System monitoring with Open Monitoring Distribution (OMD), hands-on tutorial

Autor: Iñigo Aldazabal Mensa <inigo_aldazabal@ehu.es>

Date: 2014/01/14 - HPC Knowledge Meeting'14, Barcelona

Step-by-step guide for a system monitoring installation and initial configuration using Nagios and Check_MK collection of extensions for Nagios. We will use the pre-packaged Open Monitoring Distribution (OMD) system which bundles both Nagios and Check_MK, as well as many other Nagios extensions into a single, pre-configured package and brings the setup, configuration and maintenance of the monitoring system to a new level of simplicity.

We will use two CentOS virtual machines (*VMs*) to be able follow this tutorial, but the same procedure should be applicable with minimal changes to any of the distributions supported by OMD.

Contents

1 Required software

- VirtualBox, version 4.3.6, has been used througout the tutorial:

  Virtualization software https://www.virtualbox.org/

  CentOS Virtual Machines from http://virtualboxes.org/images/centos/
- Virtual Machine for the OMD server: CentOS 6.3 with GNOME desktop graphical environment

  http://sourceforge.net/projects/virtualboximage/files/CentOS/6.3/CentOS-6.3-x86.7z
- Virtual Machine to be monitored: CentOS 5.7 base

  http://sourceforge.net/projects/virtualboximage/files/CentOS/5.7/CentOS-5.7-i386.7z
- Open Monitoring Distribution - OMD, version 1.10

  http://omdistro.org/
- Check_MK

  Collection of Nagios extensions / plugins, already integrated in OMD.

  Check_MK monitoring and inventory agent to be installed in the monitored systems http://mathias-kettner.com/check_mk.html, version 1.2.2p3.

2 VirtualBox configuration

The two virtual machines we are using are (see http://virtualboxes.org/images/centos/)

CentOS 5.7 base x86

Size:	(compressed/uncompressed) 173 MBytes / 1.3 GBytes
Link:	http://sourceforge.net/projects/virtualboximage/files/CentOS/5.7/CentOS-5.7-i386.7z
Active user account(s):	(username/password) root/reverse.
Notes:	text mode installed, no graphics

CentOS 6.3 Gnome Desktop x86

Size:	(compressed/uncompressed) 492 MBytes / 2.2 GBytes
Link:	http://sourceforge.net/projects/virtualboximage/files/CentOS/6.3/CentOS-6.3-x86.7z
Active user account(s):	(username/password) root/reverse, centos/reverse.
Notes:	GNOME desktop environement, install from LiveCD; Guest Additions NOT installed.

Now we download the virtual machines, extract them and open the corresponding `.vbox` files from the VirtualBox Manager (`Machine | Add`).

> ***Note***
>
> If we get an error about the disc UUID already being used (eg. because we just copied this virtual machine for another test) we have to change the `.vdi` virtual disk UUID with the command:
>
> ```
> VBoxManage internalcommands sethduuid CentOS-5.7.vdi
> ```

and update the "HardDisk uuid" section in the configuration file `.vbox`.

2.1 Internal network configuration

We want to set up an internal network for the virtual machines to be able to communicate each other.

First we make sure we have an internal network configured in the VirtualBox server (`VirtualBox Manager -> File | Preferences | Network | Host-only Networks`). Make sure you have:

PC VirtualBox Host

IP: 192.168.56.1

We also have to add a `Host-only Adapter` to each virtual machine (`Virtual Machine Manager: select the VM -> settings | network | Adapter 2 | Enable + attached to "Host-only Adapter"`).

From the "Advanced" section We write down the network "card" MAC address in order to later set up static IP addresses whithin the internal network. In this case the MACs we have and IPs we will use are:

CentOS 6.3 - OMD monitoring server

MAC: 08:00:27:C1:99:2D
IP: 192.168.56.10

CentOS 5.7 - monitored system

MAC: 08:00:27:42:79:DF
IP: 192.168.56.11

3 Monitoring Server configuration and Nagios / OMD install

3.1 Network configuration

After booting the virtual machine first enable ssh access as it is disabled by default:

```
chkconfig ssh on
service sshd on
```

Then setup the static IP by creating the file `/etc/sysconfig/network-scripts/ifcfg-eth1`:

```
#/etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
BOOTPROTO=none
IPADDR=192.168.56.10
NETMASK=255.255.255.0
ONBOOT=yes
HWADDR=08:00:27:C1:99:2D
DEFROUTE=yes
NAME="eth1"
```

and restart the network:

```
service network restart
```

3.2 email sending configuration

First lets check whether we already can send emails straigth from postfix over port 25:

```
echo "Test mail from postfix" | mail -s "Test Postfix" user@domain
```

If we do not get the message at user@domain check the postfix log at `/var/log/maillog`. In this case it may be necessary to set up a relay host for postfix in `/etc/ postfix/main.cf`. We can eg. use Google SMTP servers for testing.

> ***Note***
>
> Use `tail -f /var/log/maillog` while testing to see the postfix behaviour. In order to check/clean the postfix queue use `mailq` and `postsuper -d ALL` commands.

3.2.1 Postfix relay using Gmail

We follow the guide at http://blog.earth-works.com/2013/05/14/postfix-relay-using-gmail-on-centos/, with a summarized version reproduced here just for completeness.

Install SASL nedded modules:

```
yum install cyrus-sasl-plain
```

Create `/etc/postfix/sasl_passwd` with just one line (adapt to your gmail user data):

```
smtp.gmail.com     GmailUsername:GmailPassword
```

Secure the thing:

```
chown postfix /etc/postfix
postmap hash:/etc/postfix/sasl_passwd
chown root:root /etc/postfix/sasl_passwd*
chmod 640 /etc/postfix/sasl_passwd*
```

Edit the `/etc/postfix/main.cf` configuration file, and add the following lines at the end:

```
#Set the relayhost to the Gmail SMTP server
relayhost = smtp.gmail.com:587

#Set the required TLS options
smtp_tls_security_level = secure
smtp_tls_mandatory_protocols = TLSv1
smtp_tls_mandatory_ciphers = high
smtp_tls_secure_cert_match = nexthop

#Check that this path exists -- these are the certificates used by TLS
smtp_tls_CAfile = /etc/pki/tls/certs/ca-bundle.crt
```

```
#Set the sasl options
smtp_sasl_auth_enable = yes
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd
smtp_sasl_security_options = noanonymous
```

Restart postfix service:

```
service postfix restart
```

Test:

```
echo "Test email from postfix with Gmail relay" | mail -s "Gmail-postfix test" user@domain
```

3.3 OMD installation

We follow the quickstart CentOS installation instructions straigth from the OMD web page at http://omdistro.org/doc/quickstart_redhat just adapting everything to our CentOS version (6) and architectura (i386).

First install the `epel` repository configuration

```
rpm -Uvh http://download.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm
```

and then download and install the ~100MB OMD rpm package:

```
wget http://files.omdistro.org/releases/centos_rhel/omd-1.10-rh61-31.i386.rpm
yum install --nogpgcheck omd-1.10-rh61-31.i386.rpm
```

In our case this installs 36 packages and upgrades 4, with a total download size of 24MB.

> ***Note***
>
> We could have instead used the Consol* Labs OMD repository in order to have the latest version available at hand. Setting it up is trivial, just follow the guidelines at https://labs.consol.de/repo/stable.

3.4 OMD initial setup

The `omd` command is used to manage OMD sites. `omd` can be executed as site user to modify just that site, or as root user. As the root user `omd` offers more option such as copying, renaming, disabling or uninstalling sites. Calling `omd` alone provides see a list of options.

3.4.1 OMD site creation and access

To create and start a new OMD test "site" instance just:

```
omd create test
omd start test
```

When creating a new *site* OMD, amongst other things, creates a new user in the system which will be used to manage this specific site. Thus we can have different *sites* for different purposes as testing, production, upgrading, etc. (see http://mathias-kettner.com/checkmk_install_with_omd.html)

In order to manage our site we just `su -` to the site/user:

```
su - test
```

The `test` user home directory is `/omd/sites/test`. Here all the local configurations, caches, performance data, etc. for this site will be kept, specifically in the `tmp`, `var` and `etc` directories (the rest of the directories are symlinked to your OMD version. See http://mathias-kettner.com/checkmk_install_with_omd.html for a detailled description of the file/folder structure and contents.

3.4.2 Web server access configuration

> ***Note***
>
> Default user/password for the OMD interface is **omdadmin/omd**

Once the test site is up we try to access to it web interface from within the own machine first at http://localhost/test. In our case we get a error "OMD: Site not started". This is documented in the OMD FAQ specifically for CentOS and related systems and it is related to the selinux configuration. Just do:

```
/usr/sbin/setsebool -P httpd_can_network_connect 1
```

`-P` makes the change persistent and it may take a while to run, even some minutes, so be patient. After this we can access the web interface from the localhost without problems.

If we want to access to the web interface from remote machines (as the VirtualBox physical host in this case) we have to enable the service in the CentOS firewall, activated by default. Just run:

```
/usr/bin/system-config-firewall-tui
```

go to *Customise* (<TAB> moves between fields), scroll down the list up to *WWW (HTTP)* and enable the service with <SPACE>. Then select *Close*, *OK* and *YES*.

Now you can access the OMD web interface at http://192.168.56.10 eg. from your VirtualBox physical host.

4 Monitorized system configuration

After booting the machine (CentOS-5.7) up we just set the static IP and then install the `check_mk` agent.

4.1 Network configuration

As before, in order to set up a static IP we create the file `/etc/sysconfig/network-scripts/ifcfg-eth1`:

```
#/etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
```

```
BOOTPROTO=none
IPADDR=192.168.56.11
NETMASK=255.255.255.0
ONBOOT=yes
HWADDR=08:00:27:42:79:DF
DEFROUTE=yes
NAME="eth1"
```

and restart the network:

```
service network restart
```

4.2 check_mk agent installation

We download and install the `check_mk` monitoring agent from the check_mk webpage without further complications, the only needed dependence being `xinetd`:

```
wget http://mathias-kettner.com/download/check_mk-agent-1.2.2p3-1.noarch.rpm
wget http://mathias-kettner.com/download/check_mk-agent-logwatch-1.2.2p3-1.noarch.rpm
yum install --nogpgcheck check_mk-agent-1.2.2p3-1.noarch.rpm \
    check_mk-agent-logwatch-1.2.2p3-1.noarch.rpm
```

If desired we can restrict the access to the agent execution in this machine to the OMD monitoring service so we have a more secure setup eg. in a production environment. In order to do this we just add to the `/etc/xinetc.d/check_mk` file the line:

```
$> vim /etc/xinetc.d/check_mk
...
only_from = 192.168.56.10
...
```

and we reload the `xinetd` daemon configuration:

```
$>/etc/init.d/xinetd reload
```

4.3 Hard disk monitoring with S.M.A.R.T.

If we are monitoring a physical host we will be interested in monitoring their hard disk health status. Check_mk does not includes S.M.A.R.T. checking by default, but provides a `plugin` that has to be explicitly installed in the remote host.

The plugin is called `smart` and it already is in the OMD server, we just have to copy over to the desired host:

```
# su - test
# scp ~/share/check_mk/agents/plugins/smart  \
      user@remote-host:/usr/lib/check_mk_agent/plugins/smart
```

If we have not yet inventorized the host the smart check will be present when doing it, otherwise you have to reinventorize it and the new check will appear. Will see later how to do it.

5 Basic Check_MK configuration

We will do the basic monitoring system setup using at first *WATO - Check_MK's Web Administrator Tool* through the *Multisite* web interface, both part of the Check_MK echosystem.

We will first setup a new user who will get the test alerts and after this we will add the hosts to be monitored ann test some alerts.

5.1 User creation

Every user (*contact* in the Nagios nomenclature) belongs to a *contact group*, which are the ones which are really assigned to host and services notifications. In the default OMD/check_MK configuration we have only one contact group, **"Everybody"**, so we will add the new contact to this group, also making sure that we check the **"Administrator"** role in the Security section and that we **"enable notifications"** in the notifications section:

```
( WATO-Configuration | Users & Contacts | New User )
```

We save the changes (**"Save"** in the lower part of the new user creation form) and we are bougth back to the "User & Contacts" main section, where we have a notice about the **"1 Changes"** done. In order to propagate the change to the Check_MK/Nagios configuration click on the **"1 Changes"** button and then on the **"Activate Changes!"** one. We can now see the newly created user in the "Users & Contacts " WATO section and also can checks that the user is a member of the "Everybody" group in the "Contact Goups" section.

5.2 Integration (inventory) of the new *host* to be monitored

In order to add (inventorize, in the Check_MK slang) a new host (in which we have already installed the check_mk agent), we just:

```
( WATO-Configuration | Hosts & Folders | New host )
```

There we just add the **"Hostname"** (CentOS-5.7), **"IP"** if needed (192.168.56.11 in this case), **"Permissions" -> "Everybody"** and **"Alias"** (if desired). Clicking on **"Save & go to Services"** brings us to the autodetected host services list, where we can choose to ignore some of the automatically detected checks. We then **"Save manual check configuration"** and as we did before we **"Activate Changes!"**.

Going to the main web interface page (Check_MK logo in the upper left or `(Views | Dashboards | Main Overview)` we see that we have one host and 19 services monotirized.

..note:

```
It is convenient to use the own monitoring server to monitor itself. For
this we just install the check_mk agent in the server and add the host
*localhost* in WATO. Do it!
```

5.3 Reinventoring

If we add new checks to a host through check_mk plugins, legacy nagios checks, NRPE nagios checks, etc., we can make Check_MK to scan this host for new, not inventorized services. Just go to `(WATO | Hosts & Folders )`, click on the desired host and then select "Services" and "Full Scan". New services will be detected and you can enable them at will, as well as disable existing checks if wanted.

> ***Note***
>
> When reinventoring a host all previously inventorized checks, performance data, graphs, etc. are kept.

5.4 email notification test

In order to test email notifications go to a host (`Views | Hosts | All hosts`) and click on a service name. In the service information page click on the hammer icon in order to run commands over this service. Then go to **"Various Commands" -> "Fake check results"** and eg. click **"Critical**". Confirm the action and see eg. in the `Dashboard | Main Overview` the service being Critical for a while and the notifications being sent. Check you email for the Critical State notification and the Recovery one a minute later when the service comes back to normal state!

6 Advanced Check_MK configuration

7 References

Virtual Machines

- Oracle VirtualBox, multiplatform virtualization system: https://www.virtualbox.org/
- CentOS preinstalled VirtualBox virtual machines: http://virtualboxes.org/images/centos/

Nagios

- Web: http://www.nagios.org/
- Official Documentation: http://nagios.sourceforge.net/docs/nagioscore/3/en/toc.html
- Nagios Exchange: Nagios extension and checks open repository http://exchange.nagios.org/
- *"Building a Monitoring Infrastructure With Nagios"*, David Josephsen, Prentice Hall 2007

Check_MK

- Web: http://mathias-kettner.com/check_mk.html
- Official Documentation: http://mathias-kettner.com/checkmk.html

OMD

- Web: http://omdistro.org/