Principles Of Good Monitoring

By Troy Lea
tlea@nagios.com
Intro

• Contractor for Nagios Enterprises
  • Develop & maintain documentation for Nagios products

• A discussion about the core principles behind monitoring
In The Beginning

• Monitoring is often implemented after an avoidable event
  • Like a server that ran out of disk space

• Initial monitoring solutions are easy to deploy for the basic 4
  • Host Up/Down; Disk; Memory; CPU

• The solution may be rushed to get it implemented
  • Not a lot of planning involved
Active vs Passive

- **ACTIVE** = your monitoring system is responsible for checking “stuff”
  - Usually on a schedule

- **PASSIVE** = “things” send “stuff” to the monitoring system
  - It is the “things” responsibility to send the “stuff”, scheduled or event triggered

- **SCENARIO**: UPS loses power due to black out
  - **ACTIVE** = You’ll find out about it when the next scheduled check occurs of the UPS
  - **PASSIVE** = The UPS can send an SNMP trap immediately to your monitoring solution
Back To The Beginning

- Expanding the monitoring solution
  - An ideal time to re-evaluate the chosen solution

- For example you may use the existing agent to do log monitoring
  - It includes a LOG module
  - The agent is being monitored with the ACTIVE method
  - Will this have a resource impact on your monitored servers?
  - Is this the best solution?
  - Are there better methods available?
# Active vs Passive Examples

<table>
<thead>
<tr>
<th>Type</th>
<th>ACTIVE</th>
<th>PASSIVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk, CPU, Mem</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Log</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>SNMP Metrics</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>SNMP Traps</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>External Devices (roaming laptops)</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Bandwidth Ingress/Egress</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Switch/Router/Firewall Flow Data</td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>
Agent or Agent-less

• Whatever method you chose will require some sort of config
  • Requires interaction with your monitored “things”

• The solution you choose requires planning
  • How will you make changes to the configuration later?
  • How will you install a newer version?

• An agent can be ACTIVE or PASSIVE
  • Their configuration dictates how they will work
Monitoring Solutions

- Nagios Core / Nagios Plugins
- Elastic Stack (ELK)
- Multi Router Traffic Grapher (MRTG)
- nfdump / nfcap
Nagios Core / Nagios Plugins

- A monitoring solution that schedules the checking of “things”
- ACTIVE and PASSIVE
- It can hook into other monitoring solutions

- Nagios Plugins is the companion project
- These are the programs that do the actual monitoring
- Anyone can write a plugin for Nagios Core
Elastic Stack (ELK)

• Log data monitoring solution
• PASSIVE
• Redundant database that can tolerate node failures without loss
• Stores the log data somewhere separate from the source
• Allows for forensic investigation later
• Powerful log data analysis functionality that is easy to use
Multi Router Traffic Grapher (MRTG)

- A tool used to monitor the traffic load on network links
- ACTIVE
- Queries SNMP enabled devices for IN and OUT metrics
- Switches, routers, firewalls, operating system NICs, OIDs like temperature and humidity
nfdump / nfcap

- Detailed network traffic (IN / OUT / PORT)
- PASSIVE
- Netflow and compatible flow data
- Switches, routers, firewalls, operating system NICs
Centralise Your Solution

• I don't mean "try and install it all on the one machine"
• Multiple solutions will have similar functionality for alerting
  • You can waste a lot of time re-inventing the wheel
• Nagios Core as an example
  • Sends notifications to contacts (with advanced escalation logic)
  • Could be an email / text / submitting a ticket to a ticketing system
• While Elastic can do similar functionality why try and duplicate it?
  • You can setup passive services on Nagios Core
  • Have Elastic send the alerts to these passive services
  • Rely on Nagios to handle the notification logic
What Is Monitoring Your Monitoring?

• How will you know when your monitoring solution is DOWN?

• Use another monitoring server to monitor the production instance

• Monitor the new server from your production instance

• If either instance goes down then you will be notified about it
Templates and Groups

• Try and approach things from a template and group methodology

• Templates allow you to define monitoring settings globally

• Use groups where possible

• Allow you to change a setting in one location
Documentation / Procedures

- Documentation beats any particular naming scheme every day of the week

- If you need to do something, document how you do it

- A document of 10 screenshots this is better than nothing
Agent Security

• Agents that use TLS for data encryption are recommended

• Define the network addresses that are allowed to talk to the agents
  • Worthwhile defining a backup address of your DR monitoring sever
Backups

Implement backups of your monitoring solution from the beginning !!!!!!!!!!!
Disaster Recovery (DR)

• How does monitoring fit into your DR plan?
The End

• Thank you for your time today!