Comparison of system monitoring tools for large cluster system

Sung-Jun Kim[1], **Joon Woo**[1]
[1]Supercomputing Center, Korea Institute of Science and Technology Information

Abstract – *For system management, System administrators always check system log and service status. To reduce these efforts, there is various open-source system monitoring tools. As system sizes are getting larger, the performance of these tools is getting important. In this paper, we compare open-source monitoring tool's performance that most popular in the world – nagios and icinga .*

Keywords: nagios, icinga, system monitoring, cluster

1 Introduction

To stable manage large cluster systems; administrator should be recognized service status and failure as soon as possible. Normally, administrator are watching log periodically or use various monitoring tools to detect failures.

Nagios has been used at KISTI supercomputing center during the last 7 years; it offers quite a lot of features. But, new systems were installed in accordance with a steadily. Recently, nagios master server that collects monitoring data from remote hosts was not work properly. So, we need a more reliable and scalable monitoring tools and open source.

In this paper, we were evaluated performance of some tools before change it. We were compare two open source monitoring tools; Nagios and Icinga. Nagios is world famous and Icinga is a fork of nagios and backward compatible.

2 Backgrounds

2.1 Nagios

Nagios is an open source computer system monitoring, network monitoring and infrastructure monitoring software application. Nagios offers monitoring and alerting service for severs, switches, applications and services. It alerts the users when things go wrong and alerts them a second time when problem has been resolved [1].

Figure 1 shows Nagios architecture. Nagios core is the monitoring and alerting engine that serves as the primary application around which hundreds of Nagios projects are built. It serves as the basic event scheduler, event processor and alert manager for elements that are monitored.

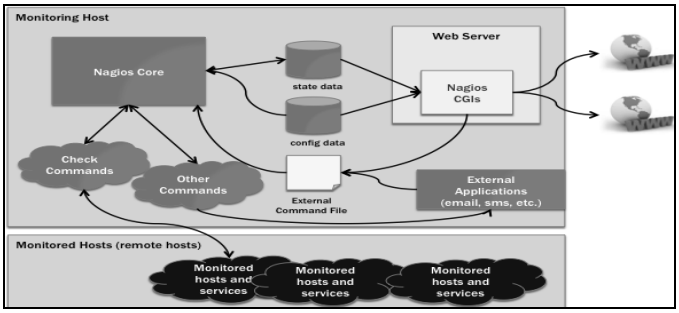

Figure 1 Nagios architecture

Plugins is used to verify services and devices. All Nagios host and service checks are performed by external plugins. A plugin command will be invoked by Nagios core as required, with arguments as specified in the command definition that was used.

2.2 Icinga

Icinga is an enterprise grade open source monitoring system which keeps watch over networks and any conceivable network resource, notifies the user of errors and recoveries and generates performance data for reporting [2].

Icinga is a fork of Nagios and is backward compatible. So, Nagios configuration, plugins and addons can all be used with Icinga. Though Icinga retains all the existing features of its predecessor, it builds on them to add many long awaited patches and features request by the user community.

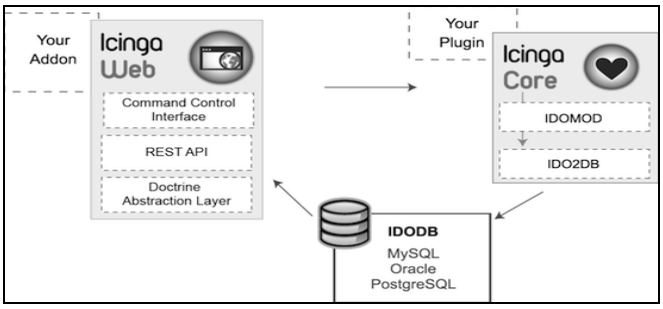

Figure 2 Icinga architecture

Figure 2 shows Icinga architecture. Like Nagios core, Icinga core does not check any services and hosts status. It is scheduling and processing of events and handle with alerts. Icinga used modern it techniques like Web 2.0 for web interfaces, mobile UI and supports Oracle and PostgreSQL.

3 Comparison

3.1 Test environments

We were built test environments using KVM to increase number of hosts dramatically. Our test servers are 17 nodes; Intel Xeon Quad Core 2.66 GHz, 4 GB memory. Test servers are consisting of three parts.

- Measure server: It is Ganglia server which checking performance of master nodes cpu, memory, I/O rate, etc.
- Master server: It is master server which collecting client server's status information.
- Client server: It is remote hosts that send their service status data to monitoring master.

Master and client servers have VMs using KVM. Master server has five VMs for combination of Nagios, Icinga and Mysql. Client servers have 9 VMs for NRPE (Nagios Remote Plugin Executor) to check their service status and report to master server. Finally, one physical node has 10 client nodes (1 domain server + 9 guest server). As a result, we built 150 virtual client servers using 15 physical servers. Each client server check 25 service status and master server is collecting about 4,000 services status check result from clients. Figure x. show architecture of testbed.

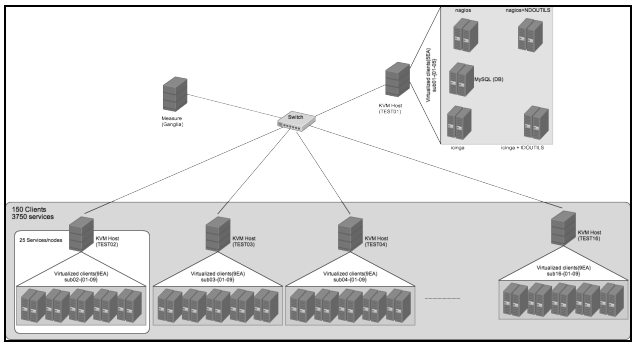

Figure 3 Architecture of Monitoring tools comaprison Testbed

3.2 Performance evaluations

As mentioned pervious section, we were using Ganglia to measure server side overload for each monitoring tools. We were tested Nagios and Icinga, with/without using database broker that stored status data to database.

For each case, testing was progressed during the week. During a test, the other monitoring servers were halt to avoid effect between test servers. We only used default setting to compare under same condition.

Table 1 Test server information

hostname	contents	hostname	contents
sub01-01	Nagios only	sub01-04	Nagios/DB
sub01-02	Icinga only	sub01-05	Icinga/DB

Table 1 is describing installed monitoring tools on each test servers.

Figure 4 shows cpu_user metric from Ganglia. It means CPU utilization used by user processor.

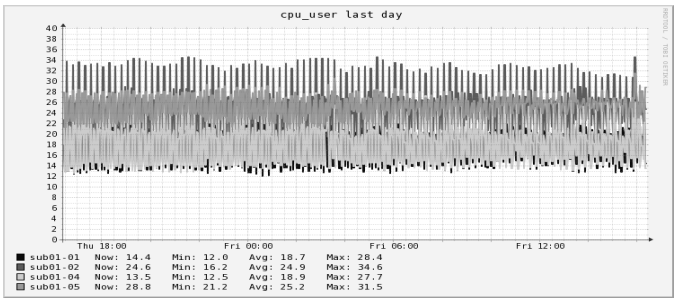

Figure 4 cpu_user metrics in Ganglia

Figure 5 shows cpu_wio metric from ganglia. It means the time that processor wait for I/O.

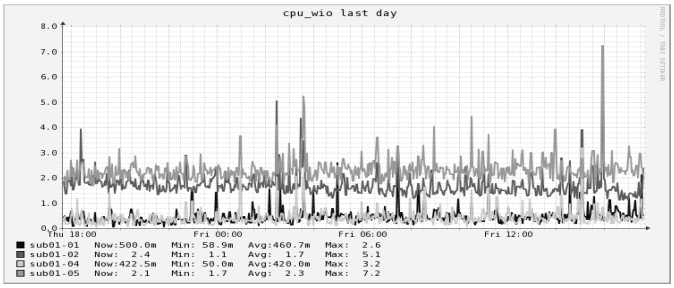

Figure 5 CPU_WIO metrics in Ganglia

Above Figures, Nagios has used fewer CPU resource and shorter I/O wait than icinga generally. In case of using database broker, usage of CPU resource was lower and I/O wait time was longer.

4 Conclusions

When we were deciding to compare to tools, we had expected that performance of icinga is better than Nagios. We have planned to migrate Nagios to icinga. However, the results were entirely opposite.

So, we are planning to change configuration Nagios for large installation tweaks instead of migrate Nagios to icinga. In the future, we will evaluate convenience of SQL queries and response time to get some data on GUI interfaces. And we are evaluating new version of these tools continually.

5 References

[1] Nagios Offical website, "http://www.nagios.org"

[2] Icinga Offical website, "http://www.icinga.org"