



Strengths and Limitations of Nagios as a Network Monitoring Solution

By
Sophon Mongkolluksamee



Agenda

- Network monitoring software
- About Nagios
- Limitations of Nagios
 - Improve with third-party add-ons
 - Without current solution
- Use Nagios as a framework for creating Network Monitoring Tool



Network monitoring software

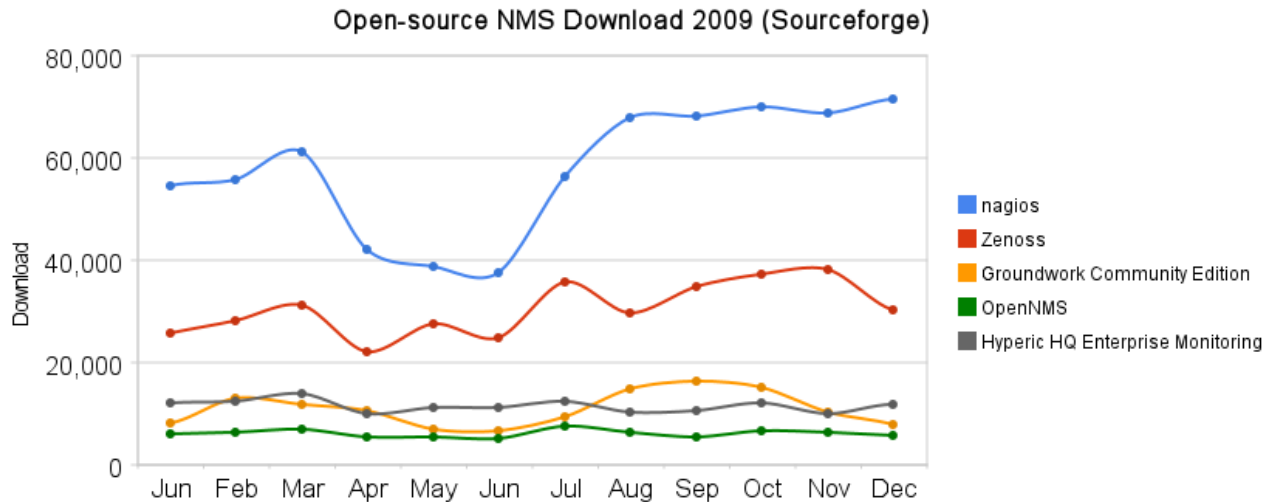
- Reduces a burden of network administrator with automatic checking of device and service status and error report
- Type of software
 - Commercial
 - Comprehensive features
 - Nice user interface
 - Cost a lot
 - Open-source
 - No technical support
 - Difficult to use
 - Free!





Nagios

- Nagios is the widely used open-source powered by a big developer community.



- Many users such as ISPs, governments and big enterprises (Yahoo, Amazon, Google)



Nagios

- The main tasks of Nagios are **to monitor status of network devices and their services** and to notify system administrators of network problems.
- Nagios perform **status check and notify a problem through the use of external “plugins”**, which are compiled executables or scripts (Perl, shell, etc.)
- The core of Nagios engine is a **scheduler daemon** that regularly executes plugins to probe specified network devices and their services.
- Nagios requires **text-based configuration** files to control all its activity.



Limitations of Nagios

- Some limitations are probably due to the minimalism philosophy of Nagios design concept.
- Some of the weaknesses can be fulfilled with add-ons or plugins from the Nagios community.
- Some limitations remain a challenge to fulfill.



Limitations that can be solved with third-party add-ons

Problem	Fixed with
Un-user friendly GUI	NagVIS, NetHAM
Lack of Database and Performance Records	NDOUtils, Opdb, NagiosGrapher
Difficult Configuration	Lilac, Fruity, NagiosQL, NConf
Lack of Automatic Device Discovery	NACE, check_find_new_hosts, Nmap2Nagios-ng

* Many add-ons are not straightforward and user-friendly. Administrator still need to tweak and adapt them to suit each network.



Limitations without current solutions

- Most of the problems are due to limitations of the native Nagios structure.
- Nagios makes no distinction among different types of devices like servers, routers, or switches.
- Nagios treats every device generically as a host.



Limitations without current solutions

- How to treat an “interface” of a router or a switch?

Treated interface as	result
Service	The interface will not show up on a network map and difficult to quickly trouble-shoot connectivity problem.
Host	<ul style="list-style-type: none">• Non-IP interface cannot be checked as a “host” by Nagios plugin• The interface shows up on a topology display as independent host device.• Need more levels of relationship between hosts.



Limitations without current solutions

- How to monitor and report status of a “link”?
 - Nagios has a link that represent parent-child relationship only.
 - No property such as propagation delay, channel quality, link utilization and bandwidth.
 - A link cannot treated as host or service.
- How to detect network **anomaly dynamically**?
 - Almost all Nagios check plugins use thresholds to classify levels of network status (critical, warning, ok).
 - User have to know all performance levels in advance before configuring plugin.

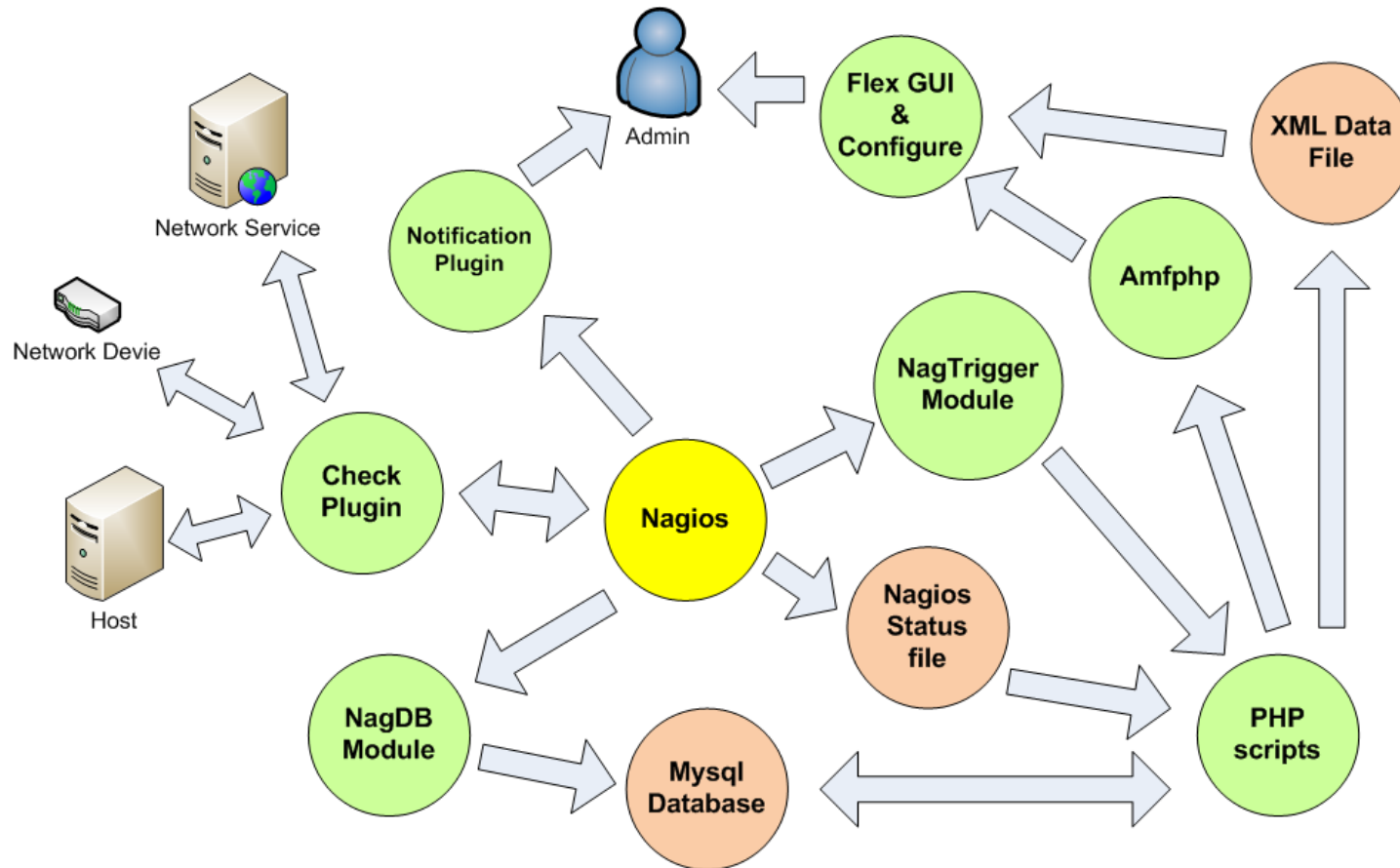


Nagios as Framework for creating a User-Friendly Network Monitoring Tool

- New prototype named NetHAM (Network Health Analysis and Monitoring)
- New user interface
 - Monitoring panel (with Adobe **Flex**)
 - Configuring panel (modified **NagiosQL**)
- New modules by using **Nagios Event Broker API**
 - **NagDB**: use to capture output returned by Nagios plugin.
 - **NagTrigger**: use to link internal Nagios event to external command executions.



NetHAM internal process





NetHAM user interface

- Monitor – Interactive GUI with flex

The screenshot displays the NetHAM user interface. At the top left is the NetHAM logo and the text "Network Health Analysis and Monitoring". At the top right are navigation links: "Monitor | Configure | Logged in : Admin | Logout".

The main content area is divided into two panels. The left panel, titled "Network topology", shows a network diagram with various devices and their connections. Devices are color-coded: green for OK, yellow for Warning, blue for Unknown, and red for Critical. The diagram includes components like FTPServer, Switch_A, IPv6_tb, wiki, Switch_B, Internet, Router_A, Switch_O, Webserver, Switch_C, dede, Epsilon, LinuxSIS, and WebServer.

The right panel shows a list of device status. It includes a "Refresh" button, tabs for "Device status", "Service status", and "Favorite", and a "Sorted by" dropdown menu with options "Status", "NAME", "TYPE", and "Operator". The status summary shows "Normal: 16 Abnormal: 18". The list includes:

- Abnormal Epsilon** (Epsilon, Admin: nagiosadmin)
- Abnormal Bacon** (BACON_Server, Admin: nagiosadmin...)
- Normal WebServer** (WebServer, Admin: nagiosadmin)

Below the list is a graph showing "Ping" results for the selected device (Epsilon) over time. The graph has a y-axis labeled "Ping" and an x-axis with time intervals from 0:00 to 8:00. The graph shows a series of red bars representing ping times, with a peak around 4:00. Below the graph are "Min: 0.28" and "Max: 4.84".

At the bottom of the interface, there is a copyright notice: "Copyright © by National Electronics and Computer Technology Center, NECTEC".

■ OK ■ Warning ■ Unknown ■ Critical



NetHAM user interface

○ Configure

NetHAM Network Health Analysis and Monitoring Monitor | **Configure** | Logged in : Admin | Logout

Configure

- Hosts
- Routers
- Services
- Host/Devices groups
- Service groups
- Host templates
- Service templates

Alarming

Commands

Specialties

Tools

Administration

Define hosts (hosts.cfg)

Basic settings | Advance settings

Basic settings

Host name*	<input type="text"/>	Description*	<input type="text"/>
Address*	<input type="text"/>		
Parents	<div style="border: 1px solid #ccc; padding: 2px;">Epsilon Bacon WebServer FTPServer</div>	Host/Device groups	<div style="border: 1px solid #ccc; height: 20px;"></div>
	<input type="radio"/> + <input type="radio"/> null <input checked="" type="radio"/> standard		<input type="radio"/> + <input type="radio"/> null <input checked="" type="radio"/> standard
Services	<div style="border: 1px solid #ccc; padding: 2px;">check-http check-alive chk-load chk-remote-cpu chk-remote-mem test-sevice-edit</div>	Device icon (*.gif)	<input type="text" value="Choose File"/> no file selected
Check status command	<input type="text" value="check-host-alive"/>	Active	<input checked="" type="checkbox"/>

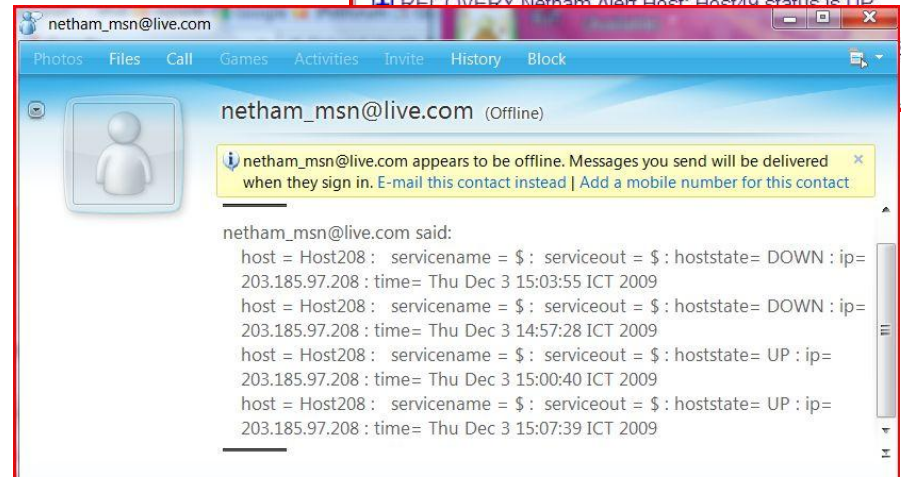
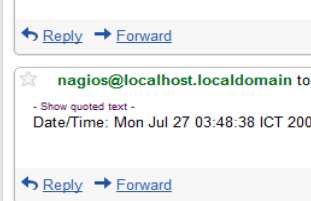
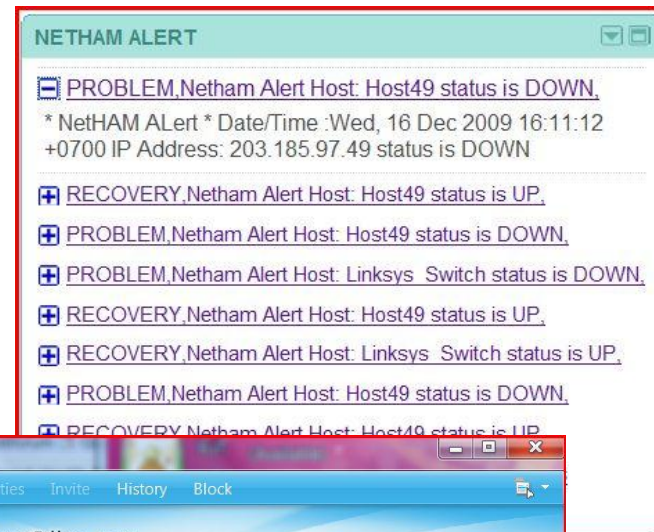
Command view ? command use



Example of NetHAM alert

When the network problem is found, NetHAM sends an alert to administrator via variety channels.

Service
HTTP
down!!





Conclusion

- Nagios is one of the most popular **open-source network monitoring tools**.
- Some limitations can be improved with third-party add-ons or plugins
- Some limitations still need to fulfill
- **Nagios can be used as a framework** for building more powerful and easy-to-use network monitoring software.



Thank you.

inms-nectec@googlegroups.com