

Para que o Nagios funcione é necessário que existam máquinas a serem monitoradas na rede e para que essas máquinas possam ser monitoradas é preciso realizarmos algumas configurações.

O primeiro passo é instalar o serviço NRPE e o suporte a plugins do Nagios.

```
ubuntu@ubuntuserver:~$ sudo apt install nagios-nrpe-server nagios-plugins
```

Depois adicione o IP do seu servidor ao arquivo de configuração do **nrpe**:  
“/etc/nagios/nrpe.cfg”

```
GNU nano 2.5.3      Arquivo: /etc/nagios/nrpe.cfg

pid_file=/var/run/nagios/nrpe.pid

# PORT NUMBER
# Port number we should wait for connections on.
# NOTE: This must be a non-privileged port (i.e. > 1024).
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

server_port=5666

# SERVER ADDRESS
# Address that nrpe should bind to in case there are more than one interface
# and you do not want nrpe to bind on all interfaces.
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

server_address=192.168.1.10

# NRPE USER
# This determines the effective user that the NRPE daemon should run as.
# You can either supply a username or a UID.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

nrpe_user=nagios

# NRPE GROUP
# This determines the effective group that the NRPE daemon should run as.
# You can either supply a group name or a GID.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

nrpe_group=nagios

# ALLOWED HOST ADDRESSES
```

O próximo passo é configurar o servidor Nagios para monitorar o host que configuramos.

Conectados ao servidor vamos ao diretório “**/usr/local/nagios/etc/servers/**“, caso não exista criamos o diretório.

```
ubuntu@ubuntu-server:~$ sudo mkdir -p /usr/local/nagios/etc/servers
ubuntu@ubuntu-server:~$
```

Em seguida criamos o arquivo de configuração do host no diretório **servers**.

```
ubuntu@ubuntu-server:~$ sudo nano -c /usr/local/nagios/etc/servers/ubuntu_host.cfg
```

Adicionamos as seguintes linhas ao arquivo:

# Ubuntu Host configuration file

```
define host {
    use                linux-server
    host_name          ubuntu_host
    alias              Ubuntu Host
    address             192.168.1.10
    register            1
}

define service {
    host_name          ubuntu_host
    service_description PING
    check_command       check_ping!100.0,20%!500.0,60%
    max_check_attempts 2
    check_interval      2
    retry_interval      2
    check_period        24x7
    check_freshness     1
    contact_groups      admins
    notification_interval 2
    notification_period 24x7
    notifications_enabled 1
    register            1
}

define service {
    host_name          ubuntu_host
    service_description Check Users
    check_command       check_local_users!20!50
    max_check_attempts 2
    check_interval      2
    retry_interval      2
    check_period        24x7
    check_freshness     1
    contact_groups      admins
    notification_interval 2
}
```

```

        notification_period      24x7
        notifications_enabled     1
        register                  1
    }

define service {
    host_name          ubuntu_host
    service_description Local Disk
    check_command       check_local_disk!20%!10%!/
    max_check_attempts  2
    check_interval      2
    retry_interval      2
    check_period        24x7
    check_freshness     1
    contact_groups      admins
    notification_interval 2
    notification_period  24x7
    notifications_enabled 1
    register             1
}

define service {
    host_name          ubuntu_host
    service_description Check SSH
    check_command       check_ssh
    max_check_attempts  2
    check_interval      2
    retry_interval      2
    check_period        24x7
    check_freshness     1
    contact_groups      admins
    notification_interval 2
    notification_period  24x7
    notifications_enabled 1
    register             1
}

define service {
    host_name          ubuntu_host
    service_description Total Process
    check_command       check_local_procs!250!400!RSZDT
    max_check_attempts  2
    check_interval      2
    retry_interval      2
    check_period        24x7
    check_freshness     1
    contact_groups      admins
    notification_interval 2
    notification_period  24x7
    notifications_enabled 1
    register             1
}

```

O arquivo “**usr/local/nagios/etc/objects/commands.cfg**” tem exemplos de outros serviços como DHCP ou POP que podem ser adicionados ao arquivo que configuramos. Basta copiar as linhas do referido serviço para o arquivo que criamos.

Abaixo demonstramos como testar se a configuração está correta e o retorno esperado:

```
ubuntu@ubuntu-server:~$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.2.0
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 08-01-2016
License: GPL

Website: https://www.nagios.org
Reading configuration data...
  Read main config file okay...
  Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
  Checked 13 services.
  Checked 2 hosts.
  Checked 1 host groups.
  Checked 0 service groups.
  Checked 1 contacts.
  Checked 1 contact groups.
  Checked 24 commands.
  Checked 5 time periods.
  Checked 0 host escalations.
  Checked 0 service escalations.
Checking for circular paths...
  Checked 2 hosts
  Checked 0 service dependencies
  Checked 0 host dependencies
  Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors: 0

Things look okay - No serious problems were detected during the pre-flight check
ubuntu@ubuntu-server:~$
```

Por fim reiniciamos todos os serviços. No servidor Nagios reiniciaremos o “**apache2**” e o “**nagios**” e no host monitorado reiniciaremos o “**nagios-nrpe-server**”.

Pronto está funcionando.