Express-Guide
~to~
Basic Setup of
NAGIOS
an IT Infrastructure Monitoring Solution

by, **ABK** ~ http://www.twitter.com/**aBionic**

::Task Detail::

- Setting up Nagios machine on a LAN to monitor resources and services
- Generating an e-mail notifications if any of them goes down

---

::Background::

Links: http://www.nagios.org/

- Nagios is one of the most popular network-based IT Infrastructure Monitoring solution enabling organizations identify and resolve IT Infrastructure problems.

- There is a Nagios Server box monitoring all desired hosts; hosts either run Nagios Clients (like NRPE, more controllable) or get accessed over SNMP Service (this is better, if SNMPv3 is used).

::Execution Method::

To set up Nagios Server box and enabling Hosts to use it follow steps given below:

- Setup a **Nagios Server** Box
  {in respect to **Fedora/CentOS**, may change slightly with distro}

  - ***Pre-Requisite***:
    Installation of Apache, PHP, GCC Compiler, GD Developer Libraries as
    - `#yum -y install httpd php gcc`
    - `#yum -y install glibc glibc-common gd gd-devel`

  - ***Installation Step***:
    - Create a user account and group for it
      - shell commands:

```
$su -l
#useradd -m nagios
#passwd nagios
#groupadd nagcmd
#usermod -a -G nagcmd nagios
#usermod -a -G nagcmd apache
```

- Download latest version of 'Nagios Core' and 'Nagios Plugins' from respective sections at
  [Nagios Core:],
  [Nagios Plugins: http://www.nagios.org/download/plugins]

- Untar, Compile and Installing of Nagios {we had core-3.2.1 and plugins-1.4.14}
  - shell commands:
    ```
    #tar -zxf nagios-3.2.1.tar.gz
    #tar -zxf nagios-plugins-1.4.14.tar.gz
    #cd nagios-3.2.1
    #./configure –with-command-group=nagcmd
    #make all
    #make install
    #make install-init
    #make install-config
    #make install-commandmode
    #make install web-conf
    #htpasswd -c /usr/local/nagios/etc/htpasswd.users
    nagiosadmin
    #cd ../nagios-plugins-1.4.14
    #./configure --with-nagios-user=nagios –with-nagios-
    group=nagios
    #make
    #make install
    ```
  - Above when you are using 'htpasswd' utility, it will ask for a password which will be your password for userid 'nagiosadmin' to login to Nagios Web-App.
  - If you forget the password later, just rerun that single command to change password.

- Making Nagios a System Service
  - shell commands:
    ```
    #chkconfig --add nagios
    #chkconfig nagios on
    #service nagios start
    ```
    Note: If SELinux is in enforcing mode then, below commands also
    ```
    #chcon -R -t httpd_sys_content_t /usr/local/nagios/sbin
    #chcon -R -t httpd_sys_content_t usr/local/nagios/share
    ```

- ***Customizing Configuration***
  - The main configuration file will be '**`/usr/local/nagios/etc/nagios.cfg`**', with listing of all other configuration file.

So, if any configuration file is added or removed its entry in '**nagios.cfg**' should be altered respectively.

- You can configure hosts, services, host-groups, service-groups, contacts, contact-groups.
  You can also configure commands to be referenced in services using which those will be checked. Now these commands actually mention all configurations along-with the Script and its parameters to be executed to check for that service.
- This scripts are by default located at '/usr/local/nagios/libexec/' and respond in general output of return codes **0, 1, 2, 3** for OK, Warning, Critical, Unknown respectively.
- Among these commands, the SNMP based commands will also require SNMP Community String to be passed as an argument.
  Security Note: Don't use '**public**' as your community string, its not safe from Security perspective.
- By default these settings are to be performed manually at file-level configuration. It gives you greater control over the configuration-model you want to follow; depending on count and type of hosts and services.
  But, if you want a GUI then you can download free Nagios Addons like NConf or NagiosQL. They are PHP-based GUI front-ends to configure your Nagios more easily.

◆ Enabling **Hosts' Services to be monitored** by Nagios
  ◦ some '***Nagios Client***' or '***SNMP daemon***' needs to be configured on hosts to be monitored

◆ for **Nagios Client**
  ◦ use '**NRPE**' on **Linux** and '**NSClient++**' on **Windows**, available free

◆ for **SNMP**
  ◦ Installation:
    - On **Windows**, install it from
      'Start Menu' > 'Control Panel' > 'Add or Remove Programs' > 'Add or Remove Windows Components' > 'Management and Monitoring Tools' > 'Details' > 'Simple Network Management Protocol' > 'OK' > 'Next' > follow next instructions
    - On **Linux** Machine
      #yum -y install net-snmp net-snmp-utils

  ◦ Configuration Steps:
    - On Windows Machine
      - open 'Control Panel' > 'Administrative Tools' > 'Services'
      - select SNMP, open its properties
      - add the Community specified in Nagios configuration
      - allow access to Nagios Server IP to access SNMP Daemon.
    - On Linux Machines
      - edit snmpd.conf file
      - remove 'public' community

- add text (within '<>') changing ' *yourCommunityNameSpecified*' to required name as per your configuration
  < `rocommunity yourCommunityNameSpecified` >

- These settings are really basic (quick to go) policies. For an advanced and more secure version, go for policies mentioned in 'SNMP for secure resource monitoring' article or simply use SNMPv3 if you already know it.

::Tools/Technology Used::

- Nagios-Core : http://www.nagios.org/download/core
- Nagios-Plugins : http://www.nagios.org/download/plugins
- Net-SNMP : http://www.net-snmp.org/
- NSClient++ : http://www.nsclient.org/nscp/
- NRPE : http://sourceforge.net/projects/nagios/files/nrpe-2.x/
- NagiosMailACK : http://exchange.nagios.org/directory/Utilities/NagiosMailACK/details

---

::Inference::

- It's a highly scalable IT infrastructure monitoring and alerting solution with great architecture allowing entire community to contribute to its features as add-ons, services making it the most popular solution.

- It don't have a great GUI, just a decent one. Entire configuration is directly via files which is in a way great, because it gives more control over what actually gets set, and backup-restore is too easy also.

::Troubleshooting/Updates::

`these are few specific issues personally faced-&-solved`

- ***Problem:*** in one pre-configured nagios box the rendering of data vanished
  Solution:
  I searched for CSS files in the Web Files directory and found that Stylesheet File somehow got truncated on the server, replacing them with the original files solved it.

- ***Problem:*** a problem occurred in the Nagios Box configured initially, the notification mails weren't getting delivered rather got piled-up in mail queue.
  Solution:
  First we thought there is some problem with mail service, so we uninstalled it and installed another mail-service but still it didn't got resolved.
  Checking 'Mail Queue' of standard service and new mail service showed that still all mails were sent via old one.

Checked for system binaries, and found there were two extra binaries per mail-service other than default 'mail' service on linux. So, changing nagios configuration to that specific binary for sending mail solved the problem.

- ***Requirement:*** Nagios Web-Interface was available only over Organization's Network, that's not a thing to debate or change with respect to security.
  But to acknowledge the Alert Notification Mails from it, access was required to its UI. So, a work-around was required. The most suitable and quick was replying to that Notification mail itself for Acknowledgment, as alert-mail-notification worked fine.
  Solution:
  The online popular & available solution for it was setting up a ProcMail server on Nagios server and make its entry in Organization's MX Server to enable it receiving all replies. Then getting a perl script running to check for received mails and acknowledge the alert on correct pattern-acknowledgment mail.
  Though it was nice, but required new entries in MX Server per Nagios box. A new service on a box means a new security vector.
  Also required one more server to be run on Nagios Box increasing the already spiking resource load.
  - So, I developed a Nagios Supporting Service '`NagiosMailACK`'
    { could be studied/downloaded: http://sourceforge.net/projects/nagiosmailack/ }
    It was low on resource consumption and just required a single GMail ID to receive acknowledgments for all Nagios Servers by using different Nagios_ID like parameters for different Nagios boxes.

- ***Requirement:*** a Guest user with permissions to only view Host & Service status
  Solution: firstly adding a guest user to htpasswd.users file
  - at shell
    #htpasswd /usr/local/nagios/etc/htpasswd.users guestUser
    remember don't attach '-c' switch to it, otherwise old users will get replaced
  - Now, open '/usr/local/nagios/etc/cgi.cfg' for editing and change in following way
    - find following (or similar) lines
      `authorized_for_all_services=nagiosadmin`
      `authorized_for_all_hosts=nagiosadmin`
    - add new user entry to make it look like
      `authorized_for_all_services=nagiosadmin, guestUser`
      `authorized_for_all_hosts=nagiosadmin,guestUser`

- ***Requirement:*** implementing Disk-Space Check for alerts
  Solution: already a Perl Script checking for desired Drive Space at SNMP was present online so simply downloaded (checked all its functions for security issue) and added it to commands list passing suitable arguments.
