PacNOG 5: Network Management Workshop

Nagios Exercises

```
PART I
-----------------------------------------------------------------------------

1. Install Nagios version 3

    Do this as root.

    # apt-get install nagios3

2. Create the Web user password file:

    # htpasswd -c /etc/nagios3/htpasswd.users nagiosadmin

    New password:
    Re-type new password:

   Please use the class password.


2. You should already have a working Nagios!

    - Open a browser, and go to

    http://localhost/nagios3/

    - At the login prompt, login as:

        user: nagiosadmin
        pass:

3. Let's look at the interface together...

    # cd /etc/nagios3/

    # ls -l
    -rw-r--r-- 1 root root    1882 2008-12-18 13:42 apache2.conf
    -rw-r--r-- 1 root root   10524 2008-12-18 13:44 cgi.cfg
    -rw-r--r-- 1 root root    2429 2008-12-18 13:44 commands.cfg
    drwxr-xr-x 2 root root    4096 2009-02-14 12:33 conf.d
    -rw-r--r-- 1 root root      26 2009-02-14 12:36 htpasswd.users
    -rw-r--r-- 1 root root   42539 2008-12-18 13:44 nagios.cfg
    -rw-r----- 1 root nagios  1293 2008-12-18 13:42 resource.cfg
    drwxr-xr-x 2 root root    4096 2009-02-14 12:32 stylesheets

    # ls -l conf.d/

    -rw-r--r-- 1 root root 1695 2008-12-18 13:42 contacts_nagios2.cfg
    -rw-r--r-- 1 root root  418 2008-12-18 13:42 extinfo_nagios2.cfg
    -rw-r--r-- 1 root root 1152 2008-12-18 13:42 generic-host_nagios2.cfg
    -rw-r--r-- 1 root root 1803 2008-12-18 13:42 generic-service_nagios2.cfg
    -rw-r--r-- 1 root root  210 2009-02-14 12:33 host-gateway_nagios3.cfg
    -rw-r--r-- 1 root root  976 2008-12-18 13:42 hostgroups_nagios2.cfg
    -rw-r--r-- 1 root root 2167 2008-12-18 13:42 localhost_nagios2.cfg
    -rw-r--r-- 1 root root 1005 2008-12-18 13:42 services_nagios2.cfg
    -rw-r--r-- 1 root root 1609 2008-12-18 13:42 timeperiods_nagios2.cfg
```

PART II

----

1. According to what we saw in class, let's add a host to monitor.

    - Pick any PC in the room, maybe your neighbor's PC.

```
    $ su -
    # cd /etc/nagios3/conf.d/
    # vi pcNNN.cfg

define host {
    use         generic-host
    host_name   pcNNN
    alias       PC NNN at APRICOT2009
    address     _______________       [pcNNN's IP address here]
}
```

    ... Save and quit.

2. Let's create a new hostgroup for the occasion, and add our host to it

    - Edit the file hostgroups_nagios2.cfg and add a new group. Do this at the bottom of the file:

```
    # vi hostgroups_nagios2.cfg

define hostgroup {
    hostgroup_name  servers
    alias           PacNOG5 PCs
    members         pcNNN
}
```

3. Now let's associate some services to that host

```
    # vi services_nagios2.cfg
```

- Find the section called "check that ssh services are running", and change the line:

```
hostgroup_name                  ssh-servers
```

to

```
hostgroup_name                  ssh-servers, servers
```

4. Verify that your configuration file is OK:

```
    # nagios3 -v /etc/nagios3/nagios.cfg
```

... You should get :

```
Total Warnings: 0
Total Errors:   0

Things look okay - No serious problems were detected during the check.
```

5. Reload/Restart Nagios

```
    # /etc/init.d/nagios3 restart
```

6. Go to the web interface (http://localhost/nagios3) and check the host you just added

7. Add ALL the PCs in the room!

- Add all the PCs in the room to the config

- Check HTTP for all PCs in the room

- Remember to verify the configuration file!

- I suggest that you create a single configuration file to do this. (i.e., pcs.cfg, or servers.cfg, etc.).

PART III

----

1. Create a parent-child relationship in Nagios. Your PC has a

parent which is the switch it is attached to. The switch has a parent, which is the router it is attached to.

Let's create this relationship.

2. Create a file to define the configuration for your switches. Maybe "/etc/nagios3/conf.d/switches.cfg". We'll start by just entering information for the switch to which your PC is attached:

```
# cd /etc/nagios3/conf.d
# touch switches.cfg
# vi switches.cfg
```

Your switch is either mgmt-sw1 or mgmt-sw2

```
define host {
      use         generic-host
      host_name   mgmt-swN
      alias       switch for 192.168.2.N/25
      address           192.168.2.NNN
      parents           bb-gwN
}
```

be sure that you enter the correct values for "N". You can refer to our classroom Network Diagram to figure these out:

http://192.168.1.224/trac/wiki/network

2. Create a file to define the configuration for the router Maybe "/etc/nagios3/conf.d/routers.cfg".

The router you use is the parent of the switch above...

```
define host {
      use         generic-host
      host_name   bb-gwN
      alias       router for for 192.168.2.N/25
      address           192.168.2.NNN
}
```

Save and exit from the file.

3. Edit the file pcNNN.cfg and add a parent entry to this file:

```
# vi pcNNN.cfg
```

```
define host {
    use         generic-host
    host_name   pcNNN
    alias       PC NNN at APRICOT2009
    address     192.168.2.NNN
    parents     mgmt-swN
}
```

Save and quit. Eventually we'll need to update the parent for your localhost as well, but we'll do this later.

4. In preparation for putting in multiple pc, switch and router entries let's create the initial hostgroups for each of these. Edit the file /etc/nagios3/conf.d/hostgroups_nagios2.cfg and at the bottom of the file add the following:

```
# vi hostgroups_nagios2.cfg
```

Go to the bottom of the file. You should already have an entry for "servers" for pcNNN:

```
##
## Our local hostgroup definitions
##

define hostgroup {
      hostgroup_name   servers
      alias      PacNOG5 PCs
      members          pcNNN
}

define hostgroup {
      hostgroup_name switches
        alias          PacNOG5 switches
      members          mgmt-swN
}

define hostgroup {
        hostgroup_name  routers
        alias           PacNOG5 routers
        members         bb-gwN
}
```

For now we won't add services to our switches or routers, but later on you may wish to check to see if SSH is running on these devices.

5. Verify that your configuration is OK:

```
# nagios3 -v /etc/nagios3/nagios.cfg
```

... You should get :

```
Total Warnings: 0
Total Errors:   0

Things look okay - No serious problems were detected during the check.
```

6. Reload/Restart Nagios

```
# /etc/init.d/nagios3 restart
```

7. Go to the web interface (http://localhost/nagios3) and check the see how things look.

7. See if the parent-child relationship seems reasonable by clicking on the Status Map link on the left-hand side of the Nagios page.

   In Status Map choose "Balanced Tree" from the Layout Method drop-down menu and click the Update button. Do you see a parent-child relationship as you might expect?

PART III
----

1. Create a complete Nagios configuration for our classroom network.

   NOTES:

   - This requires more planning. You have switches, routers, and the NOC (if you wish to add it). In addition, the IP addresses that you use are for your network router, the classroom router, and the other network's router depend on your position in the network.

   - You want to use internal IP address for your network's router, and the gateway router.

   - Note that the switches are not running Telnet, they are using ssh. So you should do either an ssh check on them or a standard ping check (the Nagios default).

   - It is important that you properly define the parent for devices. Some examples are given below. Devices can have more than one parent, and in our classroom this is true. The two switches lan1-lan2-sw and lan3-lan4-sw have two parents since they have a single administrative interface, but they are connected by two routers each.

3. Complete the switches configuration (/etc/nagios3/conf.d/switches.cfg). There should be thre entries in this file, the 2 switches for 192.168.2.0/25 and for 192.168.2.128/25, and the backbone switch.

4. Complete the routers configuration (/etc/nagios3/conf.d/routers.cfg). There should be two entries in this file for router bb-gw1 and bb-gw2.

5. Complete the PCs configuration. Perhaps change the filename:

   /etc/nagios3/conf.d/pcNNN.cfg to:

   /etc/nagios3/conf.d/servers.cfg

   ```
   # cd /etc/nagios3/conf.d
   # mv pcNNN.cfg servers.cfg
   ```

   This file should have a entries for each classroom pc. Remember to choose the correct parent for each one, including the NOC box.

6. In the file "/etc/nagios3/conf.d/hostgroups_nagios2.cfg" complete the hostgroups for all the routers, switches and pcs in the classroom.

   Sample entry:

```
# hostgroup definition for APRICOT 2009 Network Management Workshop
define hostgroup {
        hostgroup_name routers
        alias          Cisco Routers at APRICOT 2009
        members
}

define hostgroup {
      hostgroup_name   servers
      alias      PacNOG5 PCs
      members          pc10N, pc20N, ...
}
```

7. In the file "/etc/nagios3/conf.d/services_nagios2.cfg" you define what groups (not individual devices) will have what service checks run on them.

   Sample entry:

```
# check that ping-only hosts are up
define service {
        hostgroup_name                  routers,switches,servers
        service_description             PING
        check_command                   check_ping!100.0,20%!500.0,60%
        use                             generic-service
        notification_interval           0 ; set > 0 if you want to be renotified
}
```

7. The file "/etc/nagios2/conf.d/extinfo_nagios2.cfg" defines details for each device defined. Feel free to take a look at the extinfo_nagios2.cfg file we are already using for our classroom to get a feel for what you can do. Compare your status map to the one on the classroom NOC machine. Notice the difference, maybe, with icons?

   You can view the NOC's extinfo_nagios2.cfg file here:

   http://192.168.1.224/configs/nagios/conf.d/extinfo_nagios2.cfg

9. You might consider changing the file /etc/nagios3/conf.d/localhost_nagios2.cfg.

10. Naturally you can get entire set of Nagios configuration files for this network that will only need a few changes for your machine from the NOC web server if you wish.

    http://192.168.1.224/configs/nagios/conf.d/

11. You sill need to update a few files. Including:

    ```
    /etc/nagios3/conf.d/routers.cfg
    /etc/nagios3/conf.d/servers.cfg
    ```

    You should make sure that you have the correct IP addresses defined in routers.cfg for your network view, and you will want to comment out your pcs entry in the file pcs.cfg

    You may have to make additional changes and to troubleshoot this using the "Nagios pre-flight check":

    ```
    # nagios3 -v /etc/nagios3/nagios.cfg
    ```

    Remember to restart Nagios for changes to take affect.

```
    # /etc/init.d/nagios3 restart
```

PART IV

---

1. Allow for "guest" user access to view your Nagios web pages.

```
    # cd /etc/nagios3
    # vi cgi.cfg
```

Find these two lines (they are together)

```
    authorized_for_all_services=nagiosadmin
    authorized_for_all_hosts=nagiosadmin
```

And change them to read:

```
    authorized_for_all_services=nagiosadmin,guest
    authorized_for_all_hosts=nagiosadmin,guest
```

2. Save the file.

3. Now you must create the guest user and password.

4.

```
    # htpasswd htpasswd.users guest
    New password:
    Re-type new password:
```

You can use any password you want. It's pretty typical if "guest" only has "view" access to the Nagios web pages to pick a very simple password, like "guest".

5. Restart Nagios for the changes to take affect:

```
    # /etc/init.d/nagios3 restart
```

6. Next time you are asked for a password to view the Nagios web pages you can use "guest/guest" (if you chose "guest" as a password) to view them instead of using "nagiosadmin".

PART V

NOT TO BE DONE.

This session is for reference only.

1.) Here we will tie in the ability of Nagios and Trac to work together to help document your network. The concept if quite simple. First, go to your local Trac project install page at:

```
http://localhost/trac/netmanage
```

Log in as the admin user so that you can edit the Trac wiki.

2.) Create an entry for your PC in the wiki. You can do this by clicking on the "Edit this page" button and entering in a link like this (example for PC1, use your PC number instead):

```
[wiki:PC1 PC1] : '''169.223.140.1'''
```

Save the page.

Alternately, have a look at the main classroom wiki to see what has been done:

```
http://noc.mgmt.conference.apricot.net/
```

3.) Click on the PC1 item that's grey with a question mark. Now create this page. Enter in some text about your PC and save the page.

4.) In Nagios you need to edit the file:

```
/etc/nagios3/conf.d/extinfo_nagios2.cfg
```

and update your PCs entry in this file with a line like this:

```
notes_url       http://localhost/trac/netmanage/wiki/PC1
```

You can place this on a line after the "host_name" entry. Remember to change "PC1" to your PCs number.

5.) Restart Nagios.

6.) If you look in your Nagios Service Detail view there should now be a new icon next to your machine's entry. This looks like a folder. Click on this and the URL you entered for the notes_url entry in the extinfo_nagios2.cfg file will open. You can, also, click on the machines' icon in the graph views, then click again and this page will open.

PART V (OPTIONAL)

1.) Now we will create a plug-in for Nagios. This plug-in will do the following:

* Ping a set of (external) servers.
* If one server is down a warning will be generated.
* If two servers are down a critical state will be generated.

This will be part of our scripting session. The instructions for doing this are here:

http://ws.edu.isoc.org/workshops/2008/ait-net-manage/presos/scripting/bash.html

These were written for Nagios version 2, but are fine for version 3. Just replace occurrences of "/etc/nagios2" with "/etc/nagios3".

PART VI

1.) We will update our Nagios contacts definion, "/etc/nagios3/conf.d/contacts_nagios3.cfg" to add a local user to that will receive alerts for certain condition.

2.) Next we will add another user for our RT ticketing system so that a ticket is automatically generated for specific events.

3.) Edit the file "/etc/nagios3/contacts_nagios3.cfg":

```
# vi /etc/nagios3/contacts_nagios3.cfg
```

In a web browser open up the sample contacts_nagios3.cfg file and adapt this to work with what you have. Basically, just replace yours with this one.

Go to:

```
http://noc.mgmt.conference.apricot.net/configs/etc/nagios3/conf.d/ \
       contacts_nagios3.cfg
```

4.) Once the files is updated you might have noticed the two lines that read:

```
    service_notification_commands   notify-service-ticket-by-email
    host_notification_commands      notify-host-ticket-by-email
```

The "notify-service-ticket-by-email" and "notify-host-ticket-by-email" commands are new. You need to create these in the file /etc/nagios3/commands.cfg.

This is not strictly necessary. For purposes of this exercise you can replace these two commands with:

```
    service_notification_commands   notify-service-by-email
    host_notification_commands      notify-ticket-by-email
```

and skip skip part "4a" if you wish.

4a) These two commands are set aside so that if you wish you can adjust the formatting of the email that Nagios sends to be more user friendly to the RT ticketing system. This is up to you. To create these two commands we simply copy the original commands and renamve them in /etc/nagios3/commands.cfg.

The easiest way to see this is to open a web browser and go to:

http://noc.mgmt.conference.apricot.net/configs/etc/nagios3/commands.cfg

and then you can copy and past the new items in to your commands.cfg file on your machine. Note that you could change the names of these if you wish as long as you match the new name to what is in the /etc/nagios3/contacts_nagios3.cfg file.

5.) Once you have updated your contacts_nagios3.cfg file, then run the Nagios pre-flight check:

```
# nagios3 -v /etc/nagios3/nagios.cfg
```

If it all looks good, then restart Nagios:

```
# /etc/init.d/nagios3 restart
```

Or, less intrusive is:

```
# /etc/init.d/nagios3 reload
```

6.) Now we need to create a proper alias in our /etc/aliases file using the rt-mailgate program to pipe email from Nagios to RT and to the correct queue.

Edit the file /etc/aliases:

```
# vi /etc/aliases
```

And add the following lines to the bottom of the file:

```
alerts:         "|/usr/bin/rt-mailgate --queue 'Network Management' --action co
alerts-comment: "|/usr/bin/rt-mailgate --queue 'Network Management' --action co
```

Make note in the file and verify that there is a line that, also, reads:

```
root: netmanage
```

This tells the mail system to deliver all mail sent to root@localhost to the netmanage account instead.

Save the file and quit. In reality we'll only be using the "alerts" alias at this time.

After you've saved and exited from the /etc/aliases file run:

```
# newaliases
```

which lets the Postfix MTA know about changes to /etc/aliases. If you run in to any problems with errors about rt-mailgate, verify that it is installed by doing:

```
# apt-get install rt3.6-clients
```

this should have been done when you first installed RT.

7.) Now you should go to your RT instance installed on your machine.

http://localhost/rt

log in as "root".

Click on the "Configuration" link, "Queues", "New queue": Be sure that you fill in the "Queue Name" field with "Network Management" - including the upper-case 'N' and 'M'.

You only need to fill in Queue Name and Description. Click the "Save Changes" button on the lower right of the screen.

Now click the "User Rights" link. You'll see that the 'root' user has no rights on this queue. Give your 'root' user enough rights on this queue to at least see tickets in the queue and see the queue itself. If you want you can be lazy and highlight all the rights and assign 'root' everything. You have to press "Modify User Rights" to do this.

At this point log out of RT and log back in. You should see the Network Management queue listed on the right of the page.

Now you need to generate a Nagios alert so that a ticket is created in RT. If you noticed in the /etc/nagios3/conf.d/contacts_nagios3.cfg the Nagios "alerts' queue only sends notifications if a service is in the "c" or "critical" state, or if a host is "d" or "down". In addition in the file /etc/nagios3/conf.d/generic-service_nagios2.cfg there is a line that reads:

```
notification_interval          0
```

This ensures that Nagios will only send one (1) email per critical or down state. If this is set to something else, then you will generate multiple tickets, which is not good.

Try to generate an alert from Nagios, which should generate a ticket in RT by doing something. You could check for a service on your neighbor's PC that does not exist. You could pull the network cable on your neighbor's PC so that it appears to be down. Otherwise, your instructor will come up with something as well.

Last update 22 February 2009 by HA