

Analysis of Network Management and Monitoring using Cloud Computing

George Suciu^{1,2}, Victor Suciu², Razvan Gheorghe¹, Ciprian Dobre¹, Florin Pop¹, Aniello Castiglione³

¹Faculty of Automatic Control and Computers, Computer Science Department, University Politehnica of Bucharest, Romania

²R&D Department, BEIA Consult International, Bucharest, Romania

³Department of Computer Science, University of Salerno, Italy

Emails: george@beia.ro, victor.suciu@beia.ro, razvan.gheorghe@beia.ro, ciprian.dobre@cs.pub.ro, florin.pop@cs.pub.ro, castiglione@ieee.org

Abstract. In the near future the number of equipment connected to the Internet will greatly increase, so that further development of applications meant to verify their operations will be required. Monitoring represents an important factor in improving the quality of the services provided in cloud computing, given the fact that it allows scaling resource utilization in an adaptive manner. This paper aims to provide a solution for the monitoring of network devices and services, allowing administrator to verify connectivity of the equipment, their performances and network security. The main contribution of the paper consists in proposing an integrated solution that is deployed in the cloud for monitoring all the network components. Finally, the paper discusses the main findings and advantages for a reference implementation of the monitoring system using a simulated network.

Keywords: Network Monitoring, Cloud Computing, Nagios, Network Management.

1. Introduction

It is considered that in the near future the number of equipment which will be connected to the Internet will greatly increase, so that further development of applications meant to verify their operations will be required. Furthermore, it is expected that the number of 50 billion devices connected to the Internet will be reached in 2020, compared to 15 billion devices in 2015 [1].

Because networks of large operators are vast, troubleshooting the various problems that may arise can take a long time if a centralized solution is not used. A simple problem like the downtime of a web site can have multiple possible explanations: a faulty router, an out of service firewall or simply the server hosting the web site being down. All these are possible causes of a rather simple problem, but if the network is extensive, solving these problems can take even a few hours if the explanations presented above are taken one at a time.

Cloud Computing represents a relatively new concept that refers to an integrated service offered as a whole application, which offers access to information and data storage without the user having to know the physical location and configuration of the systems providing these services. Cloud computing is a general term for anything that involves delivering services on the Internet. It can be divided into three categories namely, Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS) in terms of classification by mode of delivery.

One of the main challenges associated to cloud computing is the resource monitoring, due to the lack of information and control regarding the customization of the parameters which describe the system. The current monitoring solutions are not entirely accurate in the cloud computing systems, given the fact that usually the resources are virtualized [2][15].

We can thereby define the monitoring of a centralized network as being a system built to monitor the performances of a network at any time and also to notify the network administrator about arising problems.

The ideal monitoring system must meet lots of requirements, the most important being:

- The integration of all the network components into the monitoring system that is deployed in the cloud. Thus, essential information can be gathered from all the network entities, which leads to an accurate understanding of the network situation. Furthermore, problem solving is accelerated.
- Increasing productivity: a company will be more productive if the network problems are solved quickly. When the network is working poorly, a quality monitoring system will quickly detect the problem and alert the IT department of the company. Consequently, data loss in a network is prevented, which will determine the productivity to decrease.
- Efficiency: an ideal monitoring system produces a decrease in the network maintenance cost. Instead of multiple monitoring departments, each one supervising an area, it can all be reduced to the employment of a single system administrator who can monitor the network from a centralized location. Thus, the hours spent troubleshooting can become more productive.

The rest of the paper is organized as follows: Section 2 details some monitoring applications, Section 3 presents the components of the monitoring system and Section 4 presents the monitoring system implementation. Section 5 presents the results of simulations and the concluding section summarizes the contributions of this paper.

2. Related Work

In this section, we analyze related work and main monitoring applications used nowadays for networks. On today's communications market, competition in monitoring applications is big, thus resulting in many companies dealing with their implementation. Among the most important applications are: Cacti, Zabbix, SCOM (System Center Operations Manager) 2012 and Wireshark.

We will further detail each approach, specifying the advantages and disadvantages.

2.1 Cacti

Cacti [3] is an open source monitoring application (free for users) based on a web server, being in fact a frontend for the standard monitoring technology RRDtool (Round Robin Database tool) [4]. Moreover, Cacti allows an easier use for inexperienced users of RRDTool.

Operation of Cacti can be reduced to three defined steps:

- Data processing: data is processed using a pooling system of the equipment connected to Cacti. More accurately, the network administrator can determine the status of a network equipment in real time, using the SNMP (Simple Network Management Protocol) protocol. Any equipment that has SNMP configured can be queried by Cacti.
- Data storage: data is stored using RRDtool. RRDtool is a data base which gathers information from the monitored network elements and then stores it efficiently and displays it as graphs.
- Presentation of data: at this point, data is processed in graphics and then presented to the network administrator.

2.2 Zabbix

Zabbix [5] is an open source monitoring application created in 2001 by Alexei Vladishev and it is used to monitor many network parameters, as well as the integrity of the servers. This uses a flexible notification mechanism which allows the administrator to receive an e-mail for every incident. The application supports both the pooling method outlined above and the trapping part (receiving data from various equipment without a prior request) [6].

In terms of software, the structure of the application is as follows:

- The Zabbix server, to which the agents report the information from various network equipment.
- The data base, where data from agents are stored.
- The web interface, where information about the network status can be accessed.
- The agent, which is installed on the equipment that is intended to be monitored. It sends the results to the main server.

2.3 SCOM 2012

System Center Operations Manager 2012 edition [7] represents the managing platform for the operating systems which features a single interface that can display data such as connectivity of the equipment, their performances, network security. It can also be used for computers that use Windows operating system and Linux. The application is used as client-server, an agent who collects and forwards data to the central SCOM server being installed on each equipment that is intended to be monitored. Later on, it can send notifications, depending on the event severity. For every type of monitored equipment (data base, Linux server, Apache web server)

specific management packs are defined. For these management packs, filtering rules for monitored information are established, thus providing great flexibility.

2.4 Wireshark

Wireshark [8] is an application that analyses packets traversing the network, trying to display them to the user in a detailed and easy manner. It works as a measuring device, examining what happens at the protocol network level in order to be able to create graphs using the obtained results. The main advantage is the detailed manner in which it verifies the packets and, on the other hand, the main disadvantage is that it does not have a notification system like other monitoring applications. It is used especially for debugging.

3 Components of the monitoring system

In this section we present the components and protocols that enable the cloud based operation for a network management and monitoring system. Furthermore, we present the components of a simulated network. Similar approach was presented in [16].

A. OpenStack

OpenStack [9] is a free open source cloud computing platform. The concept of open source describes how to produce or develop certain finished products, to allow users to engage in production or development. OpenStack is developed in the first stage model IaaS. The technology of this model consists in series of interrelated projects with, which activities are coordinated and monitored the processing, storage and network resources on activities. These activities are passed through a data center.

The main target of the open source project is to make OpenStack Cloud implementing a simple solution with a broad set of features. From this point of view, the IaaS model, OpenStack can provide many types of services such as basic services, warehousing services (Storage) sharing services (Shared Services) and service-level (High Level Services).

Furthermore, OpenStack provides an IaaS solution through a variety of complementary services, each service offering an API that helps the integration with other platforms and services.

On top of the OpenStack components there are deployed several services such as the Image one [10], which enables users to discover, register, and retrieve the virtual machine images.

B. Simulated network

The main components of the simulated network are presented in Fig. 1 and described below.

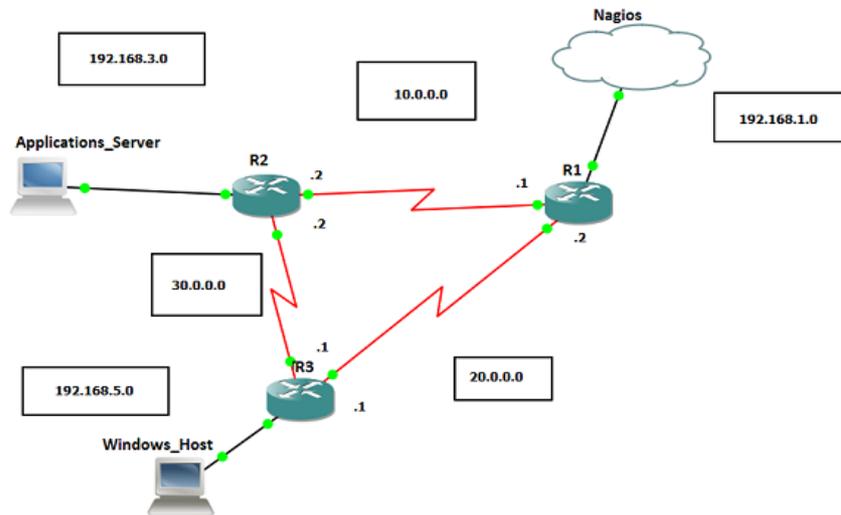


Fig. 1. The main components of the simulated network.

The main components of the simulated network are as follows:

- **Applications_Server:** A virtual machine having the Ubuntu 14.04 operating system, on which are installed HTTP (*Hypertext Transfer Protocol*), FTP (*File Transfer Protocol*), SMTP (*Simple Mail Transfer Protocol*) servers and also the monitoring agent used by Nagios Core, NRPE [11].
- **Windows_Host:** A virtual machine on which Windows 7 is installed as an operating system and NsClient++, used by Nagios Core, as a monitoring.
- **Nagios:** A virtual machine having the Ubuntu 14.04 operating system, on which the central server of Nagios Core is installed. Nagios Core is an open source version created by Nagios, representing now a standard for monitoring. Nagios monitors permanently the network equipment to verify the precision of their operation. The monitoring system used by Nagios serves two main components: hardware and software. The hardware component represents the physical equipment of the network: computers, printers, routers, servers, etc.. The software component represents the processes run by the physical network equipment, such as: supported web sites, server applications, etc.. By making this physical delimitation, Nagios manages to quickly identify network issues.
- **R1, R2 and R3 routers** are responsible with traffic routing between equipment from different networks. The routing protocol used is RIP (*Routing Information Protocol*).

C. Used protocols

In order to facilitate users' access to information on the Internet on World Wide Web (WWW) servers, HTTP (*Hypertext Transfer Protocol*) protocol is being used,

where response time is the main parameter monitored [12]. Next, we monitored the FTP [13] protocol that is a client-server protocol which facilitates file transfer between two devices using TCP/IP. Finally, we monitored the SMTP, a client-server, application level protocol used to transfer email messages between two network devices [14].

4 Monitoring system implementation

In order to monitor all the network services, we need to define a central monitoring instance on which the central server functions. The virtual machine on which the central server operates uses Ubuntu 14.04 operating system. On the operating system Nagios Core 3.0 application is installed. The application is installed as precompiled packages located at `/etc/nagios3` and afterwards the “plugin” modules will be installed separately.

Its role is monitoring the network devices (Applications_Server, R1, R2, R3 and Windows_Host) through Nagios Core 3.0 application. As stated earlier, each one of these devices is monitored using a different protocol: NRPE for Applications_Server, NSClient++ for Windows_Host and SNMP for the three routers.

To achieve this, the server must have “plugin” modules for the required verifications.

In addition to “plugin” modules, the server needs a main configuration file, located at `/etc/nagios3/`, named `nagios.cfg`. Within it are defined attributes, including the location where the application will write its logs, the location of the secondary configuration files for each new monitored equipment, the username used for GUI authentication and the interval between verifications.

After setting up all the aspects of connectivity between equipment and creating statements for each system, Nagios Core 3.0 will operate properly. Network administrators must be able to observe the behavior of services and equipment in real time and also to predict which services and equipment cause more problems.

All of this is achieved through a graphical interface that can be used by any existing browser. The graphical interface gives details about the status of services and equipment, their history, performances and many more besides.

For the graphical interface to function, setting up a web server using CGI is required to generate dynamic web pages and also a module using PHP and a programming language for faster functioning.

5. Measurement Results

For each of the routers is monitored the response time, the status of the port 1 and uptime of the router, as shown in Fig. 2, Fig. 3 and Fig. 4.

R1	PING	OK	2015-06-14 10:43:39	9d 0h 7m 54s	1/4	PING OK - Packet loss = 0%, RTA = 19.64 ms
	Port 1 Link Status	OK	2015-06-14 10:41:38	0d 0h 23m 30s	1/4	SNMP OK - up(1)
	Uptime	OK	2015-06-14 10:41:33	0d 2h 3m 35s	1/4	SNMP OK - Timeticks: (783508) 2:10:35.08

Fig. 2. Services monitored for the router R1.

From Fig. 2 it can be seen that for the router R1:

- PING command has 0% packet loss and RTA of 19.63 milliseconds;
- status 1 port is functional;
- the time that the system has been in continuous operation is 2 hours 10 minutes and 35.8 seconds.

R2	PING	OK	2015-06-14 10:43:31	0d 1h 26m 37s	1/4	PING OK - Packet loss = 0%, RTA = 27.84 ms
	Port 1 Link Status	OK	2015-06-14 10:43:48	0d 0h 21m 20s	1/4	SNMP OK - up(1)
	Uptime	OK	2015-06-14 10:41:50	0d 2h 3m 18s	1/4	SNMP OK - Timeticks: (785076) 2:10:50.76

Fig. 3. Services monitored for the router R2.

From Fig. 3 it can be seen that for the router R2:

- PING command has 0% packet loss and RTA of 27.84 milliseconds;
- status 1 port is functional;
- the time that the system has been in continuous operation is 2 hours 10 minutes and 50.76 seconds.

R3	PING	OK	2015-06-14 10:41:42	0d 0h 2m 5s	1/4	PING OK - Packet loss = 0%, RTA = 26.70 ms
	Port 1 Link Status	OK	2015-06-14 10:43:39	0d 0h 21m 29s	1/4	SNMP OK - up(1)
	Uptime	OK	2015-06-14 10:43:58	0d 2h 1m 10s	1/4	SNMP OK - Timeticks: (798750) 2:13:07.50

Fig. 4. Services monitored for the router R3.

Tactical Monitoring Overview
 Last Updated: Sun Jun 14 10:39:35 PDT 2015
 Updated every 90 seconds
 Nagios® Core™ 3.5.1 - www.nagios.org
 Logged in as nagiosadmin

Monitoring Performance

Service Check Execution Time:	0.00 / 4.06 / 0.537 sec
Service Check Latency:	0.01 / 0.45 / 0.134 sec
Host Check Execution Time:	0.02 / 0.05 / 0.041 sec
Host Check Latency:	0.01 / 0.24 / 0.136 sec
# Active Host / Service Checks:	6 / 27
# Passive Host / Service Checks:	0 / 0

Network Outages
0 Outages

Hosts
0 Down 0 Unreachable 6 Up 0 Pending

Services
1 Critical 2 Warning 0 Unknown 24 Ok 0 Pending

Monitoring Features

Flap Detection	Notifications	Event Handlers	Active Checks	Passive Checks
✓ All Services Enabled No Services Flapping All Hosts Enabled No Hosts Flapping	✓ 1 Service Disabled All Hosts Enabled	✓ All Services Enabled All Hosts Enabled	✓ All Services Enabled All Hosts Enabled	✓ All Services Enabled All Hosts Enabled

Network Health
 Host Health: ██████████
 Service Health: ██████████

Fig. 5. General system view option.

From Fig. 4 it can be seen that for the router R3:

- PING command has 0% packet loss and RTA of 26.70 milliseconds;
- status 1 port is functional;
- the time that the system has been in continuous operation is 2 hours 13 minutes and 7.50 seconds.

Through the network general view option (Fig. 5) we can see if there are common problems with equipment or network services how long it lasts the verifications or the duration between two consecutive checks.

There is also an option to view a map of the network, as shown in Fig. 6. Through its, the application knows the position of the equipment.

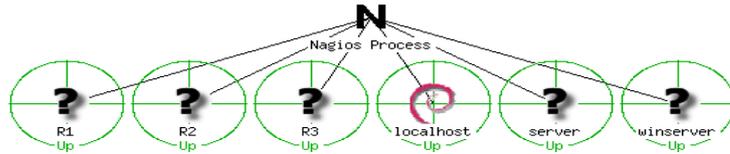


Fig. 6. Map of the monitored network.

Customers view option gives the opportunity to observe what equipment works or not, according to Fig. 7. We can also find out the date and time that occurred last check, the time when the equipment was placed into the application and effect of the command PING on equipment.

Host Status Details For All Host Groups

Limit Results:

Host	Status	Last Check	Duration	Status Information
R1	UP	2015-06-08 12:26:10	0d 1h 25m 51s	PING OK - Packet loss = 0%, RTA = 34.11 ms
R2	UP	2015-06-08 12:26:00	3d 1h 51m 43s	PING OK - Packet loss = 0%, RTA = 36.72 ms
R3	UP	2015-06-08 12:26:50	3d 1h 50m 53s	PING OK - Packet loss = 0%, RTA = 39.55 ms
localhost	UP	2015-06-08 12:27:40	211d 8h 50m 36s	PING OK - Packet loss = 0%, RTA = 0.03 ms
server	UP	2015-06-08 12:28:30	3d 1h 49m 13s	PING OK - Packet loss = 0%, RTA = 16.88 ms
winserver	UP	2015-06-08 12:29:20	2d 7h 11m 9s	PING OK - Packet loss = 0%, RTA = 36.34 ms

Results 1 - 6 of 6 Matching Hosts

Fig. 7. The status of the equipment in the network.

A feature of the GUI is to provide reports and graphs for equipment or services. In it we have several options, such as generating a report to check availability thrust while equipment or services, according to Fig. 8.

Service State Breakdowns:

State	Type / Reason	Time	% Total Time	% Known Time
OK	Unscheduled	0d 0h 0m 0s	0.000%	0.000%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
	Total	0d 0h 0m 0s	0.000%	0.000%
WARNING	Unscheduled	2d 13h 20m 22s	36.512%	36.512%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
	Total	2d 13h 20m 22s	36.512%	36.512%
UNKNOWN	Unscheduled	0d 12h 57m 23s	7.712%	7.712%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
	Total	0d 12h 57m 23s	7.712%	7.712%
CRITICAL	Unscheduled	3d 21h 42m 15s	55.776%	55.776%
	Scheduled	0d 0h 0m 0s	0.000%	0.000%
	Total	3d 21h 42m 15s	55.776%	55.776%
Undetermined	Nagios Not Running	0d 0h 0m 0s	0.000%	
	Insufficient Data	0d 0h 0m 0s	0.000%	
	Total	0d 0h 0m 0s	0.000%	
All	Total	7d 0h 0m 0s	100.000%	100.000%

Fig. 8. Changes of the system in the last 7 days.

From the "problems" option can be seen warnings on the equipment or service for faster troubleshooting of network issues, as shown in Fig. 9.

Display Filters:
 Host Status Types: All
 Host Properties: Any
 Service Status Types: All Problems
 Service Properties: Any

Limit Results: 100

Service Status Details For All Hosts

Host	Service	Status	Last Check	Duration	Attempt	Status Information
localhost	SSH	CRITICAL	2015-06-14 10:38:48	217d 6h 58m 7s	4/4	Connection refused
	Total Processes	WARNING	2015-06-14 10:39:08	217d 6h 57m 17s	4/4	PROCS WARNING: 339 processes
server	Total Processes	WARNING	2015-06-14 10:40:15	0d 1h 15m 17s	4/4	PROCS WARNING: 318 processes

Results 1 - 3 of 3 Matching Services

Fig. 9. The period during which the system will not issue notifications.

6. Conclusions

The results from simulations give us great flexibility on the multitude of devices and services that can be monitored and their presentation to the network administrator very easy to understand and very extensive as options.

For the average user, Nagios Core 3.0 shows its presence through a very good operating rate of networks and can monitor nearly any existing device, from a computer with a limited relevance to a router or server that is very important.

Thus, with this monitoring system we can perform the checking of very common contract clauses related to the total uptime of network components of 99.9%.

As future work we envision to develop a portable container Docker-based solution.

Acknowledgments. The work has been funded by the Sectorial Operational Programme Human Resources Development 2007-2013 of the Ministry of European Funds through the Financial Agreement POSDRU/159/1.5/S/134398. The work is supported by in part by UEFISCDI Romania under grants: “Scalable Radio Transceiver for Instrumental Wireless Sensor Networks - SaRaT-IWSN” (20/2012), MobiWay (PN-II-PT-PCCA-2013-4), DataWay (PNII-RU-TE-2014-4-2731) and by European Commission by: grant no. 262EU/2013 - “eWALL” support project, grant no. 337E/2014 - “Accelerate” project and by FP7 IP project no. 610658/2013 “eWALL for Active Long Living”. This work has been partially supported by the Italian Ministry of Research within PRIN project “GenData 2020” (2010RTFWBH).

We would like to thank the reviewers for their time and expertise, constructive comments and valuable insight.

References

1. J. Alcaraz Calero and Aguado Gutierrez, "MonPaaS: An Adaptive Monitoring Platform as a Service for Cloud Computing Infrastructures and Services", *Services Computing, IEEE Transactions*, 2015, vol.8, no.1, pp. 65-78
2. G. Suci, A. Vulpe, S. Arseni, A. Stancu, C. Butca and V. Suci, "Monitoring a Cloud-based Speech Processing System", *Electronics, Computers and Artificial Intelligence*, 2015, Romania
3. Arcaro, S.; Di Carlo, S.; Indaco, M.; Pala, D.; Prinetto, P.; Vatajelu, E.I., "Integration of STT-MRAM model into CACTI simulator," *Design & Test Symposium (IDT), 2014 9th International*, vol., no., pp. 67-72, 16-18 Dec. 2014
4. Muralimanohar, N.; Balasubramanian, R.; Jouppi, N., "Optimizing NUCA Organizations and Wiring Alternatives for Large Caches with CACTI 6.0," *Microarchitecture, 2007. MICRO 2007. 40th Annual IEEE/ACM International Symposium*, Dec. 2007
5. Hernantes, Josune; Gallardo, Gorka; Serrano, Nicolas, "IT Infrastructure-Monitoring Tools," *Software, IEEE*, vol.32, no.4, pp. 88-93, July-Aug. 2015
6. Dalle Vacche, A., & Lee, S. K. (2013). *Mastering Zabbix*. Packt Publishing Ltd.
7. Marik, Ondrej, and Stanislav Zitta. "Comparative analysis of monitoring system for data networks.", *Multimedia Computing and Systems (ICMCS), 2014 International Conference on*. IEEE, 2014.
8. Orebaugh, A., Ramirez, G., & Beale, J. (2006). "Wireshark & Ethereal network protocol analyzer toolkit". Syngress.
9. Lee, C.A.; Desai, N., "Approaches for Virtual Organization Support in OpenStack," *Cloud Engineering (IC2E), 2014 IEEE International Conference*, March 2014
10. G. Suci, S. Halunga, A. Ochian and V. Suci, "Network Management and Monitoring for Cloud Systems", *Electronic, Computers and Artificial Intelligence International Conference*, 2014, Romania.
11. Magda, S.M.; Rus, A.B.; Dobrota, V., "Nagios-based network management for Android, Windows and Fedora Core terminals using Net-SNMP agents," *Roedunet International Conference (RoEduNet), 2013 11th*, 2013
12. Jestratjew, A.; Kwiecien, A., "Performance of HTTP Protocol in Networked Control Systems," *Industrial Informatics, IEEE*, 2013
13. Netto, J.E.; Paulicena, E.H.; Silva, R.A.; Anzaloni, A., "Analysis of Energy Consumption Using HTTP and FTP Protocols Over IEEE 802.11," *Latin America Transactions, IEEE*, 2014
14. Sureswaran, R.; Al Bazar, H.; Abouabdalla, O.; Manasrah, A.M.; El-Taj, H., "Active e-mail system SMTP protocol monitoring algorithm," *Broadband Network & Multimedia Technology, 2009. IC-BNMT '09. 2nd IEEE International Conference*, Oct. 2009
15. Ghit, Bogdan, Florin Pop, and Valentin Cristea. "Epidemic-style global load monitoring in large-scale overlay networks." In *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2010 International Conference on*, pp. 393-398. IEEE, 2010.
16. Dobre, C., Pop, F., & Cristea, V. (2008, September). "A simulation framework for dependable distributed systems". In *Parallel Processing-Workshops, 2008. ICPP-W'08. International Conference on* (pp. 181-187). IEEE.