugins or addons Nagios can monitor all infrastructure.

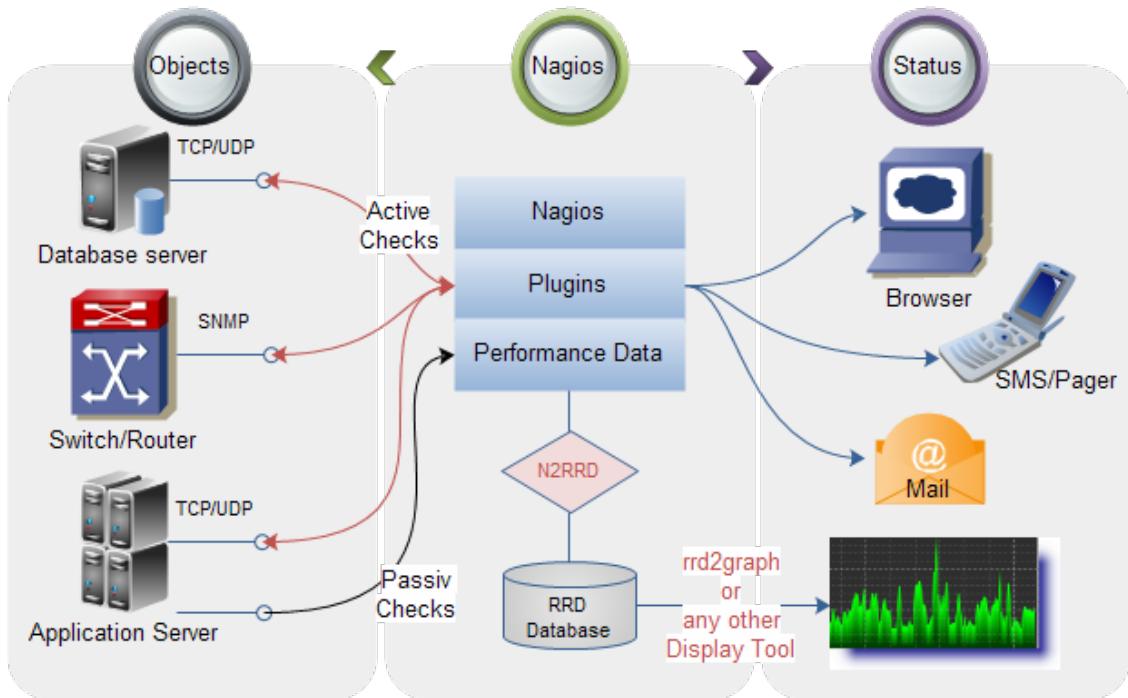


Figure 5 Nagios Architecture

1.2.4 Conclusion

Nagios has some issues:

- Progress for open source version is slow compared to the fork Icinga
- No built-in graphing system. Needs external plugins like Graphite, PNP4Nagios
- Configuration is static in flat files

But Nagios is a good solution with many useful features.

1.3 Icinga

Icinga is an open source network and computer system monitoring application. Was originated as a fork of Nagios in 2009 but maintaining the compatibility with Nagios plugins. Goals over Nagios were to add a new modern user interface, multiple database connectors and a REST API for developers. Additionally of Nagios contains improvements to SLA accuracy.

Is distributed under GPL v2 license.

1.3.1 Dashboard

D3.6.1: Management Tools V1



The screenshot shows the Icinga dashboard interface. On the left, a sidebar menu includes options like Dashboard, Problems, Overview, History, System, and Logout. The main area has tabs for 'Current incidents' and 'Landing page'. Under 'Service Problems', there is a list of critical issues:

- mysql-uptime on c2-mysql-1 (Unreachable)**: CRITICAL - connection could not be established within 60 seconds
- mysql-index-usage on c2-mysql-2 (Unreachable)**: CRITICAL - connection could not be established within 60 seconds
- mysql-connection-time on c2-mysql-1 (Unreachable)**: CRITICAL - connection could not be established within 60 seconds
- mysql-index-usage on c2-mysql-1 (Unreachable)**: CRITICAL - connection could not be established within 60 seconds
- mysql-open-files on c2-mysql-1 (Unreachable)**: CRITICAL - connection could not be established within 60 seconds
- mysql-tablecache-hitrate on c2-mysql-1 (Unreachable)**: CRITICAL - connection could not be established within 60 seconds
- mysql-connection-time on c2-mysql-2 (Unreachable)**: CRITICAL - connection could not be established within 60 seconds

Under 'Recently Recovered Services', a list of services is shown:

- win-mem physical on c1-active directory-3**: OK (4m 20s)
- win-mem physical on c1-windows-1**: OK (8m 55s)
- win-mem physical on c2-windows-1**: OK (8m 55s)
- win-mem physical on c1-exchange-1**: OK (8m 56s)
- win-mem physical on c1-active directory-1**: OK (8m 57s)
- win-mem physical on c1-active directory-2**: OK (9m 7s)
- win-cpu3 on c1-exchange-1**: OK (2d 6h)
- win-disk C: on c1-active directory-3**: OK (2d 6h)
- win-cpu on c1-active directory-3**: OK (2d 7h)
- win-upptime on c1-active directory-3**: OK (2d 7h)

Figure 6 Icinga Dashboard

The screenshot shows the Icinga Services Dashboard. The top navigation bar includes 'Monitoring', 'Admin', and 'Help'. The main area has tabs for 'Overview' and 'Downtimes'. The 'Downtimes' tab is active, displaying a table of scheduled downtime entries:

Host	Service	Service status	Scheduled start	Scheduled end	Author	Comment
web_de-pop	POP3	CRITICAL	2012-05-16 17:23:19	2012-05-16 19:25:19	guest	123
google-www	HTTP	CRITICAL	2012-03-07 16:42:57	2037-03-07 18:42:57	guest	it's doooooowwn
google-www	HTTP	CRITICAL	2012-04-03 14:57:36	2012-04-03 16:58:36	guest	xboksaoidhosa sd
google-www	HTTP	CRITICAL	2012-05-16 17:23:19	2012-05-16 19:25:19	guest	123
web_de-pop		UP	2012-04-02 11:55:49	2012-04-02 13:55:49	root	comment
web_de-pop		UP	2012-05-16 17:21:26	2012-05-16 17:25:26	guest	Test
web_de-pop		UP	2014-05-05 11:19:21	2014-05-05 13:19:21	guest	x
c2-printserver-1		UP	2012-04-02 11:57:02	2012-04-02 13:57:02	root	comment
grnx-pop	POP3	CRITICAL	2011-12-21 01:46:39	2011-12-21 03:49:39	guest	Die Kast eist Down, muss ...
grnx-pop	POP3	CRITICAL	2012-02-13 18:46:01	2012-02-13 20:46:01	guest	-
grnx-pop	POP3	CRITICAL	2012-02-29 21:40:39	2012-02-26 23:40:39	guest	fhghj
grnx-pop	POP3	CRITICAL	2012-03-22 15:53:38	2012-03-22 15:53:38	guest	test
grnx-pop	POP3	CRITICAL	2012-04-23 09:42:22	2012-05-23 00:00:00	guest	doin stuff and such
grnx-pop	POP3	CRITICAL	2012-05-16 17:23:19	2012-05-16 19:25:19	guest	123
web_de-www	HTTP	CRITICAL	2012-01-31 12:10:00	2012-01-31 13:00:00	Admin	Test

Figure 7 Icinga Services Dashboard

1.3.2 Architecture

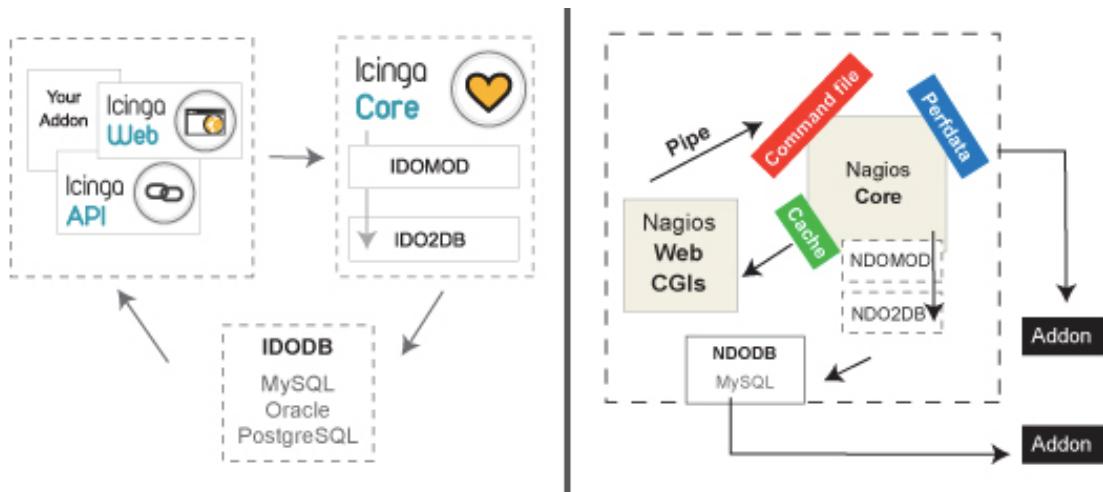


Figure 8 Icinga Architecture

1.3.3 Conclusion

Considering the improvements from Nagios, Icinga can be a better option than Nagios. Has the same shortcoming like Nagios for graphs but Icinga2 has a built-in connector for Graphite.

1.4 Graphite

Graphite is an open source project to store numeric time-series data and render graphs of this data. Graphite doesn't collect the information, is only storing it. To send data to Graphite can be used Nagios, collectd, StasD.

Graphite is distributed under an Apache 2.0 License.

1.4.1 Architecture

Graphite is developed in Python and uses Django framework for Webapp.

1.4.1.1 Components:

- Whisper – time series database. Is an alternative to RRD
- Webapp – a Python/Django web application that renders graphs
- Carbon – Daemon that waits for the data

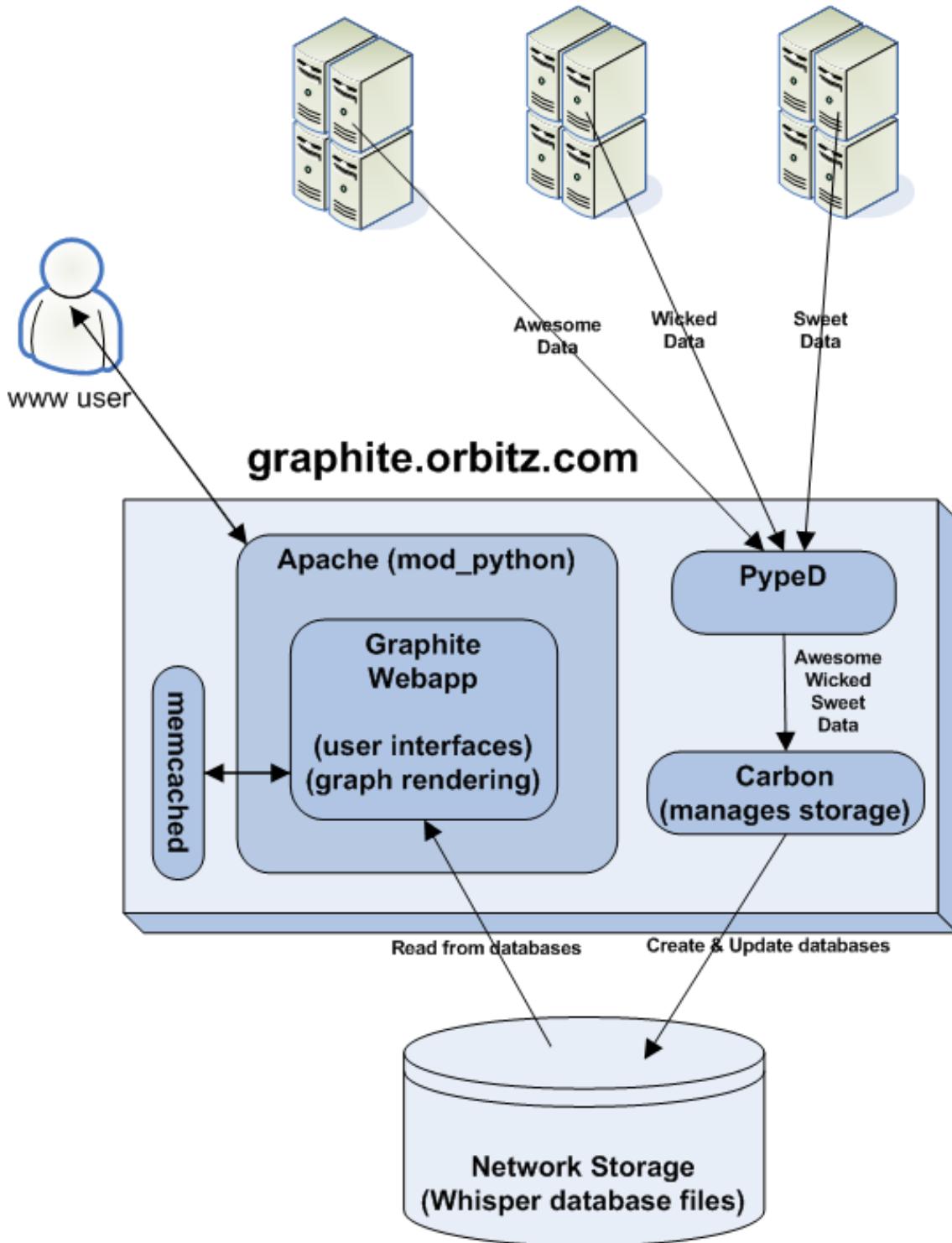


Figure 9 Graphite Architecture

1.4.2 Conclusion

Graphite is a great graph rendering tool but needs to be integrated with other systems like Nagios, collectd, StatsD to be useful.

1.5 Zabbix

Zabbix is an open source software for monitoring of networks, computer systems and services. Graphing is built-in. Can also monitor websites and web services.

Is distributed under a GPL v2 license.

1.5.1 Dashboard

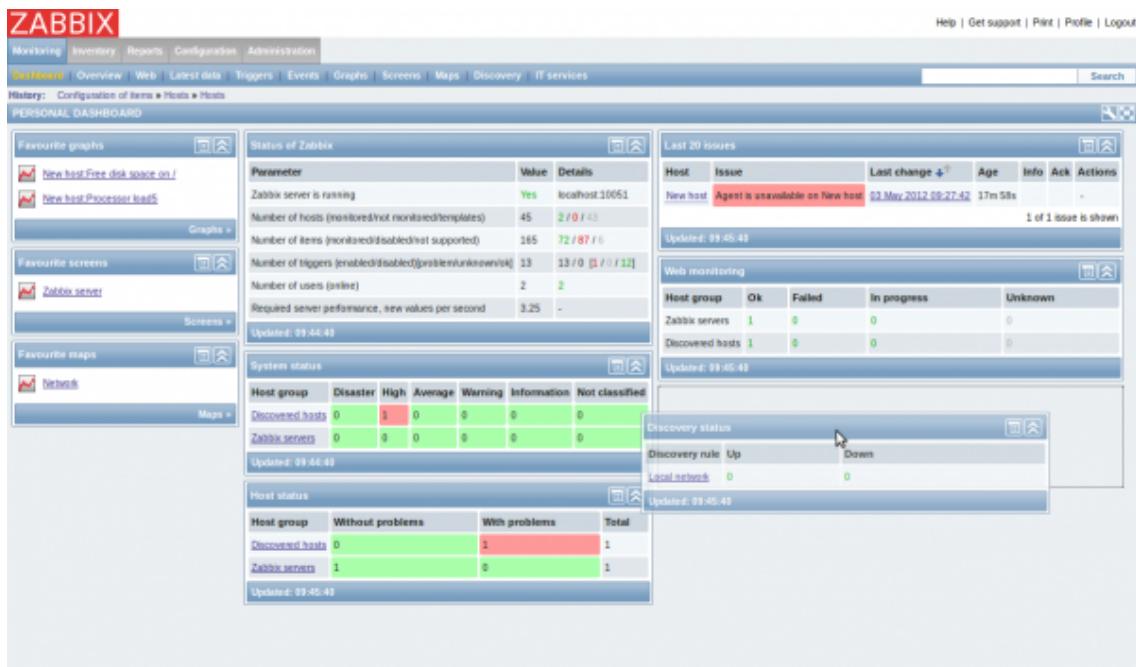


Figure 10 Zabbix Main Dashboard

1.5.2 Features

- Data gathering
- Flexible threshold definitions
- Highly configurable alerting
- Real-time graphing
- Web monitoring capabilities
- Extensive visualization options
- Historical data storage
- Easy configuration
- Use of templates
- Network discovery
- Fast web interface
- Zabbix API
- Permissions system
- Full featured and easily extensible agent
- Binary daemons
- Ready for complex environments

1.5.3 Architecture

Stores the configuration and monitoring data in a database.

The following are supported:

- MySQL

- PostgreSQL
- Oracle
- SQLite

1.5.3.1 Components

- Server
- Agents
- Frontend

Web frontend is developed in PHP, rest of the components are in C.

1.5.4 Conclusion

Zabbix is the most integrated solution with many features.

1.6 Ganglia

Ganglia is a scalable distributed monitoring system for high-performance computing systems such as clusters and Grids. It is based on a hierarchical design targeted at federations of clusters.

It leverages widely used technologies such as XML for data representation, XDR for compact, portable data transport, and RRDtool for data storage and visualization. It uses carefully engineered data structures and algorithms to achieve very low per-node overheads and high concurrency.

The implementation is robust, has been ported to an extensive set of operating systems and processor architectures, and is currently in use on thousands of clusters around the world. It has been used to link clusters across university campuses and around the world and can scale to handle clusters with 2000 nodes.

Ganglia is distributed under a BSD License.

1.6.1 Dashboard

D3.6.1: Management Tools V1

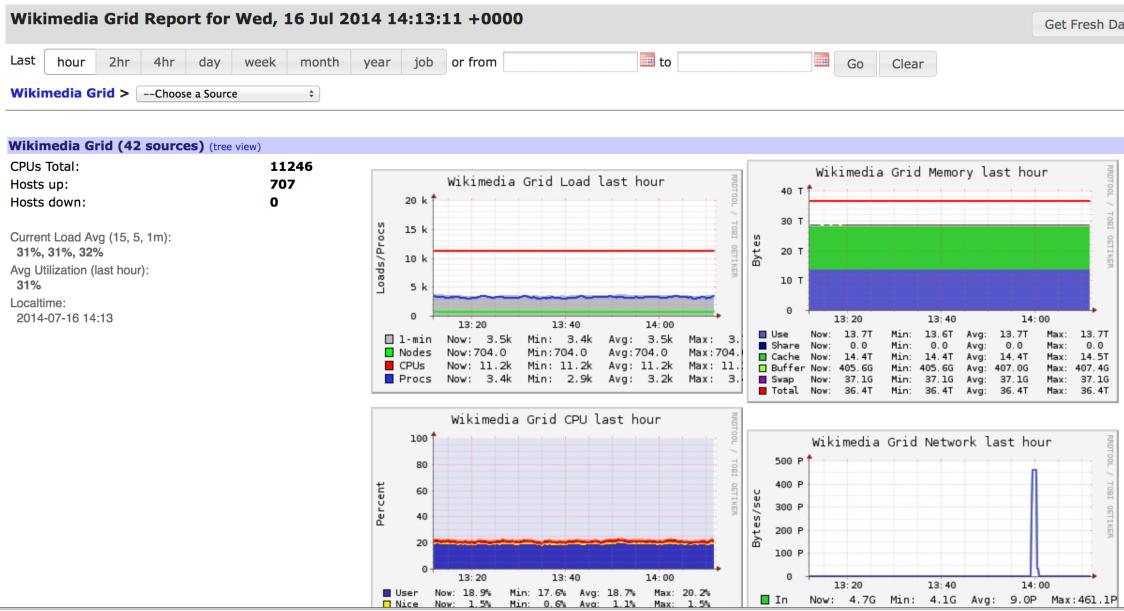


Figure 11 Ganglia Chart for Wikimedia

1.6.2 Conclusion

Main task of Ganglia is monitor large distributed systems and alerting, reports are not included.

1.7 Zenoss

Zenoss is an open-source software that allows to monitor availability, inventory/configuration, performance, and events.

Is distributed under a GPL v2 License.

1.7.1 Dashboard

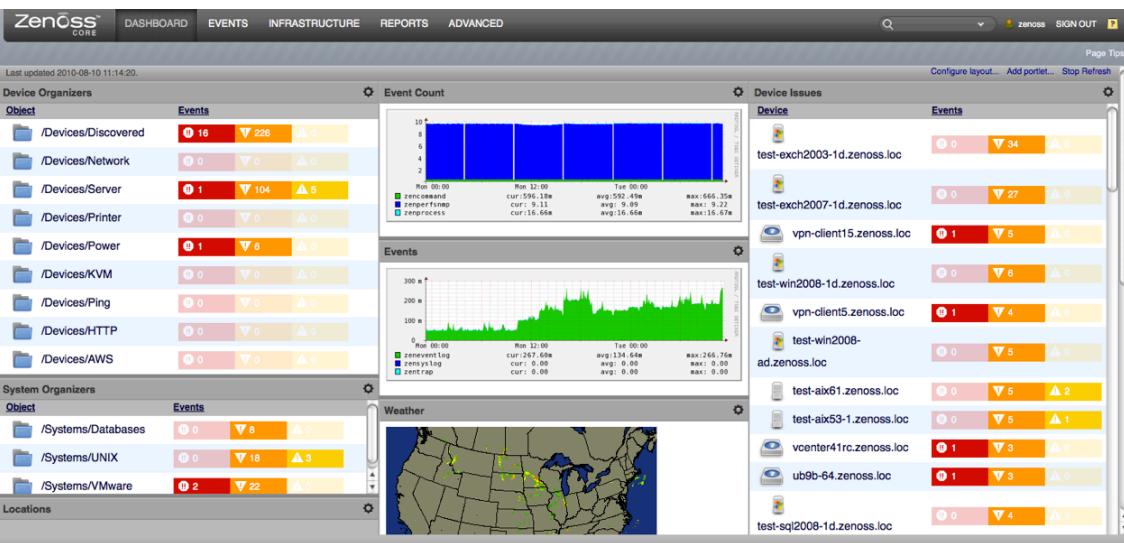


Figure 12 Zenoss Dashboard

1.7.2 Features

- Monitoring availability of network devices using SNMP, SSH, WMI
- Monitoring of network services (HTTP, POP3, NNTP, SNMP, FTP)

- Monitoring of host resources (processor, disk usage) on most network operating systems.
- Time-series performance monitoring of devices (based on RRDtool)
- Extended Microsoft Windows monitoring via Windows Management Instrumentation using SAMBA and Zenoss open source extensions
- Event management tools to annotate system alerts
- Automatically discovers network resources and changes in network configuration
- Alerting system provides notifications based on rule sets and on-call calendars
- Supports Nagios plug-in format

1.7.3 Architecture

Built with Python and Zope application server. Uses MySQL to store configuration.

1.8 Conclusion

For VM management we propose to use OpenStack management solution (Horizon). This will allow management of instances, networking configuration, virtual machines migration, disk management, images management and orchestration services.

For monitoring we propose to deploy and integrate Icinga in Horizon. Icinga will add following capabilities to the management console:

- Monitoring of host machines nodes and virtual machines instances
- Alerts
- Reports.

Data for alerts will be pushed to Icinga from Ceilometer when appropriate and Icinga agents.

For graphing system, Graphite will be integrated in Icinga with data provided by Ceilometer and Icinga agents.

2 NUBOMEDIA Management Console

For management console we choose to use OpenStack Dashboard. Horizon will be the core platform for further extensions with other open source tools to add more capabilities.

Horizon is covering basic capabilities for managing virtual infrastructure but lacks advanced graphing system and web alerting tools. These capabilities are needed by a NUBOMEDIA instance provider to have a better understanding of the platform usage and early alerts of performance degradation or unavailability.

2.1 Why we choose Horizon

From our analysis of various monitoring tools, Horizon from OpenStack had already most of functionality that we needed in managing the virtual servers.

2.2 Architecture

Horizon is built on Python and Django Framework and provides access to OpenStack services.

On NUBOMEDIA is running on Apache with WSGI module. Currently is installed on the controller machine on both development and production instance.

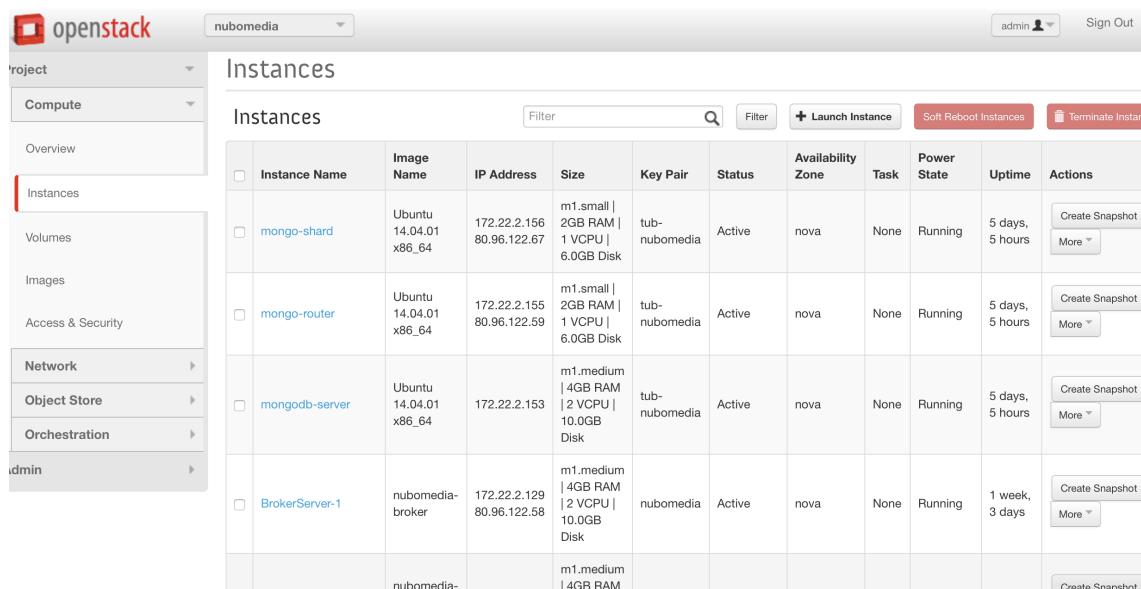
Frontend is built with: Bootstrap, jQuery, AngularJS, LESS CSS.

2.3 Presentation

In the following will be presented current NUBOMEDIA management console.

2.3.1 Instance management

From instances are managed all states of a virtual machine. From the GUI is possible to create a server from an existing image or from scratch. Also is possible to edit the virtual machine by changing network and security settings. If a server is not needed anymore can be also deleted from the GUI.



The screenshot shows the Horizon VM Management Dashboard. The left sidebar has a 'Compute' section with 'Instances' selected. The main area is titled 'Instances' and lists four servers:

Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Uptime	Actions
mongo-shard	Ubuntu 14.04.01 x86_64	172.22.2.156 80.96.122.67	m1.small 2GB RAM 1 VCPU 6.0GB Disk	tub-nubimedia	Active	nova	None	Running	5 days, 5 hours	<button>Create Snapshot</button> <button>More</button>
mongo-router	Ubuntu 14.04.01 x86_64	172.22.2.155 80.96.122.59	m1.small 2GB RAM 1 VCPU 6.0GB Disk	tub-nubimedia	Active	nova	None	Running	5 days, 5 hours	<button>Create Snapshot</button> <button>More</button>
mongodb-server	Ubuntu 14.04.01 x86_64	172.22.2.153	m1.medium 4GB RAM 2 VCPU 10.0GB Disk	tub-nubimedia	Active	nova	None	Running	5 days, 5 hours	<button>Create Snapshot</button> <button>More</button>
BrokerServer-1	nubimedia-broker	172.22.2.129 80.96.122.58	m1.medium 4GB RAM 2 VCPU 10.0GB Disk	nubimedia	Active	nova	None	Running	1 week, 3 days	<button>Create Snapshot</button> <button>More</button>
	nubimedia-		m1.medium 4GB RAM							<button>Create Snapshot</button>

Figure 13 Horizon VM Management Dashboard

D3.6.1: Management Tools V1



<input type="checkbox"/>	mongo-router	Ubuntu 14.04.01 x86_64	172.22.2.155 80.96.122.59	m1.small 2GB RAM 1 VCPU 6.0GB Disk	tub-nubimedia	Active	nova	None	Running	5 days, 5 hours	Create Snapshot	More ▾
<input type="checkbox"/>	mongodb-server	Ubuntu 14.04.01 x86_64	172.22.2.153	m1.medium 4GB RAM 2 VCPU 10.0GB Disk	tub-nubimedia	Active	nova	None	Running	5 days, 5 hours	Associate Floating IP	Disassociate Floating IP
<input type="checkbox"/>	BrokerServer-1	nubimedia-broker	172.22.2.129 80.96.122.58	m1.medium 4GB RAM 2 VCPU 10.0GB Disk	nubimedia	Active	nova	None	Running	1 week, 3 days	Edit Instance	Edit Security Groups
<input type="checkbox"/>	MediaServer-1	nubimedia-media-server	172.22.2.127 80.96.122.69	m1.medium 4GB RAM 2 VCPU 10.0GB Disk	nubimedia	Active	nova	None	Running	1 week, 3 days	Console	View Log
<input type="checkbox"/>	ControllerServer-1	nubimedia-control-	172.22.2.128 80.96.122.68	m1.medium 4GB RAM 2 VCPU	nubimedia	Active	nova	None	Running	1 week, 3 days	Pause Instance	Suspend Instance

Figure 14 Actions from Horizon VM Management

Logs from a server can be accessed remotely from the GUI:

The screenshot shows the Horizon interface for managing instances. On the left, there's a sidebar with navigation links like Project, Compute, Overview, Instances, Volumes, Images, Access & Security, Network, Object Store, Orchestration, and Admin. The main area is titled "Instance Details: mongo-shard". Below the title, there are tabs for Overview, Log, and Console. The Log tab is active, showing the "Instance Console Log". The log content is as follows:

```
* Starting automatic crash report generation[74G[ OK ]
* Starting OpenSSH server[74G[ OK ]
open-vm-tools: not starting as this is not a VMware VM
landscape-client is not configured, please run landscape-config.
* Restoring resolver state... [80G[74G[ OK ]
* Stopping System V runlevel compatibility[74G[ OK ]
* Starting Seed the pseudo random number generator on first boot[74G[31mfail[39;49m]
Ubuntu 14.04.1 LTS mongo-shard tty50

mongo-shard login: Cloud-init v. 0.7.5 running 'modules:config' at Wed, 17 Dec 2014 10:05:39 +0000. Up 31.60 seconds.
Generating locales...
  en_US.UTF-8... up-to-date
Generation complete.
Cloud-init v. 0.7.5 running 'modules:final' at Wed, 17 Dec 2014 10:05:47 +0000. Up 40.09 seconds.
ci-info: =====Authorized keys from /home/ubuntu/.ssh/authorized_keys for user ubuntu=====
```

Figure 15 Horizon Logs Interface

2.3.2 Images management

From GUI of images can be managed an image. The following actions can be performed:

- Create an image
- Delete an image
- Launch a server from an image
- Edit name, description and security settings for an image

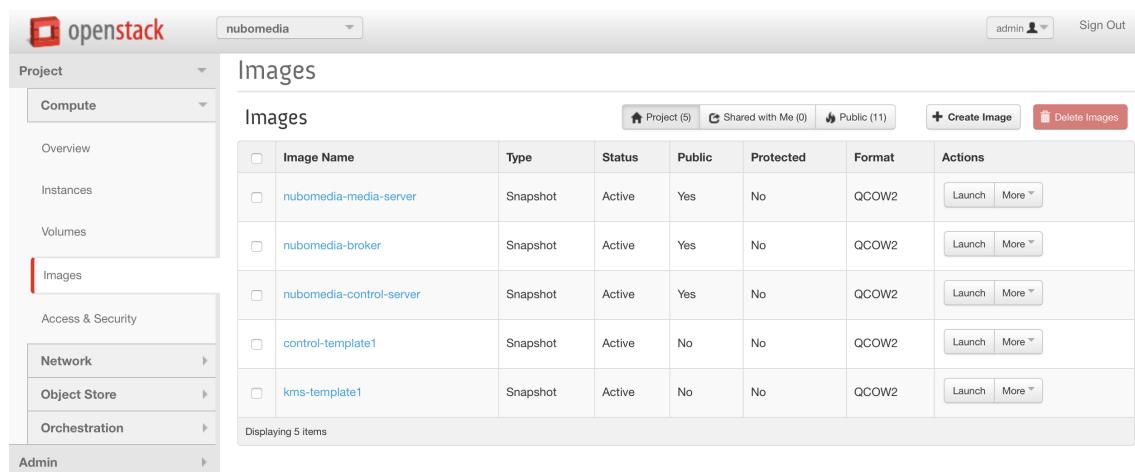


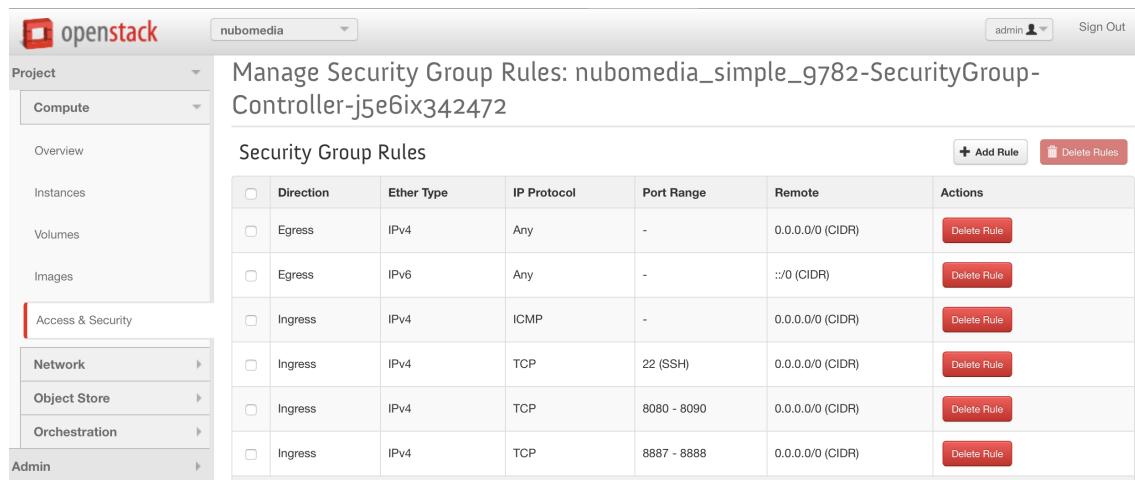
Image Name	Type	Status	Public	Protected	Format	Actions
nubomedia-media-server	Snapshot	Active	Yes	No	QCOW2	<button>Launch</button> <button>More</button>
nubomedia-broker	Snapshot	Active	Yes	No	QCOW2	<button>Launch</button> <button>More</button>
nubomedia-control-server	Snapshot	Active	Yes	No	QCOW2	<button>Launch</button> <button>More</button>
control-template1	Snapshot	Active	No	No	QCOW2	<button>Launch</button> <button>More</button>
kms-template1	Snapshot	Active	No	No	QCOW2	<button>Launch</button> <button>More</button>

Figure 16 Horizon Images Management

2.3.3 Security management

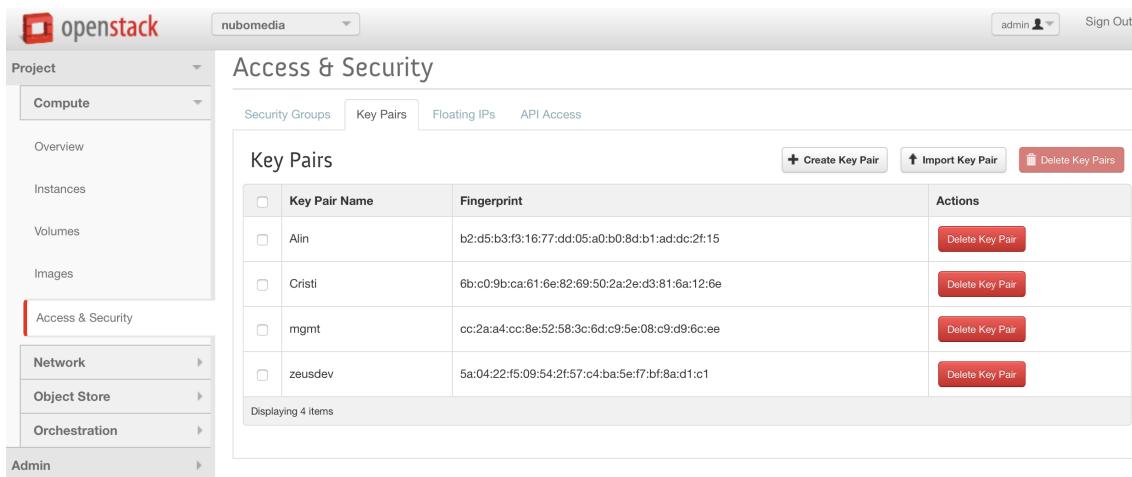
Security for each server can be configured from the Access & Security Tab. From the GUI can be added multiple ssh keys for the servers or managed the security rules for networking.

Networking rules can be grouped as Security Groups for easier management.



Direction	Ether Type	IP Protocol	Port Range	Remote	Actions
Egress	IPv4	Any	-	0.0.0.0/0 (CIDR)	<button>Delete Rule</button>
Egress	IPv6	Any	-	::/0 (CIDR)	<button>Delete Rule</button>
Ingress	IPv4	ICMP	-	0.0.0.0/0 (CIDR)	<button>Delete Rule</button>
Ingress	IPv4	TCP	22 (SSH)	0.0.0.0/0 (CIDR)	<button>Delete Rule</button>
Ingress	IPv4	TCP	8080 - 8090	0.0.0.0/0 (CIDR)	<button>Delete Rule</button>
Ingress	IPv4	TCP	8887 - 8888	0.0.0.0/0 (CIDR)	<button>Delete Rule</button>

Figure 17 Horizon Security Management



<input type="checkbox"/> Key Pair Name	Fingerprint	Actions
<input type="checkbox"/> Alin	b2:d5:b3:f3:16:77:dd:05:a0:b0:8d:b1:ad:dc:2f:15	<input type="button" value="Delete Key Pair"/>
<input type="checkbox"/> Cristi	6b:c0:9b:ca:61:6e:82:69:50:2a:2e:d3:81:6a:12:6e	<input type="button" value="Delete Key Pair"/>
<input type="checkbox"/> mgmt	cc:2a:a4:cc:8e:52:58:3c:6d:c9:5e:08:c9:d9:6c:ee	<input type="button" value="Delete Key Pair"/>
<input type="checkbox"/> zeusdev	5a:04:22:f5:09:54:2f:57:c4:ba:5e:f7:bf:8a:d1:c1	<input type="button" value="Delete Key Pair"/>

Displaying 4 items

2.3.4 Monitoring

3 Installation

This section will cover the installation of OpenStack, NUBOMEDIA software

3.1 Install OpenStack with RDO

For the v1 of the autonomous installer, the initial setup process is manual and will be configured automatically in v2 with a configuration management tool.

Deploying RDO is an easy process, setting up an OpenStack cloud takes approximately 15 minutes. It can be as short as 3 steps if you want to deploy it on a single server, but if you want to deploy it to add more nodes it can take more time.

RDO is maintained by RedHat, and it is more suitable for production environments.

The deployment script by RDO is licensed under the [Creative Commons Attribution-ShareAlike 3.0 Unported license](#).

3.1.1 Software prerequisites:

For installing OpenStack RDO, you will need a RHEL-based Linux distribution, such as CentOS, Scientific Linux, or Fedora 20 or later.

3.1.2 Hardware prerequisites:

It is recommended a machine with at least 2GB of RAM, and hardware virtualization extension with at least 1 network adapter for single node deployment. For multi-node deployment, at least two network adapters are needed.

For multi-node deployment, you will also need a Layer 2 Switch that supports 802.11Q VLANs (VLAN tagging).

3.1.3 Operating system preparation

- You will first need to add RDO repositories:

```
yum install -y http://rdo.fedorapeople.org/rdo-release.rpm
```

- You will need to update your current packages using:

```
yum update -y
```

- Then you need to enable ssh key login:

```
cd ~
mkdir .ssh
chmod 700 .ssh
cd .ssh
nano -w authorized_keys # here you should add your
public key
chmod 600 authorized_keys
restorecon -R -v /root/.ssh
```

- Disable selinux or set it in permissive mode (if there is a reason not to have it in enforcing mode).

In file: /etc/selinux/config edit:

```
SELINUX=permissive
```

After this if you do not want to reboot the system you should:

```
setenforce 0
```

If you have previously disabled SELinux, you will need to re-label the filesystem, since when SELinux is disabled, this does not happen for new files, and failing to relabel will likely cause many false positive issues. The easiest way to do that is to do the following as root:

```
touch /.autorelabel
reboot
```

- After this you should install NTP client on all servers because all servers should have date in sync with each other:

```
yum install ntp -y
chkconfig ntpd on
ntpdate pool.ntp.org
/etc/init.d/ntp start
```

3.1.4 Install with Packstack

- You should first generate the configuration file for the deployment with the following command:

```
packstack --gen-answer-file=icehouse_deployment_vlan.cfg
```

- After this, you should configure the file accordingly with your hardware configuration and also configure the deployment location for every service, if you use multi-node deployment.

```
nano icehouse_deployment_vlan.cfg
```

- Next, you should run packstack to deploy OpenStack RDO to all instances configured:

```
packstack --answer-file=icehouse_deployment_vlan.cfg
```

During this process, you will be required to type the root password for all node that you use in your deployment in order for OpenStack to be able to add its public key to each one of them.

Once the process is complete, you can log in to the OpenStack web interface "Horizon" by going to [http://\\$YOURIP/dashboard](http://$YOURIP/dashboard).

The username is "admin". The password can be found in the file `keystonerc_admin` in the `/root/` directory of the control node.

3.2 Icinga Installation

Icinga was installed using a Chef recipe. Chef is a configuration management tool that helps for automatic installation and configuration of applications and servers.

Source code of Chef recipe can be found on NUBOMEDIA git repository:

- <http://git.nubomedia.eu/usv/chef-icinga>

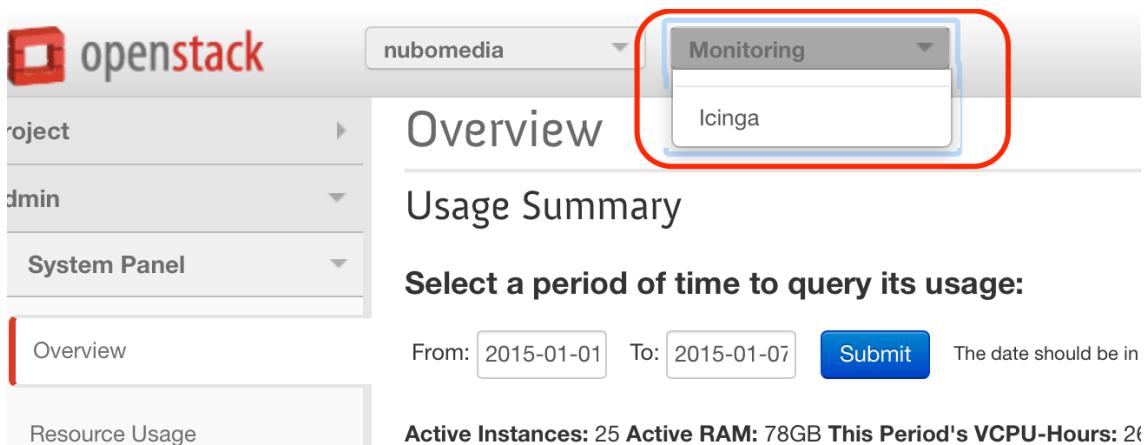


Figure 18 Icinga added to NUBOMEDIA Console

3.2.1 How to install

- Install Chef

```
curl -L https://www.opscode.com/chef/install.sh | bash
```

- Download the chef icinga recipe

<http://git.nubomedia.eu/usv/chef-icinga/repository/archive.tar.gz>

- Run installation

`chef-solo -c solo.rb -j web.json`

References

- [1]. Openstack: Open source software for creating public and private clouds. See <http://www.openstack.org/>.
- [2]. IT Infrastructure Monitoring <http://www.nagios.org>
- [3]. Icinga monitoring tool <https://www.icinga.org>
- [4]. Graphite - Scalable Realtime Graphing <http://graphite.wikidot.com>
- [5]. Zabbix is an enterprise-class open source distributed monitoring solution for networks and applications <http://www.zabbix.com>
- [6]. Ganglia is a scalable distributed monitoring system for high-performance computing systems such as clusters and Grids <http://ganglia.info>
- [7]. Zenoss IT monitoring <http://www.zenoss.com>
- [8]. Saltstack - Infrastructure automation and management system <http://www.saltstack.com>