Nagios Installation and Configuration

Notes:
------
* Commands preceded with "$" imply that you should execute the command as
  a general user - not as root.
* Commands preceded with "#" imply that you should be working as root.
* Commands with more specific command lines (e.g. "RTR-GW>" or "mysql>")
  imply that you are executing commands on remote equipment, or within
  another program.

Exercises

PART IX
Optional Exercises
---

1. Check that nagios is Running

As opposed to just checking that a web server is
running on the classroom PCs, you could also check that the nagios3
service is available, by requesting the /nagios3/ path. This means
passing extra options to the check_http plugin.

For a description of the available options, type this:

```
# /usr/lib/nagios/plugins/check_http
# /usr/lib/nagios/plugins/check_http --help
```

and of course you can browse the online nagios documentation or google
for information on check_http. You can even run the plugin by hand to
perform a one-shot service check:

```
# /usr/lib/nagios/plugins/check_http -H localhost -u /nagios3/
```

So the goal is to configure nagios to call check_http in this way.

```
define command{
        command_name    check_http_arg
        command_line    /usr/lib/nagios/plugins/check_http -H '$HOSTADDRESS$' $ARG1$
        }

define service {
        hostgroup_name                  nagios-servers
        service_description             NAGIOS
        check_command                   check_http_arg!-u /nagios3/
        use                             generic-service
}
```

and of course you'll need to create a hostgroup called nagios-servers to
link to this service check.

Once you have done this, check that Nagios warns you about failing
authentication (because it's trying to fetch the page without providing
the username/password). There's an extra parameter you can pass to
check_http_arg to provide that info, see if you can find it.

WARNING: in the tradition of "Debian Knows Best", their definition of the

check_http command in /etc/nagios-plugins/config/http.cfg
is *not* the same as that recommended in the nagios3 documentation.
It is missing $ARG1$, so any parameters to pass to check_http are
ignored. So you might think you are monitoring /nagios3/ but actually
you are monitoring root!

This is why we had to make a new command definition "check_http_arg".
You could make a more specific one like "check_nagios", or you could
modify the Ubuntu check_http definition to fit the standard usage.

2. Check that SNMP is running on the classroom NOC

- First you will need to add in the appropriate service check for SNMP in the file /etc/nagios3/conf.d/services_nagios2.cfg. This is where Nagios is impressive. There are hundreds, if not thousands, of service checks available via the various Nagios sites on the web. You can see what plugins are installed by Ubuntu in the nagios3 package that we've installed by looking in the following directory:

```
# ls /usr/lib/nagios/plugins
```

  As you'll see there is already a check_snmp plugin available to us. If you are interested in the options the plugin takes you can execute the plugin from the command line by typing:

```
# /usr/lib/nagios/plugins/check_snmp
# /usr/lib/nagios/plugins/check_snmp --help
```

  to see what options are available, etc. You can use the check_snmp plugin and Nagios to create very complex or specific system checks.

- Now to see all the various service/host checks that have been created using the check_snmp plugin you can look in /etc/nagios-plugins/config/snmp.cfg. You will see that there are a lot of preconfigured checks using snmp, including:

  snmp_load
  snmp_cpustats
  snmp_procname
  snmp_disk
  snmp_mem
  snmp_swap
  snmp_procs
  snmp_users
  snmp_mem2
  snmp_swap2
  snmp_mem3
  snmp_swap3
  snmp_disk2
  snmp_tcpopen
  snmp_tcpstats
  snmp_bgpstate
  check_netapp_uptime
  check_netapp_cupuload
  check_netapp_numdisks
  check_compaq_thermalCondition

  And, even better, you can create additional service checks quite easily.
  For the case of verifying that snmpd (the SNMP service on Linux) is running we need to ask SNMP a question. If we don't get an answer, then Nagios can assume

that the SNMP service is down on that host. When you use service checks such as check_http, check_ssh and check_telnet this is what they are doing as well.

- In our case, let's create a new service check and call it "check_system". This service check will connect with the specified host, use the private community string we have defined in class and ask a question of snmp on that ask - in this case we'll ask about the System Description, or the OID "sysDescr.0" -

- To do this start by editing the file /etc/nagios-plugins/config/snmp.cfg:

editor /etc/nagios-plugins/config/snmp.cfg

At the top (or the bottom, your choice) add the following entry to the file:

```
# 'check_system' command definition
define command{
       command_name    check_system
       command_line    /usr/lib/nagios/plugins/check_snmp -H '$HOSTADDRESS$' -C '$ARG1$' -o s
        }
```

You may wish to copy and paste this vs. trying to type this out.

Note that "command_line" is a single line. If you copy and paste in your editor, the line may not wrap properly and you may have to manually "join" the two lines so they are one.

- Now you need to edit the file /etc/nagios3/conf.d/services_nagios2.cfg and add in this service check. We'll run this check against all our servers in the classroom, or the hostgroup "debian-servers"

- Edit the file /etc/nagios3/conf.d/services_nagios2.cfg

editor /etc/nagios3/conf.d/services_nagios2.cfg

At the bottom of the file add the following definition:

```
# check that snmp is up on all servers
define service {
        hostgroup_name                  snmp-servers
        service_description             SNMP
        check_command                   check_system!xxxxxx
        use                             generic-service
        notification_interval           0 ; set > 0 if you want to be renotified
}
```

The "xxxxxx" is the community string previously (or to be) defined in class.

Note that we have included our own community string here vs. hard-coding it in the snmp.cfg file earlier. You must change the "xxxxx" to be the snmp community string given in class or this check will not work.

- Now we must create the "snmp-servers" group in our hostgroups_nagios2.cfg file. Edit the file /etc/nagios3/conf.d/hostgroups_nagios2.cfg and go to the end of the file. Add in the following hostgroup definition:

```
# A list of snmp-enabled devices on which we wish to run the snmp service check
define hostgroup {
       hostgroup_name   snmp-servers
               alias    snmp servers
               members  noc
      }
```

- Note that for "members" you could, also, add in the switches and routers for group 1 and 2. But, the particular item (MIB) we are checking for "sysDescr.0" may not be available on the switches and/or routers, so the check would then fail.

- Now verify that your changes are correct and restart Nagios.

- If you click on the Service Detail menu choice in web interface you should see the SNMP check appear for the noc host.

- After we do the SNMP presentation and exercises in class, then you could come back to this exercise and add in all the classroom PCs to the members list in the hostgroups_nagios2.cfg file, snmp-servers hostgroup definition. Remember to list your PC as "localhost".