## 搭建nagios环境

### 基本介绍

网络服务监控(SMTP、POP3、HTTP、NNTP、ICMP、SNMP、FTP、SSH)

主要资源监控(CPU load、disk usage、system logs)，也报过windows主机(使用NSClient++ plugin)

可以指定自己编写的Plugin通过网络收集数据来监控任何情况(温度、警告…)

可以通过配置Nagios 远程执行插件远程执行脚本

远程监控支持SSH或SSL加通道方式进行监控

简单的plugin设计允许用户很容易的开发自己需要的检查服务，支持很多开发语言(shell scripts、c++、Perl、ruby、Python、PHP、C#)

包含很多图形化数据Plugins（Nagiosgraph、Nagiosgrapher、PNP4Nagios等）

可并行服务检查

能够定义网络主机的层次，允许逐级检查，就是从父主机开始向下检查

当服务或主机出现问题时发出通告，可通过email, pager, sms 或任意用户自定义的plugin进行通知

能够自定义事件处理机制重新激活出问题的服务或主机

自动日志循环

支持冗余监控

包括Web界面可以查看当前网络状态，通知，问题历史，日志文件等

### nagios工作原理

Nagios 通过NRPE来远端管理服务

1. Nagios 执行安装在它里面的check\_nrpe 插件，并告诉check\_nrpe去检测那些服务
2. 通过SSL，check\_nrpe连接远端机子上的NRPE daemon
3. NRPE运行本地的各种插件去检测本地的服务和状态(check\_disk,..etc)
4. 最后,NRPE 把检测的结果传给主机端的check\_nrpe,check\_nrpe再把结果送到Nagios状态队列中
5. Nagios 依次读取队列中的信息,再把结果显示出来

为了需要 先关闭防火墙 /iptables

SELINUX=disabled

### ngios 服务端安装

#### 主程序安装

gcc glibc glibc-common gd gd-devel xinetd openssl-devel

rpm -q gcc glibc glibc-common gd gd-devel xinetd openssl-devel

如果没有安装 就

yum install -y gcc glibc glibc-common gd gd-devel xinetd openssl-devel

创建nagios用户和用户组

useradd -s /sbin/nologin nagios //表示不能通过这个用户登录

mkdir /usr/local/nagios

chown -R nagios.nagios /usr/local/nagios/

tar -zxf nagios-4.2.0.tar.gz 地址[http://prdownloads.sourceforge.net/sourceforge/nagios](http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-3.4.3.tar.gz)

./configure --prefix=/usr/local/nagios/

make all

unzip: command not found

解决：yum install unzip -y

make install

make install-init

make install-commandmode

make install-config

//加入服务

chkconfig --add nagios

chkconfig --level 35 nagios on //表示系统在3,5,的启动级别是程序都启动

chkconfig --list //查看启动情况

进入到/usr/local/nagios

目录结构

|  |  |
| --- | --- |
| nin | Nagios 可执行程序所在目录 |
| etc | Nagios 配置文件所在目录 |
| sbin | Nagios CGI 文件所在目录，也就是执行外部命令所需文件所在的目录 |
| share | Nagios网页文件所在的目录 |
| libexec | Nagios 外部插件所在目录 |
| var | Nagios 日志文件、lock 等文件所在的目录 |
| var/archives | Nagios 日志自动归档目录 |
| var/rw | 用来存放外部命令文件的目录 |

#### 插件安装

地址<https://www.nagios.org/downloads/nagios-plugins/>

nagios-plugins-2.0.3.tar.gz

yum install –y gd-devel

./configure --prefix=/usr/local/nagios

make && make install

#### 配置apache

User nagios

Group Nagios

上面主要是为了创建一个专用的用户和组 来运行httpd这个服务针对大多数据服务

ScriptAlias /nagios/cgi-bin "/usr/local/nagios/sbin"

<Directory "/usr/local/nagios/sbin">

AuthType Basic

Options ExecCGI

AllowOverride None

Order allow,deny

Allow from all

satisfy any

AuthName "Nagios Access"

AuthUserFile /usr/local/nagios/etc/htpasswd

Require valid-user

</Directory>

Alias /nagios "/usr/local/nagios/share"

<Directory "/usr/local/nagios/share">

AuthType Basic

Options none

AllowOverride None

Order allow,deny

Allow from all

AuthName "Nagios Access"

AuthUserFile /usr/local/nagios/etc/htpasswd

Require valid-user

</Directory>

/usr/local/apached2/bin/htpasswd -c /usr/local/nagios/etc/htpasswd admin

创建一个htpasswd验证文件 这个时候通过192.168.118.129/Nagios/ 访问时需要输入用户名和密码

启动nagios 服务

service nagios start

error while loading shared libraries: libltdl.so.3: cannot open shared object file: No such file or directory

这个是缺少libtool-ltdl

可以源码安装 也可以直接yum install libtool-ltdl

出现错误

You don't have permission to access /nagios/ on this server

解决：DirectoryIndex index.html index.php

#### 配置Nagios

Nagios 主要用于监控一台或者多台本地主机及远程的各种信息，包括本机资源及对外的服务等。默认的Nagios 配置没有任何监控内容，仅是一些模板文件。若要让Nagios提供服务，就必须修改配置文件，增加要监控的主机和服务

##### 默认配置文件介绍

Nagios安装完毕后，默认的配置文件在/usr/local/nagios/etc

[root@localhost etc]# pwd

/usr/local/nagios/etc

[root@localhost etc]# ls

cgi.cfg htpasswd nagios.cfg objects resource.cfg

每个文件或目录含义如下表所示：

|  |  |
| --- | --- |
| **文件名或目录名** | **用途** |
| cgi.cfg | 控制CGI访问的配置文件 |
| nagios.cfg | Nagios 主配置文件 |
| resource.cfg | 变量定义文件，又称为资源文件，在些文件中定义变量，以便由其他配置文件引用，如$USER1$ |
| objects | objects 是一个目录，在此目录下有很多配置文件模板，用于定义Nagios 对象 |
| objects/commands.cfg | 命令定义配置文件，其中定义的命令可以被其他配置文件引用 |
| objects/contacts.cfg | 定义联系人和联系人组的配置文件 |
| objects/localhost.cfg | 定义监控本地主机的配置文件 |
| objects/printer.cfg | 定义监控打印机的一个配置文件模板，默认没有启用此文件 |
| objects/switch.cfg | 定义监控路由器的一个配置文件模板，默认没有启用此文件 |
| objects/templates.cfg | 定义主机和服务的一个模板配置文件，可以在其他配置文件中引用 |
| objects/timeperiods.cfg | 定义Nagios 监控时间段的配置文件 |
| objects/windows.cfg | 监控Windows 主机的一个配置文件模板，默认没有启用此文件 |

##### 配置文件之间的关系

Nagios的配置过程中涉及到几个定义：主机、主机组、服务、服务组、联系人、联系人组，监控时间，监控命令 各个配置文件之间是互为关联的，彼此引用，所以要弄清楚配置文件之间依赖与被依赖的关系 ，最重要的有四点：

第一：定义监控那些主机、主机组、服务和服务组；

第二：定义这个监控要用什么命令实现；

第三：定义监控的时间段；

第四：定义主机或服务出现问题时要通知的联系人和联系人组

##### 配置Nagios

为了能更清楚的说明问题，同时也为了维护方便，建议将nagios各个定义对象创建独立的配置文件

创建hosts.cfg文件来定义主机和主机组

创建remotehosts.cfg文件定义服务和服务组

用默认的contacts.cfg文件来定义联系人和联系人组

用默认commands.cfg文件来定义命令

用默认的timeperiods.cfg来定义监控时间段

用默认的templates.cfg文件作为资源引用文件

templates.cfg文件

Nagios主要用于监控主机资源以及服务，在nagios配置中称为对象，为了不必重复定义一些监控对象，nagios引入了一个模板配置文件，将一些共性的属性定义成模板，以便于多次引用，这就是templates.cfg的作用

##### 验证nagios配置文件的正确性

Nagios在验证配置文件方面做的非常道为，只需通过一个命令即可完成

/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

问题汇总

1. he requested URL /nagios/cgi-bin/extinfo.cgi was not found on this server.

这个可能存在问题是：

1. 在httpd.conf配置有问题

以下是标准配置

ScriptAlias /nagios/cgi-bin "/usr/local/nagios/sbin"

<Directory "/usr/local/nagios/sbin">

AuthType Basic

Options ExecCGI

AllowOverride None

Order allow,deny

Allow from all

satisfy any

AuthName "Nagios Access"

AuthUserFile /usr/local/nagios/etc/htpasswd

Require valid-user

</Directory>

Alias /nagios "/usr/local/nagios/share"

<Directory "/usr/local/nagios/share">

AuthType Basic

Options none

AllowOverride None

Order allow,deny

Allow from all

AuthName "Nagios Access"

AuthUserFile /usr/local/nagios/etc/htpasswd

Require valid-user

</Directory>

1. 在配置文件cgi.cfg

use\_authentication=1

default\_user\_name=admin

authorized\_for\_system\_information=nagiosadmin,admin

authorized\_for\_configuration\_information=nagiosadmin,admin

authorized\_for\_system\_commands=admin

authorized\_for\_all\_services=nagiosadmin,admin

authorized\_for\_all\_hosts=nagiosadmin,admin

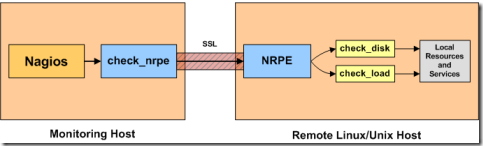
authorized\_for\_all\_service\_commands=nagiosadmin,admin

authorized\_for\_all\_host\_commands=nagiosadmin,admin

### 远端环境搭建（初步就打算在终端上搭建）

利用NRPE监控远程linux上的”本地信息”

NRPE工作原理



NRPE总共由两部份组成：

check\_nrpe插件，位于监控主机上

NRPE daemon，运行在远程的linux主机上(通常就是被监控机)

当Nagios 需要监控某个远程Linux 主机的服务或者资源情况时：

1. Nagios 会运行check\_nrpe 这个插件，告诉它要检查什么；
2. check\_nrpe 插件会连接到远程的NRPE daemon，所用的方式是SSL(不一定使用)；
3. NRPE daemon 会运行相应的Nagios 插件来执行检查；
4. NRPE daemon 将检查的结果返回给check\_nrpe 插件，插件将其递交给nagios做处理。

注意：NRPE daemon 需要Nagios 插件安装在远程的Linux主机上，否则，daemon不能做任何的监控。

#### 编译终端上nagios-plugins

./configure CC=arm-linux-gcc CXX=arm-linux-g++ LD=arm-linux-ld AR=arm-linux-ar AS=arm-linux-as --host=arm-linux --prefix=/usr/local/nagios

make&&make install

编译终端上的nrpe

./configure CC=arm-linux-gcc CXX=arm-linux-g++ LD=arm-linux-ld AR=arm-linux-ar AS=arm-linux-as --host=arm-linux --enable-ssl=no --enable-command-args --with-nrpe-user=root --with-nrpe-group=root

make all

make install-daemon  
make install-daemon-config

直接将/usr/local/nagios 拷贝到终端就好

xinetd 安装

centos7 上 启动服务 systemctl restart xinetd

需要理解xinetd 和 /etc/services 作用

如果没有xinetd可以配置inetd.conf

在 inetd.conf中需要些的内容是

终端配置

在配置文件/etc/inetd.conf

nrpe stream tcp nowait root /usr/local/nagios/bin/nrpe nrpe -c /usr/local/nagios/etc/nrpe.cfg --inetd

在配置文件/etc/services

nrpe 5666/tcp #nrpe

在/usr/local/Nagios/etc/nrpe.cfg配置相关内容

远程需要使用ftp命令使用终端需要配置

etc/bftpd.conf

远程可以ftp到终端需要的配置

ROOTDIR="/" 更改进入目录

下面是为了允许root用户ftp登录

#user root {

# DENY\_LOGIN="Root login not allowed."

#}

#### 在nagios服务端上的需求配置有nrpe

./configure --with-nrpe-user=root --with-nrpe-group=root --enable-ssl=no --enable-command-args

make all

make install-plugin

### 配置nagios的配置文件详解

[centos@yf objects]$ cat hosts.cfg

define host{

use linux-server

host\_name Nagios\_linux

alias Nagios\_linux

address 192.168.118.63

}

[centos@yf objects]$ cat remotehosts.cfg

define service{

use generic-service

host\_name Nagios\_linux

service\_description CPU Load

check\_command check\_nrpe!check\_load

}

define service{

use generic-service

host\_name Nagios\_linux

service\_description Current Users

check\_command check\_nrpe!check\_users

}

define service{

use generic-service

host\_name Nagios\_linux

service\_description Total Processes

check\_command check\_nrpe!check\_total\_procs

}

define service{

use generic-service

host\_name Nagios\_linux

service\_description Check Zombie Procs

check\_command check\_nrpe!check\_zombie\_procs

}

define service{

use generic-service

host\_name Nagios\_linux

service\_description Check swap

check\_command check\_nrpe!check\_swap

}

在"commands.cfg 中添加命令

#'check\_nrpe' command definition

define command{

command\_name check\_nrpe

command\_line $USER1$/check\_nrpe -H $HOSTADDRESS$ -c $ARG1$

}

在nagios.cfg中加入这两个

cfg\_file=/usr/local/nagios/etc/objects/hosts.cfg

cfg\_file=/usr/local/nagios/etc/objects/remotehosts.cfg

在cgi.cfg 这个文件是配置用户权限的

default\_user\_name=admin

authorized\_for\_system\_information=nagiosadmin,admin

authorized\_for\_configuration\_information=nagiosadmin,admin

authorized\_for\_system\_commands=nagiosadmin,admin

authorized\_for\_all\_services=nagiosadmin,admin

authorized\_for\_all\_hosts=nagiosadmin,admin

authorized\_for\_all\_service\_commands=nagiosadmin,admin

authorized\_for\_all\_host\_commands=nagiosadmin,admin

验证nagios的正确性

/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

### 后续问题

#### nrpe可以传参数

如果需要nrpe进行参数传递需要主要一下几个方面：  
1、编译nrpe的时候需要增加参数--enable-command-args  
2、修改nrpe.cfg文件中参数dont\_blame\_nrpe=1  
3、command.cfg文件中命令参数的定义为：  
command\_line $USER1$/check\_nrpe -H $HOSTADDRESS$ -c $ARG1$ -a $ARG2$  
  
4、services定义 check\_nrpe!命令！参数

针对使用让nagios 服务端直接传参数需要的一些改动 需要注意

首先编译的时候带上--enable-command-args

接着在被监控端需要在nrpe.cfg中dont\_blame\_nrpe设为1

以及相关命令设置

command[check\_users]=/usr/local/nagios/libexec/check\_users $ARG1$

在服务端配置需要

define command{

command\_name check\_nrpe

command\_line $USER1$/check\_nrpe -H $HOSTADDRESS$ -c $ARG1$ -a $ARG2$

}

增加-a $ARG2$

define service{

use generic-service

host\_name Nagios\_linux

service\_description CPU Load

check\_command check\_nrpe!check\_load!"-w 15,10,5 -c 30,25,20"

}

#### 配置发送邮件

##### 安装sendmail服务

yum install -y sendmail

yum install -y sendmail-cf

yum install mailx -y

启动服务

service sendmail start

测试邮件

echo "hello"|mail -s test test@qq.com

##### nagios配置文件

