Nagios

Michael Pobega (pobegam@sunyit.edu)

May 30, 2017

Abstract

Documentation of the Nagios host at 150.156.16.78. Covers the installation and configuration of Nagios, deploying and configuring NRPE hosts and using Ansible to automate some tasks.

Contents

1 Quick System Information

1.1 Software Versions

- **Oracle Enterprise Linux** release 6.5 (Santiago)
- **Kernel** 3.8.13-26.1.1.el6uek.x86_64
- **Nagios** 3.5.1-1.el6.x86_64
- **nagios-plugins-all** 1.4.16-10.el6.x86_64

1.2 Important File Locations

- **Nagios configuration files** /etc/nagios/
- **Nagios HTTPD config file** /etc/httpd/conf.d/nagios.conf
- **Nagios plugins for yum install** /usr/lib64/nagios/plugins
- **Nagios plugins for source install** /usr/local/nagios/libexec
- **NRPE configuration file** /etc/nagios/nrpe.cfg
- **NRPE configuration directory** /etc/nrpe.d/
- **Related Ansible Playbooks (avl-sysans-001)** :
  - **~ansible/scripts/nagios/nrpe.yml:** installs nrpe from source on the target machine(s). Specify your arguments on the command line as per the documentation in the header comments of the playbook. Also opens port 5666 for Nagios in /etc/sysconfig/iptables to allow connections to the NRPE daemon.
  - **~ansible/scripts/nagios/custom-commands.yml:** update the custom-commands configuration file and upload/update custom shell scripts to /etc/nrpe.d/. Used to deploy NRPE check scripts to all hosts simultaneously.
- **Nagios Graph** :
  - **Installation Directory** /usr/local/nagiosgraph/
  - **Enabling in Apache2** added *include /usr/local/nagiosgraph/etc/nagiosgraph-apache.conf* to **/etc/conf/httpd.conf**

2 Preface

The information below may or may not be applicable to systems installed after the original date. Version numbers are included for completeness and may not reflect the exact versions installed from the official OEL repositories.

If you are using this documentation to reinstall Nagios please read the official Nagios documentation located at `http://www.nagios.org/documentation` to make sure that there haven't been any huge changes between the version documented here and the version being installed.

Compare with the version numbers above!

3 Nagios

3.1 Installation

Nagios is installed from the standard OEL repositories, while nrpe and nagios-plugins-all are installed from EPEL. Make sure EPEL is enabled and run

Extended Packages for Enterprise Linux

 # yum install nagios nrpe nagios-plugins-all

This will install Nagios, NRPE and all of the plugins the standard CentOS way (see section 1 for more information.) At this point in time your Nagios installation is actually done, just open port 80 and configure all of your hosts.

If you are migrating to a newer version of Nagios copying and pasting the configuration files over should be enough. Use *nagios -v /etc/nagios/nagios.conf* to verify your configuration before starting/stopping the daemon.

3.2 Configuration

There are multiple configure files and directories of interest under /etc/nagios

- **/etc/nagios/nagios.cfg** Main Nagios configuration file.
- **/etc/nagios/hosts/** directory of host configuration files. One file per host, with hostname.cfg as the file name (ex. hosts/avl-sysamp-001.cfg)
- **/etc/nagios/hostgroups/** directory of hostgroup configuration files. One file per hostgroup, with hostgroup.cfg as the file name (ex. hostgroups/nrpe.cfg)
- **/etc/nagios/objects/** all of the template files are included here. These are a bit on the unorganized side, but each file should be self explanatory based on the filename.
- **/etc/nagios/httpd-group.cfg** HTTPD hostgroup file. A list of usernames that are allowed to access and modify Nagios from the web interface.

3.3 Allowing New Users to view Nagios

Add the specified user to the **/etc/nagios/httpd-group.cfg** file above and restart httpd. After the web server comes back up the new user will be allowed access (assuming correct credentials have been supplied).

3.4 Adding Hosts

To add a host create a new file under */etc/nagios/hosts/* and call it hostname.cfg (ex. avl-sysamp-001.cfg).

This is just to make our lives easier and make the directory easy to parse with a simple *ls*

```
define host {
   use        linux-server
   host_name     avl-sysamp-001
   alias      avl-sysamp-001
   hostgroups    linux-servers,nrpe
```

```
   address       150.156.16.5
   contact_groups  systems
   }

define service {
   use        generic-service
   host_name     avl-sysamp-001
   service_description No. of Sessions
   check_command    check_nrpe!check_logins!-a 7 10
    }
```

Ampere's config file

Make sure that you put the host into the proper hostgroup(s) it belongs to and don't redefine services unless you are trying to override the hostgroup's configuration.

see section 3.5 for information on configuring hostgroups

3.5 Adding Hostgroups

To add a new hostgroup create a new file under */etc/nagios/hostgroups/* and call it hostgroupname.cfg (ex. nrpe.cfg).

```
define hostgroup{
       hostgroup_name nrpe
       alias          NRPE servers
       }

define service{
   use        generic-service
   hostgroup_name  nrpe
   service_description RAM
   check_command    check_nrpe!check_ram
}

define service{
   use        generic-service
   hostgroup_name  nrpe
   service_description Root FS
   check_command    check_nrpe!check_root
}
```

NRPE hostgroup's config file

Make sure to put the host into the proper hostgroup(s) it belongs to and don't redefine services unless you are trying to override the hostgroup's configuration.

3.5.1 Adding Hosts to Hostgroups

Adding hosts to hostgroups is a simple, if sometimes time consuming task. To add a host to the hostgroup you just list the hostgroup in that host's configuration file (see Ampere's configuration file above). A host may be in more

than one hostgroup, and services defined by hostgroups may be overridden in the host's file (in the case where you want HTTP to be checked on a port other than 80, etc) as long as the **service_description** directives match.

3.6 Advantages and Disadvantages

The configuration layout described above has it's advantages and disadvantages. One disadvantage is that having everything in separate files could become troublesome down the line (*"Just where is that thing configured?"*), but if all future changes are documented and follow the hierarchical structure described in the document, it should stay fairly readable.

One huge advantage of this layout is that it's easy to use command line tools to find pertinent information in the configuration files. Wondering how SSH is defined on all hosts? *grep ssh /etc/nagios/hosts/*.cfg.* Want to grep through the hostgroups and hosts? *grep ssh /etc/nagios/host*/*cfg.*

The configuration becomes a little more complicated, but the management of the system becomes worlds easier.

4 Nagios Graphing

Nagiosgraph was installed from `http://nagiosgraph.sourceforge.net/`. It's the default installation with no modifications made to anything. It is installed to */usr/local/nagiosgraph/*

To get graphing to work on a specific services you have to add an **action_url** to the service. The **action_url** is recognized in both host files and hostgroup files.

```
define service {
   use        generic-service
   host_name     avl-sysamp-001
   service_description No. of Sessions
   check_command    check_nrpe!check_logins!-a 7 10
    action_url
        /nagiosgraph/cgi-bin/show.cgi?host=$HOSTNAME$&service=$SERVICEDESC$'
        onMouseOver='showGraphPopup(this)' onMouseOut='hideGraphPopup()'
        rel='/nagiosgraph/cgi-bin/showgraph.cgi?host=$HOSTNAME$&service=$SERVICEDESC$
    }
```

Snipper from Ampere's config file demoing action_url

The action_url is what makes the graph show up on the Nagios frontend. Mouse over for quick access to the last 24 hours of monitoring!

4.1 Nagiosgraph Workaround

For some reason Nagiosgraph would stop updating graphs automatically after a while for seemingly no reason. To work around this I've added a cronjob that runs */usr/local/nagiosgraph/bin/insert.pl* as root every 24 hours to force the graphs to update.

5 NRPE

NRPE stands for **Nagios Remote Plugin Executor**. It runs on the host being monitored by Nagios and listens on port 5666 for a TCP connection. After connecting to NRPE Nagios runs a script on the remote host to do a local check (such as RAM or disk usage.)

Don't forget to open 5666 on your firewall!

5.1 Installing NRPE on a host

First we are going to need some build tools, so run

 # yum install gcc make automake openssl openssl-devel xinetd

Next we're going to compile Nagios Plugins from source. Fetch nagios-plugins-1.5.tar.gz from `http://www.nagios.org/download/plugins`. Untar it and run configure with these options:

 # ./configure –sysconfdir=/etc/nagios –enable-ssl –enable-command-args

This file is also available on avl-sysans-001 in /srv/nfs/nrpe/

Finalize the installation by running *make* and *make install*.

Now we need to compile the nrpe daemon itself. Grab nrpe-2.15.tar.gz from `http://sourceforge.net/projects/nagios/files/nrpe-2.x/nrpe-2.15/nrpe-2.15.tar.gz/download`. Just like before we are going to use the same arguments to configure the installation

 # ./configure –sysconfdir=/etc/nagios –enable-ssl –enable-command-args

Again, this file is available on avl-sysans-001 in the same directory as the Nagios plugins

Then similar to above, run *make all* and *make install* to install it.

Now we need a user for NRPE to run as. Go ahead and **add a nagios user and group** to the system.

5.1.1 Configuration Files

For simplicity sake rather than explain every single configuration option I will list below where the files exist on avl-sysans-001 and where they belong on the NRPE host.

avl-sysamp-001	NRPE host
/srv/nfs/nrpe/nrpe.cfg	/etc/nagios/nrpe.cfg
/srv/nfs/nrpe/xinetd.d/nrpe	/etc/xinetd.d/nrpe
/srv/nfs/nrpe/libexec/check_mem.pl	/usr/local/nagios/libexec/check_mem.pl
/srv/nfs/nrpe/custom-commands.cfg	/etc/nrpe.d/custom-commands.cfg

Table 1: Config file locations

Make sure to copy these files over and chown them to nagios:nagios (or nrpe:nrpe if nrpe was installed via EPEL.)

Don't forget to open port 5666 on the firewall and make sure SELinux is either disabled or allowing NRPE to do it's thing.

Add nrpe to the list of system services by appending the following line to **/etc/services**

nrpe 5666/tcp # Nagios NRPE

Finally, make sure that xinetd is started and enabled at boot by issuing the following commands

 # chkconfig xinetd on
 # service xinetd restart

5.2 Host Discrepancies

When originally deploying NRPE we used the packages from EPEL instead of compiling from source. The problem is that the yum packages expect certain files to be in different directories than the source installation does.

	EPEL installation	Source installation
daemon:	nrpe	xinetd
user:	nrpe	nagios
plugin dir:	/usr/lib64/nagios/plugins/	/usr/local/nagios/libexec/

Table 2: Discrepancies between EPEL and source NRPE

(There are also some legacy hosts that have nrpe installed in an odd way – you can figure out which these are by using check_nrpe from Nagios with no commands to see the nrpe version)

5.2.1 Handling discrepancies during automation

The beginning of custom-commands.yml displays a great way to differentiate the hosts from each another based on if and how nrpe has been installed.

```
- name: check if /etc/nrpe.d/ exists
  stat: path=/etc/nrpe.d/
  register: nrpe_exists
- name: check if /etc/init.d/nrpe exists
  stat: path=/etc/init.d/nrpe
  register: nrpe_initd
- name: check if /etc/xinetd.d/nrpe exists
  stat: path=/etc/xinetd.d/nrpe
  register: nrpe_xinetd
```

Snippet from "custom-commands.yml"

The logic here should be obvious. If the directory /etc/nrpe.d/ exists, NRPE is installed. If /etc/init.d/nrpe exists, it's been installed via EPEL. And failing that if /etc/xinetd.d/nrpe exists it's installed from source.

5.3 Adding custom commands to NRPE hosts

Custom commands are defined in the NRPE configuration files. You can define them in /etc/nagios/nrpe.cfg, or any other file/directory that configuration includes. Our deployment includes every cfg file in the directory /etc/nrpe.d/, and generally in that directory there are two files: **custom-commands.cfg** and **local-commands.cfg**. This directory also houses any custom made shell scripts that are used in-house to run as nrpe custom commands.

custom-commands.cfg should be on every server. It is what we use to deploy new commands to all of our NRPE hosts, and it is updated on all NRPE servers simultaneously using Ansible.

local-commands.cfg is not on every server. It is only for commands local to that server.

5.4 NRPE as a Nagios hostgroup

There are two Nagios hostgroups for NRPE hosts; *nrpe* and *oldnrpe*. *nrpe* represents all of the hosts that are running a more recent version of NRPE, including our **custom-commands.cfg**. When using Ansible to keep everything up to date, all the servers in the *nrpe* group can be expected to have the same behavior. *oldnrpe* is the Nagios hostgroup for the legacy hosts running NRPE. These hosts were configured and managed by a previous system administrator, and are currently next on the slate for retirement. We don't update these, and just pray to $DIETY that they don't break.

6 NSClient++

Grab the NSClient++ installer from the netapp. The installer is located at:

\\files.sunyit.edu\information_technology_services\Systems\Software\NSClient++

Install either **NSCP-0.4.1.102-Win32.msi** or **NSCP-0.4.1.102-x64.msi** based on the server's architecture. Be sure to read **NSClient_Installation.docx** for more documentation on installing and configuring NSClient++

7 Related Ansible Playbooks

7.1 nrpe.yml

nrpe.yml is our playbook for deploying NRPE to new hosts. It must be run with the argument *–extra-vars="hosts=hostgroup"* (as documented in the comments of the playbook) or like so:

ansible-playbook all -i avl-syswut-001, nrpe.yml

(keep in mind the trailing comma)

7.2 custom-commands.yml

custom-commands.yml updates all of the hosts in the *[nrpe]* group in **/etc/ansible/hosts**. It uploads the custom-commands.yml from /srv/nfs/nrpe/ to the remote hosts, swaps out the [NAGPLUGDIR] with the proper plugin directory for the type of NRPE installation, and uploads/updates all of the scripts from /srv/nfs/nrpe/libexec/.

To add a new commands to all of the NRPE hosts you would start by adding it to **/srv/nfs/nrpe/custom-commands.yml**. If it requires a new executable (a custom shell script or binary) place the file into **/srv/nfs/nrpe/libexec/**. Finally when everything is in place run **/home/ansible/scripts/nagios/custom-commands.yml** by invoking it on the command line:

ansible-playbook custom-commands.yml

To have the changes reflected in Nagios add a new command to the nrpe group in **/etc/nagios/hostgroups/nrpe.cfg**, then restart Nagios and all of the hosts should now have the new command!

8 Future Plans for the Server

- **Monitoring switches** This is entirely feasible using SNMP, but within the time constraints I don't think this will be possible. If everything else gets finished quicker than expected I will come back to this.
- **Reorganizing Contact Groups** Simple configuration changes.
- **24-Hour Digest** a script that would dump a short version of the daily problems/recoveries and e-mail them out. Would be useful for people who want to know what problems have arisen but don't want to be subscribed to every admin email alert.