Práctica BRS
Monitorización de hosts, servizos e redes: Nagios

ESCENARIO

Máquinas virtuais:
RAM ≤ 2048MB CPU ≤ 2 PAE/NX habilitado
Rede: 192.168.120.0/24
Disco duro dinámico: 20GB →/dev/sda1 →persistence
BIOS: Permite arranque dispositivo extraíble: CD/DVD, USB

Máquina virtual A:
Rede: NAT(eth0) + Interna(eth1)

Máquina virtual B:
Rede: NAT(eth0) + Interna(eth1)

Servidor Web: Apache (apache2)
Servidor Nagios + Plugins Nagios

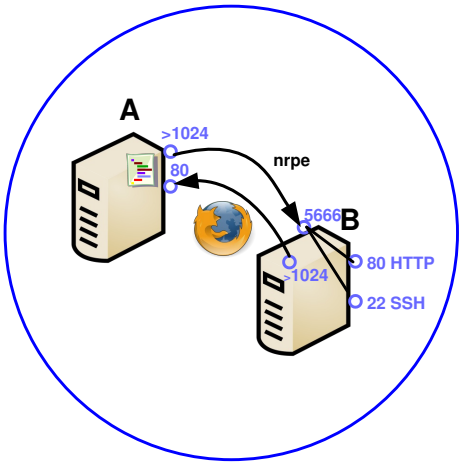
HTTP
80

Cliente Web: Navegador (firefox)
Plugins Nagios

Cliente NRPE: nagios-nrpe-plugin

nrpe
5666

Servidor NRPE: nagios-nrpe-server (5666)
- Servidor Web (apache2)
- Servidor SSH (openssh-server)

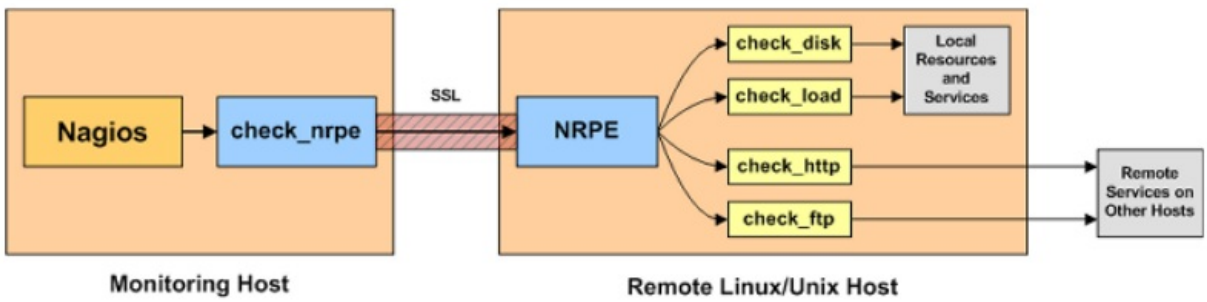


ISO: Kali Live amd64
IP/MS(eth1): 192.168.120.100/24

ISO: Kali Live amd64
IP/MS(eth1): 192.168.120.101/24

NOTAS:

- Prerrequisito: **HTTP Basic: Apache**
- **[1] Linux Monitoring With Nagios**
- **[2] Documentación Nagios**
- **[3] Debian Easy Guide Installation**
- **[4] NRPE**



Máquina virtual A: Kali amd64

1. Na contorna gráfica abrir un terminal e executar:
   kali@kali:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   kali@kali:~$ passwd kali #Cambiar o contrasinal do usuario kali. Por como contrasinal **abc123.** (Ollo que o contrasinal ten un caracter punto final).

2. Crear persistencia na Live de Kali:
   kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
      root@kali:~# parted --script /dev/sda mklabel msdos;done #Crear a etiqueta de disco (táboa de particións) ao dispositivo /dev/sda sen ter que acceder ao prompt de parted
      root@kali:~# parted --script /dev/sda mkpart primary 0 100% -a cylinder;done #Crear unha partición primaria no disco que ocupe todo o espazo posible, alineando a cilindros, sen ter que acceder ao prompt de parted
      root@kali:~# mkfs.ext4 -L 'persistence' /dev/sda1 #Formatear en ext4 coa etiqueta **persistence** a partición /dev/sda1
      root@kali:~# mount -t auto /dev/sda1 /mnt #Montar a partición 1 do disco duro /dev/sda no directorio da live /mnt. Coa opción -t auto solicitamos ao comando mount a autodetección do sistema de ficheiros de montaxe.
      root@kali:~# echo "/ union" | tee /mnt/persistence.conf #Crea o ficheiro de configuración persistence.conf. Isto significa que calquera cambio que se faga (instalación de software, ficheiros no escritorio ou no home, configuracións do sistema) gardarase usando un sistema de ficheiros "union" (overlay) na partición de persistencia /dev/sda1.

      root@kali:~# cd && umount /mnt Desmontar o sistema de ficheiros montados
      root@kali:~# reboot #Reiniciar para no próximo arranque escoller a opción de persistencia.

3. Unha vez reiniciado escoller a primeira opción de arranque que poña **persistence**: Live System with USB persistence (check kali.org/prst)

4. Cambiar hostname da máquina virtual A. Por kaliA como hostname:
   kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
      root@kali:~# echo 'kaliA' > /etc/hostname #Indicar ao sistema o valor do hostname.
      root@kali:~# echo 'kernel.hostname=kaliA' >> /etc/sysctl.conf #Indicar ao kernel o valor do hostname.
      root@kali:~# sysctl -p #Activar o cambio de hostname sen ter que pechar sesión nin reiniciar
      root@kali:~# exit #Saír da consola local sudo na que estabamos a traballar para voltar á consola local de **kali**.
   kali@kali:~$ exit #Pechar o terminal saíndo da consola local do usuario **kali**.

5. Configurar a rede. Na contorna gráfica abrir un terminal e executar:
   kali@kaliA:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
      root@kaliA:~# /etc/init.d/avahi-daemon stop || systemctl stop avahi-daemon #Parar o demo avahi-daemon(control resolución de nomes) para poder configurar de forma manual a configuración de rede e non ter conflicto con este demo.
      root@kaliA:~# /etc/init.d/network-manager stop || pkill NetworkManager #Parar o demo network-manager(xestor de rede) ou o script NetworkManager (executado sen ser demo) para poder configurar de forma manual a configuración de rede e non ter conflicto con este xestor.
      root@kaliA:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina A, as tarxetas de redes: loopback(lo) e interna(eth1).
      root@kaliA:~# ip addr add 192.168.120.100/24 dev eth1 #Configurar a tarxeta de rede interna eth1, coa IP: 192.168.120.100 e máscara de subrede: 255.255.255.0.
      root@kaliA:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina A, as tarxetas de redes: loopback(lo) e interna(eth1).
      root@kaliA:~# ping -c4 192.168.120.100 #Comprobar mediante o comando ping a conectividade coa interface de rede local eth1

Máquina virtual B: Kali amd64

6. Na contorna gráfica abrir un terminal e executar:
   kali@kali:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.

7. Crear persistencia na Live de Kali:
   kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
   root@kali:~# parted --script /dev/sda mklabel msdos;done #Crear a etiqueta de disco (táboa de particións) ao dispositivo /dev/sda sen ter que acceder ao prompt de parted
   root@kali:~# parted --script /dev/sda mkpart primary 0 100% -a cylinder;done #Crear unha partición primaria no disco que ocupe todo o espazo posible, alineando a cilindros, sen ter que acceder ao prompt de parted
   root@kali:~# mkfs.ext4 -L 'persistence' /dev/sda1 #Formatear en ext4 coa etiqueta **persistence** a partición /dev/sda1
   root@kali:~# mount -t auto /dev/sda1 /mnt #Montar a partición 1 do disco duro /dev/sda no directorio da live /mnt. Coa opción -t auto solicitamos ao comando mount a autodetección do sistema de ficheiros de montaxe.
   root@kali:~# echo "/ union" | tee /mnt/persistence.conf #Crea o ficheiro de configuración persistence.conf. Isto significa que calquera cambio que se faga (instalación de software, ficheiros no escritorio ou no home, configuracións do sistema) gardarase usando un sistema de ficheiros "union" (overlay) na partición de persistencia /dev/sda1.

   root@kali:~# cd && umount /mnt Desmontar o sistema de ficheiros montados
   root@kali:~# reboot #Reiniciar para no próximo arranque escoller a opción de persistencia.

8. Unha vez reiniciado escoller a primeira opción de arranque que poña **persistence**: Live System with USB persistence (check kali.org/prst)

9. Configuración da rede. Na contorna gráfica abrir un terminal e executar:
   kali@kali:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
   root@kali:~# /etc/init.d/avahi-daemon stop || systemctl stop avahi-daemon #Parar o demo avahi-daemon(control resolución de nomes) para poder configurar de forma manual a configuración de rede e non ter conflicto con este demo.
   root@kali:~# /etc/init.d/network-manager stop || pkill NetworkManager #Parar o demo network-manager(xestor de rede) ou o script NetworkManager (executado sen ser demo) para poder configurar de forma manual a configuración de rede e non ter conflicto con este xestor.
   root@kali:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina B as tarxetas de redes: loopback(lo) e interna(eth1).
   root@kali:~# ip addr add 192.168.120.101/24 dev eth1 #Configurar a tarxeta de rede interna eth1, coa IP: 192.168.120.101 e máscara de subrede: 255.255.255.0.
   root@kali:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina B as tarxetas de redes: loopback(lo) e interna(eth1).
   root@kali:~# ping -c4 192.168.120.101 #Comprobar mediante o comando ping a conectividade coa interface de rede local eth1
   root@kali:~# ping -c4 192.168.120.100 #Comprobar mediante o comando ping a conectividade coa interface de rede da máquina virtual A
   root@kali:~# echo '192.168.120.100 kaliA' >> /etc/hosts #Engadir no ficheiro /etc/hosts, é dicir, na táboa estática de búsqueda para nomes de host (DNS) o nome kaliA, para que atenda á IP 192.168.120.100
   root@kali:~# ping -c4 kaliA #Comprobar mediante o comando ping a conectividade coa NIC da máquina virtual A

10. Cambiar hostname da máquina virtual B. Por kaliB como hostname:
    root@kali:~# echo 'kaliB' > /etc/hostname #Indicar ao sistema o valor do hostname.
    root@kali:~# echo 'kernel.hostname=kaliB' >> /etc/sysctl.conf #Indicar ao kernel o valor do hostname.
    root@kali:~# sysctl -p #Activar o cambio de hostname sen ter que pechar sesión nin reiniciar
    root@kali:~# exit #Saír da consola local sudo na que estabamos a traballar para voltar á consola local de **kali**.
    kali@kali:~$ exit #Pechar o terminal saíndo da consola local do usuario **kali**.

Nagios

11. **kaliA → Instalar Nagios (Motor GUI Web + Servidor Nagios)**: Realizar o procedemento descrito en [3]. Basicamente:

A. Procedemento instalación nagios-core (Motor GUI Web + Servidor Nagios)

kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

root@kaliA# apt-get update
root@kaliA# apt-get install -y autoconf gcc libc6 make wget unzip apache2 apache2-utils php \
libgd-dev libssl-dev
root@kaliA# cd /tmp
root@kaliA# wget -O nagioscore.tar.gz \
https://github.com/NagiosEnterprises/nagioscore/releases/download/nagios-4.5.9/nagios-4.5.9.tar.gz
root@kaliA# tar xzf nagioscore.tar.gz
root@kaliA# cd /tmp/nagioscore-nagios-4.5.9/
root@kaliA# ./configure --with-httpd-conf=/etc/apache2/sites-enabled
root@kaliA# make all
root@kaliA# make install-groups-users
root@kaliA# usermod -a -G nagios www-data
root@kaliA# make install
root@kaliA# make install-daemoninit
root@kaliA# make install-commandmode
root@kaliA# make install-config
root@kaliA# make install-webconf
root@kaliA# a2enmod rewrite
root@kaliA# a2enmod cgi
root@kaliA# htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
root@kaliA# systemctl restart apache2.service
root@kaliA# systemctl status apache2.service --no-pager
root@kaliA# systemctl start nagios.service
root@kaliA# systemctl status nagios.service --no-pager

NOTA: No comando htpasswd solicítase un contrasinal para o usuario ***nagiosadmin***. Imos pór como contrasinal ***abc123.*** (Ollo que o contrasinal ten un caracter punto final)

Comandos

```
apt-get update
apt-get install -y autoconf gcc libc6 make wget unzip apache2 apache2-utils php libgd-dev libssl-dev
cd /tmp
wget -O nagioscore.tar.gz \
https://github.com/NagiosEnterprises/nagioscore/releases/download/nagios-4.5.9/nagios-4.5.9.tar.gz
tar xzf nagioscore.tar.gz
cd /tmp/nagioscore-nagios-4.5.9/
./configure --with-httpd-conf=/etc/apache2/sites-enabled
make all
make install-groups-users
usermod -a -G nagios www-data
make install
make install-daemoninit
make install-commandmode
make install-config
make install-webconf
a2enmod rewrite
a2enmod cgi
htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
systemctl restart apache2.service
systemctl status apache2.service --no-pager
systemctl start nagios.service
systemctl status nagios.service --no-pager
```

B. Procedemento instalación nagios-plugins (plugin → permite monitorizar recursos)

root@kaliA# apt-get install -y autoconf gcc libc6 libmcrypt-dev make libssl-dev wget bc gawk dc \
build-essential snmp libnet-snmp-perl gettext
root@kaliA# cd /tmp
root@kaliA# wget --no-check-certificate -O nagios-plugins.tar.gz \
https://github.com/nagios-plugins/nagios-plugins/releases/download/release-2.4.12/nagios-plugins-2.4.12.tar.gz
root@kaliA# tar zxf nagios-plugins.tar.gz
root@kaliA# cd /tmp/nagios-plugins-release-2.4.12/
root@kaliA# ./configure
root@kaliA# make
root@kaliA# make install
root@kaliA# systemctl restart nagios.service
root@kaliA# exit
kali@kaliA:~$

Comandos

```
apt-get install -y autoconf gcc libc6 libmcrypt-dev make libssl-dev wget bc gawk dc build-essential snmp
apt-get install -y libnet-snmp-perl gettext
cd /tmp
wget --no-check-certificate -O nagios-plugins.tar.gz \
https://github.com/nagios-plugins/nagios-plugins/releases/download/release-2.4.12/nagios-plugins-2.4.12.tar.gz
tar zxf nagios-plugins.tar.gz
cd /tmp/nagios-plugins-release-2.4.12/
./configure
make
make install
systemctl restart nagios.service
exit
```

Verificar

12. **kaliB → URL http://kaliA/nagios**: Lanzar na máquina virtual B (Kali) un navegador e visitar a URL http://kaliA/nagios

    - Acceder coas credenciais xeradas anteriormente: **nagiosadmin/abc123.**
    - Menú lateral esquerdo → Hosts
    - Menú lateral esquerdo → Services

13. **kaliB → Instalar o servidor nagios NRPE[4]**: Permite executar nagios-plugin nos hosts clientes a monitorizar por Nagios

    kali@kaliB:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

    root@kaliB:~# apt-get update
    root@kaliB:~# apt-get install -y nagios-nrpe-server
    root@kaliB:~# sed -i -e 's/allowed_hosts=127.0.0.1,::1/allowed_hosts=127.0.0.1,::1,192.168.120.100/' \
    -e 's/hda1/sda1/g' /etc/nagios/nrpe.cfg
    root@kaliB:~# /etc/init.d/nagios-nrpe-server status
    root@kaliB:~# /etc/init.d/nagios-nrpe-server start
    root@kaliB:~# /etc/init.d/nagios-nrpe-server status
    root@kaliB:~# netstat -natp | grep -i nrpe
    root@kaliB:~# ps -ef | grep -i [n]rpe

14. **kaliA → Instalar cliente NRPE no Servidor Nagios[4]**: Plugin NRPE para poder executar comandos nos clientes Nagios dende o servidor Nagios

    kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

    root@kaliA# apt-get update
    root@kaliA# apt-get install -y nagios-nrpe-plugin
    root@kaliA# /usr/lib/nagios/plugins/check_nrpe -H 192.168.120.101
    NRPE v4.1.3
    root@kaliA# cat > /tmp/command-nrpe.txt <<EOF

```
####### NRPE #######
define command {
    command_name    check_nrpe
    command_line    \$USER1\$/check_nrpe -H \$HOSTADDRESS\$ -c \$ARG1\$
}
EOF
```

    root@kaliA# cat /tmp/command-nrpe.txt >>/usr/local/nagios/etc/objects/commands.cfg
    root@kaliA# cat -n /usr/local/nagios/etc/objects/commands.cfg| tail -5

```
   254  ####### NRPE #######
   255  define command {
   256      command_name check_nrpe
   257      command_line $USER1$/check_nrpe -H $HOSTADDRESS$ -c $ARG1$
   258  }
```

    root@kaliA# cd /usr/local/nagios/etc/objects/

```
root@kaliA# cat > kaliB.cfg <<EOF
define host{
 use linux-server
 host_name kaliB
 alias cliente
 address 192.168.120.101
}

define service{
 use generic-service
 host_name kaliB
 service_description CPU Load
 check_command check_nrpe!check_load
}

define service{
 use generic-service
 host_name kaliB
 service_description Current Users
 check_command check_nrpe!check_users
}

define service{
 use generic-service
 host_name kaliB
 service_description /dev/sda1 Free Space
 check_command check_nrpe!check_sda1
}
define service{
 use generic-service
 host_name kaliB
 service_description Total Processes
 check_command check_nrpe!check_total_procs
}

define service{
 use generic-service
 host_name kaliB
 service_description Zombies Processes
 check_command check_nrpe!check_zombie_procs
}
EOF

root@kaliA# echo 'cfg_file=/usr/local/nagios/etc/objects/kaliB.cfg' >> /usr/local/nagios/etc/nagios.cfg
root@kaliA# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
root@kaliA# ln -s /usr/lib/nagios/plugins/check_nrpe /usr/local/nagios/libexec/check_nrpe
root@kaliA# systemctl restart nagios.service
root@kaliA# systemctl status nagios.service --no-pager
```

```
root@kaliA# for i in $(grep check kaliB.cfg | awk -F'!' '{print $NF}' | xargs)
do
/usr/lib/nagios/plugins/check_nrpe -H 192.168.120.101 -c $i
done
```

```
LOAD WARNING - scaled load average: 0.13, 0.09, 0.13 - total load average:
0.26, 0.17, 0.26|load1=0.260;;;0; scaled_load1=0.130;0.150;0.300;0;
load5=0.170;;;0; scaled_load5=0.085;0.100;0.250;0; load15=0.260;;;0;
scaled_load15=0.130;0.050;0.200;0;
USERS OK - 2 users currently logged in |users=2;5;10;0
DISK OK - free space: /run/live/persistence/sda1 18510MiB (97.7%
inode=100%);|
/run/live/persistence/sda1=457179136B;16765471948;18861155942;0;20956839936
PROCS WARNING: 199 processes | procs=199;150;200;0;
PROCS OK: 0 processes with STATE = Z | procs=0;5;10;0;
```

Verificar

15. **kaliB → URL http://kaliA/nagios**: Actualizar na máquina virtual B (Kali) a páxina referente á URL http://kaliA/nagios
    - Menú lateral esquerdo → Hosts
    - Menú lateral esquerdo → Services

16. **kaliB → Agregar servizos a chequear mediante NRPE[4]**:

kali@kaliB:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

```
root@kaliB:~# echo 'command[check_http]=/usr/lib/nagios/plugins/check_http -H 127.0.0.1 -w 5 -c 10' \
>> /etc/nagios/nrpe.cfg
root@kaliB:~# echo 'command[check_ssh]=/usr/lib/nagios/plugins/check_ssh -H 127.0.0.1' \
>> /etc/nagios/nrpe.cfg
root@kaliB:~# /etc/init.d/nagios-nrpe-server restart
root@kaliB:~# /etc/init.d/nagios-nrpe-server status
```

17. **kaliA → Chequear os anteriores servizos mediante NRPE no Servidor Nagios[4]**:

kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

```
root@kaliA# cd /usr/local/nagios/etc/objects/
root@kaliA# cat >> kaliB.cfg <<EOF

define service{
 use generic-service
 host_name kaliB
 service_description Check HTTP Service
 check_command check_nrpe!check_http
}

define service{
 use generic-service
 host_name kaliB
 service_description Check SSH Service
 check_command check_nrpe!check_ssh
}
EOF
root@kaliA# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
root@kaliA# systemctl restart nagios.service
root@kaliA# systemctl status nagios.service --no-pager
```

Verificar

18. **kaliB → URL http://kaliA/nagios**: Actualizar na máquina virtual B (Kali) a páxina referente á URL http://kaliA/nagios
    - Menú lateral esquerdo → Hosts
    - Menú lateral esquerdo → Services

19. **kaliB → Activar Servidor HTTP (Apache)**:
    kali@kaliB:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
    root@kaliB:~# nc -vz localhost 80 ; [$(echo $?) -ne 0] && /etc/init.d/apache2 start #Verificar se o servizo HTTP(Apache) está activo. Se non está activo, arráncase
    root@kaliB:~# /etc/init.d/apache2 status #Comprobar o estado do servidor HTTP(Apache), agora debe estar arrancado.
    root@kaliB:~# exit #Saír da consola local sudo na que estabamos a traballar para voltar á consola local de **kali**.
    kali@kaliB:~$

20. **kaliB → Activar Servidor SSH**:
    kali@kaliB:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
    root@kaliB:~# nc -vz localhost 22 ; [$(echo $?) -ne 0] && /etc/init.d/ssh start #Verificar se o servizo SSH está activo. Se non está activo, arráncase
    root@kaliB:~# /etc/init.d/ssh status #Comprobar o estado do servidor SSH, agora debe estar arrancado.
    root@kaliB:~# exit #Saír da consola local sudo na que estabamos a traballar para voltar á consola local de **kali**.
    kali@kaliB:~$

21. **kaliB → URL http://kaliA/nagios**: Actualizar na máquina virtual B (Kali) a páxina referente á URL http://kaliA/nagios
    - Menú lateral esquerdo → Hosts
    - Menú lateral esquerdo → Services

Ricardo Feijoo Costa