

# Supervision Nagios SNMP

Documentation à lire uniquement si on a fait le choix d'utiliser SNMP au lieu des agents installés sur les machines. Sinon, se reporter à la documentation nommé [Supervision Nagios Agent](#).

## Convention de nom des communautés SNMP

- COMMUNAUTE\_SERVEUR : pour les serveurs Windows et Linux.
- COMMUNAUTE\_RESEAU : pour les routeurs (existants pour tout les routeurs), les switchs, les bornes wifi et les faisceau hertziens.
- COMMUNAUTE\_SECURITE : pour les éléments de la baie sécurité sauf la console de management.

## Installer les plugins SNMP sur le serveur

Afin de récolter les informations des systèmes, il est nécessaire d'utiliser des plugins supplémentaires plus précis. J'ai utilisé les plugins du site manubulon (<http://nagios.manubulon.com>). Un paquetage est à télécharger et à installer.

Installer les dépendances perl nécessaires.

```
yum install perl-Net-SNMP
```

Aller sur le site <http://nagios.manubulon.com/> pour récupérez la liste de scripts suivante disponible sur le site dans une archive.

```
check_snmp_boostedge.pl  
check_snmp_cpfw.pl  
check_snmp_css.pl  
check_snmp_env.pl  
check_snmp_int.pl  
check_snmp_linkproof_nhr.pl  
check_snmp_load.pl  
check_snmp_mem.pl  
check_snmp_nsbox.pl  
check_snmp_process.pl  
check_snmp_processus_loaded.pl  
check_snmp_script_result.pl  
check_snmp_storage.pl  
check_snmp_vrrp.pl  
check_snmp_win.pl
```

Positionner ces scripts dans le dossier `/usr/lib/nagios/plugins` et leurs appliquer les droits.

```
chmod 755 check_snmp_*
```

Vous remarquerez qu'un script check\_snmp est déjà présent. Cependant, il est un peu moins simple et précis à utiliser que les autres que nous venons d'installer. On va tout de même s'en servir pour l'uptime des systèmes par exemple.

Récupérer le script PERL nommé check\_snmp\_netint.pl développé par William Leibzon et disponible à cette adresse : <http://wleibzon.bol.ucla.edu/nagios/>. Il a été créé à partir du script de Patrick Proy. Le positionner aussi dans le dossier /usr/lib/nagios/plugins de manière manuelle.

```
cp /root/nagios-server/check_snmp_supp/check_snmp_netint.pl
/usr/lib/nagios/plugins
```

Lui appliquer les droits.

```
chmod 755 check_snmp_netint.pl
```

et il est opérationnel.

Une fois cette opération réalisée, il faut créer les commandes avec les paramètres qu'on passe pour que nous puissions utiliser ces commandes dans la définition des services.

Les paramètres entre des Windows, des Linux ou des ESX sont différents, pour contrer cela, une définition de commande par type de système a été créée. On obtient les commandes suivantes.

```
check_win_storage
check_win_load
check_win_mem
...

check_lin_storage
check_lin_load
check_lin_mem
...
```

```
#####
# Commandes ajoutées en SNMP et normales #
#####

##### Windows #####

define command{
    command_name check_win_storage
    command_line $USER1$/check_snmp_storage.pl -H $HOSTADDRESS$ -C
$ARG1$ $ARG2$ -m $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_win_load
    command_line $USER1$/check_snmp_load.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -T $ARG3$ -w $ARG4$ -c $ARG5$
```

```
}

define command{
    command_name check_win_mem
    command_line $USER1$/check_snmp_storage.pl -H $HOSTADDRESS$ -C
$ARG1$ $ARG2$ -m $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_win_int
    command_line $USER1$/check_snmp_netint.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -n $ARG3$ -a -m -k -M -w $ARG4$ -c $ARG5$
}

##### Linux #####
define command{
    command_name check_lin_storage
    command_line $USER1$/check_snmp_storage.pl -H $HOSTADDRESS$ -C
$ARG1$ $ARG2$ -m $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_lin_load
    command_line $USER1$/check_snmp_load.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -T $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_lin_mem
    command_line $USER1$/check_snmp_mem.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_lin_int
    command_line $USER1$/check_snmp_netint.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -n $ARG3$ -a -m -k -M -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_lin_procs
    command_line $USER1$/check_procs -u $ARG1$ -m $ARG2$ -w $ARG3$ -c
$ARG4$
}

##### ESX #####
define command{
    command_name check_esx_storage
    command_line $USER1$/check_snmp_storage.pl -H $HOSTADDRESS$ -C
$ARG1$ $ARG2$ -m $ARG3$ -w $ARG4$ -c $ARG5$
}
```

```
define command{
    command_name check_esx_load
    command_line $USER1$/check_snmp_load.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -T $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_esx_mem
    command_line $USER1$/check_snmp_mem.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_esx_int
    command_line $USER1$/check_snmp_netint.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -n $ARG3$ -a -m -k -M -w $ARG4$ -c $ARG5$
}

##### Network #####
define command{
    command_name check_net_int
    #command_line $USER1$/check_snmp_netint.pl -H $HOSTADDRESS$ -C
$ARG1$ $ARG2$ -n $ARG3$ -a -k -M -w $ARG4$ -c $ARG5$
    command_line $USER1$/check_snmp_netint.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -n $ARG3$ -a -k -y -M -w $ARG4$ -c $ARG5$
}
```

La convention de nommage que j'ai utilisée permet de facilement savoir si le système du serveur est Windows, Linux ou ESX par les trois lettres au milieu du nom de commande.

Vérifier la configuration pour éliminer les éventuelles erreurs.

```
nagios -v /etc/nagios/nagios.cfg
```

Redémarrer le service Nagios.

```
/etc/init.d/nagios restart
```

Accéder à l'interface web de nagios : <http://srv-supervision.domaine.local/nagios/>

## Méthode de création de commandes

Cf. documentation [creation\\_commandes\\_services](#).

# Installer SNMP sous Windows

Cf. documentation [installer\\_activer\\_snmp](#).

Valider et la réception d'information SNMP par Nagios commence directement au prochain check.

## Ajout d'un système Windows en SNMP

Au lieu d'utiliser NSClient++, nous pouvons utiliser SNMP pour récupérer les informations que nous voulons récolter.

Si le service SNMP n'est pas dans la liste des services du serveur, il faut l'installer comme expliqué dans la section précédente.

Ensuite, il faut créer les commandes dans le fichier de définition des commandes : `commands.cfg`. On choisit les arguments en fonction de ce qu'on va passer dans l'appel de la commande.

Je m'explique, si on veut avoir une valeur warning et critique, et qu'on passe les deux en paramètres lors de l'appel d'une commande, on doit intercepter le premier argument (`$ARG1$`) en le donnant à l'option `-w` (option des warnings) et le deuxième argument (`$ARG2`) en le donnant à l'option `-c` (option des critiques).

`templates.cfg`

```
# Windows host definition template - This is NOT a real host, just a
template!

define host{
    name                windows-server    ; The name of this host template
    use                  generic-host     ; Inherit default values from the
generic-host template
    check_period        24x7             ; By default, Windows servers are
monitored round the clock
    check_interval      3                ; Actively check the server every 3
minutes
    retry_interval      1                ; Schedule host check retries at 1 minute
intervals
    max_check_attempts  10               ; Check each server 10 times (max)
    check_command        check-host-alive ; Default command to check if
servers are "alive"
    notification_period 24x7             ; Send notification out at any time -
day or night
    notification_interval 0              ; Resend notifications every 5 hours
    notification_options d,r            ; Only send notifications for
specific host states
    contact_groups       admins          ; Notifications get sent to the
admins by default
```

```

    hostgroups      grp-win      ; Host groups that Windows servers
should be a member of
    register        0            ; DONT REGISTER THIS - ITS JUST A TEMPLATE
}

```

#### commands.cfg

```

##### Windows #####

define command{
    command_name check_win_storage
    command_line $USER1$/check_snmp_storage.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -m $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_win_load
    command_line $USER1$/check_snmp_load.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -T $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_win_mem
    command_line $USER1$/check_snmp_storage.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -m $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_win_int
    command_line $USER1$/check_snmp_netint.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -n $ARG3$ -a -m -k -M -w $ARG4$ -c $ARG5$
}

```

#### grp-srv.cfg

```

define hostgroup{
    hostgroup_name  grp-win      ; The name of the hostgroup
    alias          Global Windows Servers Group ; Long name of the group
    #This group contain all the other Windows groups
    hostgroup_members  grp-win-vm, grp-win-ph
}

define hostgroup{
    hostgroup_name  grp-win-vm   ; The name of the hostgroup
    alias          VM Windows Servers Group ; Long name of the group
}

define hostgroup{
    hostgroup_name  grp-win-ph   ; The name of the hostgroup
    alias          Physical Windows Servers Group ; Long name of the group
}

```

```
}
```

hst-win.cfg

```
define host{
    use                windows-server        ; Inherit default values from a template
    host_name          srv-w2k3-sup         ; The name we're giving to this host
    address             @IP                 ; IP address of the host
    hostgroups          grp-win-vm          ; Host belong group
    parents             win
}
```

Ensuite, il faut utiliser ces commandes dans les services. Extrait du fichier ser-win.cfg qui définit les services associés au serveurs Windows.

```
# Create a service for monitoring Windows C: disks with SNMP
define service{
    use                generic-service
    hostgroup_name     windows-servers
    service_description chk-win-disk
    check_command       check_win_storage!COMMUNAUTE_SERVEUR_srv!--
v2c!^[CDEFGHIJKLMNOPQRSTUVWXYZ]:!85!95
    servicegroups      ser-win-base
}

# Create a service for monitoring Windows CPU load with SNMP
define service{
    use                generic-service      ;Use generic-service
    template
    hostgroup_name     windows-servers      ;Apply this
    service to hostgroup
    service_description chk-win-load        ;Description of
    service
    check_command       check_win_load!COMMUNAUTE_SERVEUR_srv!--
v2c!stand!85!95 ;Command
    servicegroups      ser-win-base        ;Service belong
    servicegroup
}

# Create a service for monitoring Windows Memory (RAM physique + virtuelle)
load with SNMP
define service{
    use                generic-service ; Inherit values from a
    template
    hostgroup_name     windows-servers
    service_description chk-win-mem
    check_command       check_win_mem!COMMUNAUTE_SERVEUR_srv!--
v2c!"^Virtual Memory$"!85!99
    servicegroups      ser-win-base        ; Service belong
    servicegroup
}
```

```
# Create a service for monitoring Windows Uptime SNMP
define service{
    use                generic-service ; Inherit values from a
template
    hostgroup_name    windows-servers
    service_description  chk-win-up
    check_command      check_snmp!-C COMMUNAUTE_SERVEUR_srv -P 2c -
o sysUpTime.0
    servicegroups      ser-win-base      ; Service belong
servicegroup
}
```

Comme d'habitude, vérifier la configuration pour éliminer les éventuelles erreurs.

```
nagios -v /etc/nagios/nagios.cfg
```

Redémarrer le service Nagios.

```
/etc/init.d/nagios restart
```

Accéder à l'interface web de nagios : <http://srv-supervision.domaine.local/nagios/>

## Installer SNMP sous Linux

Cf. documentation [installer\\_activer\\_snmp](#).

## Ajout d'un système Linux en SNMP

Au lieu d'utiliser NRPE, nous pouvons utiliser SNMP pour récupérer les informations que nous voulons récolter.

templates.cfg

```
# Linux host definition template - This is NOT a real host, just a template!
define host{
    name                linux-server      ; The name of this host template
    use                generic-host      ; This template inherits other values
from the generic-host template
    check_period        24x7              ; By default, Linux hosts are
checked round the clock
    check_interval      3                 ; Actively check the host every 3
minutes
    retry_interval      1                 ; Schedule host check retries at 1
minute intervals
```



```

    max_check_attempts      10          ; Check each Linux host 10 times
(max)
    check_command            check-host-alive ; Default command to
check Linux hosts
    notification_period      workhours    ; Linux admins hate to be woken
up, so we only notify during the day
                                ; Note that the notification_period variable is
being overridden from
                                ; the value that is inherited from the generic-
host template!
    notification_interval    0           ; Resend notifications every 5
hours
    notification_options     d,u,r       ; Only send notifications for
specific host states
    contact_groups           admins      ; Notifications get sent to the
admins by default
    hostgroups               grp-lin     ; Host groups that Linux servers
should be a member of
    register                 0           ; DONT REGISTER THIS DEFINITION - ITS NOT A
REAL HOST, JUST A TEMPLATE!
}

```

#### commands.cfg

```

##### Linux #####

define command{
    command_name check_lin_storage
    command_line $USER1$/check_snmp_storage.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -m $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_lin_load
    command_line $USER1$/check_snmp_load.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -T $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_lin_mem
    command_line $USER1$/check_snmp_mem.pl -H $HOSTADDRESS$ -C $ARG1$ $ARG2$
$ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_lin_int
    command_line $USER1$/check_snmp_netint.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -n $ARG3$ -a -m -k -M -w $ARG4$ -c $ARG5$
}

define command{

```

```

        command_name check_lin_procs
        command_line $USER1$/check_procs -u $ARG1$ -m $ARG2$ -w $ARG3$ -c
$ARG4$
    }

```

#### grp-srv.cfg

```

define hostgroup{
    hostgroup_name grp-lin          ; The name of the hostgroup
    alias          Global Linux Servers Group      ; Long name of the
group
    #This group contain all the other Linux groups
    hostgroup_members  grp-lin-vm, grp-lin-ph
    }

define hostgroup{
    hostgroup_name  grp-lin-vm      ; The name of the hostgroup
    alias          VM Linux Servers Group      ; Long name of the group
    }

define hostgroup{
    hostgroup_name  grp-lin-ph      ; The name of the hostgroup
    alias          Physical Linux Servers Group      ; Long name of the
group
    }

```

#### hst-lin.cfg

```

define host{
    use          linux-server      ; Inherit default values from a template
    host_name    srv-lin-test      ; The name we're giving to this host
    address      @IP              ; IP address of the host
    hostgroups   grp-lin-vm        ; Host belong group
    parents      lin
    }

```

#### ser-lin.cfg

```

# Create a service for monitoring Linux partitions levels with SNMP
# Checks that "/", "/tmp", "/usr", "/var" mountpoints usage is < 85 and 95%
define service{
    use          generic-service    ;Use generic-service
template
    hostgroup_name  grp-lin          ;Apply this service to hostgroup
    service_description  chk-lin-disk      ;Description of service
    check_command   check_lin_storage!COMMUNAUTE_SERVEUR!--
v2c!"^/$|tmp|usr|var|data"!85!95      ;Command
    servicegroups   ser-lin-base      ;Service belong servicegroup
    }

```

```

# Create a service for monitoring Linux load (CPU + processus + disk charge
confondu) with SNMP
#3 values : load average on 1 min, 5 min, 15 min (absolute)
define service{
    use                generic-service                ;Use generic-service
template
    hostgroup_name    grp-lin                ;Apply this service to hostgroup
    service_description    chk-lin-load                ;Description of service
    check_command      check_lin_load!COMMUNAUTE_SERVEUR!--
v2c!netsl!2,1,1!3,2,2    ;Command
    servicegroups      ser-lin-base                ;Service belong servicegroup
}

# Create a service for monitoring Linux memory (RAM + Swap) with SNMP
# Linux_memory : warning at 95% memory used and 20% swap used, critical at
99% mem and 70% swap
define service {
    use                generic-service                ;Use generic-service
template
    hostgroup_name    grp-lin                ;Apply this service to
hostgroup
    service_description    chk-lin-mem                ;Description of
service
    check_command      check_lin_mem!COMMUNAUTE_SERVEUR!--v2c!-
N!95,20!99,70                ;Command
    servicegroups      ser-lin-base                ;Service belong servicegroup
}

define service{
    use                generic-service ; Inherit values from a
template
    hostgroup_name    grp-lin
    service_description    chk-lin-up
    check_command      check_snmp!-C COMMUNAUTE_SERVEUR -P 2c -o
sysUpTime.0
    servicegroups      ser-lin-base                ; Service belong
servicegroup
}

# Network interfaces checks
define service {
    use                generic-service                ;Use generic-service
template
    hostgroup_name    grp-lin-vm                ;Apply this service to
hostgroup
    service_description    chk-lin-int                ;Description of
service
    #warning to 15 Mo/s (input) and 15 Mo/s (output) and critical to 25 Mo/s
(input) and 25 Mo/s (output)
    check_command      check_lin_int!COMMUNAUTE_SERVEUR!--
v2c!"eth0"!15,15!25,25

```

```

    servicegroups      ser-lin-base      ;Service belong servicegroup
    }

define service {
    use                generic-service      ;Use generic-service
template
    hostgroup_name    grp-lin-ph          ;Apply this service to
hostgroup
    service_description  chk-lin-int      ;Description of
service
    #warning to 15 Mo/s (input) and 15 Mo/s (output) and critical to 25 Mo/s
(input) and 25 Mo/s (output)
    check_command      check_lin_int!COMMUNAUTE_SERVEUR! - -
v2c!"bond0"!15,15!25,25
    servicegroups      ser-lin-base      ;Service belong servicegroup
    }

#####
###
#####
###
#
# SERVICE GROUPS
#
#####
###
#####
###

define servicegroup{
    servicegroup_name  ser-lin-base
    alias              Supervision Linux de base
}

```

Comme d'habitude, vérifier la configuration pour éliminer les éventuelles erreurs.

```
nagios -v /etc/nagios/nagios.cfg
```

Redémarrer le service Nagios.

```
/etc/init.d/nagios restart
```

Accéder à l'interface web de nagios : <http://srv-supervision.domaine.local/nagios/>

# Installer SNMP sous ESX

Cf. documentation [installer\\_activer\\_snmp](#).

Redémarrer le service avec un `/etc/init.d/snmpd restart`

## Ajout d'un système ESX en SNMP

templates.cfg

```
# ESX host definition template - This is NOT a real host, just a template!

define host{
    name                esx-server        ; The name of this
host template
    use                 generic-host      ; This template
inherits other values from the generic-host template
    check_period        24x7              ; By default, Linux
hosts are checked round the clock
    check_interval      3                 ; Actively check the
host every 5 minutes
    retry_interval      1                 ; Schedule host
check retries at 1 minute intervals
    max_check_attempts  10                ; Check each Linux
host 10 times (max)
    check_command        check-host-alive ; Default command
to check Linux hosts
    notification_period workhours         ; ESX admins hate to
be woken up, so we only notify during the day
                                                ; Note that the
notification_period variable is being overridden from
                                                ; the value that is
inherited from the generic-host template!
    notification_interval 300             ; Resend
notifications every 5 hours
    notification_options d,u,r           ; Only send
notifications for specific host states
    contact_groups       admins          ; Notifications get
sent to the admins by default
    hostgroups           esx-servers     ; Host groups that
Linux servers should be a member of
    register              0               ; DONT REGISTER THIS
DEFINITION - ITS NOT A REAL HOST, JUST A TEMPLATE!
}
```

commands.cfg

```
##### ESX #####
define command{
    command_name check_esx_storage
    command_line $USER1$/check_snmp_storage.pl -H $HOSTADDRESS$ -C
$ARG1$ $ARG2$ -m $ARG3$ -w $ARG4$ -c $ARG5$
}
```

```

}

define command{
    command_name check_esx_load
    command_line $USER1$/check_snmp_load.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -T $ARG3$ -w $ARG4$ -c $ARG5$
}

define command{
    command_name check_esx_mem
    command_line $USER1$/check_snmp_mem.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ $ARG3$ -w $ARG4$ -c $ARG5$
}

```

### grp-srv.cfg

```

define hostgroup{
    hostgroup_name  esx-servers          ; The name of the hostgroup
    alias           VMware ESX Servers Group ; Long name of the
group
}

```

### hst-esx.cfg

```

define host{
    use           esx-server          ; Inherit default values from a
template
    host_name    srv-esx             ; The name we're giving to this host
    address      @IP                 ; IP address of the host
    hostgroups   esx-servers         ; Host belong group
}

```

### ser-esx.cfg

```

# Create a service for monitoring Linux partitions levels with SNMP
# Checks that "/", "/tmp", "/usr", "/var" mountpoints usage is < 85 and 95%
define service{
    use           generic-service     ;Use generic-service
template
    hostgroup_name  esx-servers       ;Apply this service
to hostgroup
    service_description  chk-esx-disk ;Description of
service
    check_command    check_esx_storage!COMMUNAUTE_SERVEUR!--
v2c!"^/$|var/log"!85!95 ;Command
    servicegroups   ser-esx-base     ;Service belong
servicegroup
}

# Create a service for monitoring Linux load (CPU + processus + disk charge
confondu) with SNMP

```

```

define service{
    use                generic-service        ;Use generic-service
template
    hostgroup_name    esx-servers            ;Apply this service
to hostgroup
    service_description  chk-esx-load        ;Description of
service
    check_command      check_esx_load!COMMUNAUTE_SERVEUR!--
v2c!netsl!4,3,3!8,5,5 ;Command
    servicegroups      ser-esx-base          ;Service belong
servicegroup
}

```

# Create a service for monitoring Linux memory (RAM + Swap) with SNMP  
# Linux\_memory : warning at 95% memory used and 20% swap used, critical at 99% mem and 70% swap

```

define service {
    use                generic-service        ;Use generic-service
template
    hostgroup_name    esx-servers            ;Apply this service to
hostgroup
    service_description  chk-esx-mem        ;Description of
service
    check_command      check_esx_mem!COMMUNAUTE_SERVEUR!--v2c!-N!95,20!99,70
;Command
    servicegroups      ser-esx-base          ;Service belong
servicegroup
}

```

```

define service{
    use                generic-service ; Inherit values from a
template
    hostgroup_name    esx-servers
    service_description  chk-esx-up
    check_command      check_snmp!-C COMMUNAUTE_SERVEUR -P 2c -o
sysUpTime.0
    servicegroups      ser-esx-base        ; Service belong
servicegroup
}

```

```

#####
###
#####
###
#
# SERVICE GROUPS
#
#####
###
#####
###

```

```
define servicegroup{
    servicegroup_name    ser-esx-base
    alias                 Supervision Linux de base
}
```

Comme d'habitude, vérifier la configuration pour éliminer les éventuelles erreurs.

```
nagios -v /etc/nagios/nagios.cfg
```

Redémarrer le service Nagios.

```
/etc/init.d/nagios restart
```

Accéder à l'interface web de nagios : <http://srv-supervision.domaine.local/nagios/>

## Activer SNMP sur les routeurs ou switchs CISCO

Cf. documentation [installer\\_activer\\_snmp](#).

## Ajout d'un élément réseau en SNMP

Pour superviser les éléments réseaux, on utilise en premier lieu le ping et les différentes vitesses de réponse des paquets et pour obtenir le trafic on utilise systématiquement SNMP. Je n'ai utilisé que la fonction par défaut de Nagios nommé `check_host_alive` qui permet avec des pings réguliers de savoir si la machine répond ou non. Cette fonction par défaut est utilisée pour tous les éléments (serveurs, routeurs...). Ensuite viennent s'ajouter les services que nous allons déclarer.

On crée un host (ici un routeur) qui s'appuie sur le template par défaut `generic-switch`. Remarque : nous utiliserons le template `generic-switch` par défaut pour tous les éléments réseaux (routeurs, switchs, bornes wifi et réseau hertzien).

templates.cfg

```
# Define a template for switches that we can reuse

define host{
    name            generic-switch    ; The name of this host template
    use             generic-host      ; Inherit default values from the
generic-host template
    check_period    24x7              ; By default, switches are monitored
round the clock
    check_interval  3                 ; Switches are checked every 3 minutes
    retry_interval  1                 ; Schedule host check retries at 1 minute
```



```

intervals
    max_check_attempts    10          ; Check each switch 10 times (max)
    check_command         check-host-alive ; Default command to check if
routers are "alive"
    notification_period   24x7        ; Send notifications at any time
    notification_interval 0           ; Resend notifications every 30
minutes
    notification_options  d,r         ; Only send notifications for
specific host states
    contact_groups       admins       ; Notifications get sent to the
admins by default
    register             0            ; DONT REGISTER THIS - ITS JUST A TEMPLATE
}

```

commands.cfg

```

##### Network #####

define command{
    command_name check_net_int
    command_line $USER1$/check_snmp_netint.pl -H $HOSTADDRESS$ -C $ARG1$
$ARG2$ -n $ARG3$ -a -k -y -M -w $ARG4$ -c $ARG5$
}

```

grp-net.cfg

```

# Create a new hostgroup for routers

define hostgroup{
    hostgroup_name  grp-rt           ; The name of the hostgroup
    alias           Routers          ; Long name of the group
}

```

hst-rt.cfg

```

# Define the router that we'll be monitoring

define host{
    use           generic-switch     ; Inherit default values from a
template
    host_name     router1            ; The name we're giving to this switch
    address       @IP                ; IP address of the switch
    hostgroups    grp-rt             ; Host groups this switch is associated
with
}

```

ser-net.cfg

```

##### Routers
#####

```

```
#####  
###  
# Create a service to PING to router  
  
define service{  
    use                generic-service ; Inherit values from a  
template  
    hostgroup_name     grp-rt        ; The name of the host the  
service is associated with  
    service_description Ping          ; The service description  
    check_command       check_ping!80.0,20%!150.0,60% ; The command  
used to monitor the service  
    normal_check_interval 5          ; Check the service every 5  
minutes under normal conditions  
    retry_check_interval 1          ; Re-check the service every  
minute until its final/hard state is determined  
    servicegroups       ser-rt-base  ; Service belong  
servicegroup  
    }  
  
# Monitor uptime via SNMP  
  
define service{  
    use                generic-service ; Inherit values from a  
template  
    hostgroup_name     grp-rt  
    service_description Uptime  
    check_command       check_snmp!-C COMMUNAUTE_RESEAU -P 2c -o  
sysUpTime.0  
    servicegroups       ser-rt-base  ; Service belong  
servicegroup  
    }  
  
# Monitor traffic interface via SNMP  
  
define service{  
    use                generic-service ; Inherit values from a  
template  
    hostgroup_name     grp-rt  
    service_description Trafic  
    #warning to 6 Mo/s (input) and 6 Mo/s (output) and critical to 8  
Mo/s (input) and 8 Mo/s (output)  
    check_command       check_net_int!COMMUNAUTE_RESEAU!--  
v2c!FastEthernet0/0|FastEthernet[12]|Ethernet1/[01]|Atm2/0!6,6!8,8  
    servicegroups       ser-rt-base  ; Service belong  
servicegroup  
    }  
}
```

# Cas particuliers

## Progos et NLB

Serveurs concernés

- srv-progos1
- srv-progos2

Les serveurs sont en NLB (Network Load Balancing). Ceci implique que l'interface réseau standard des serveurs n'est plus visible en SNMP et est remplacé par un identifiant NLB de l'interface. J'ai récupéré l'identifiant 0x564d77617265... qui est malheureusement le nom de l'interface réseau. On ne dispose pas d'un descriptif plus parlant. J'ai ajouté cet identifiant à l'expression régulière afin de pouvoir superviser son trafic. Ce n'est que pour la supervision réseau du serveur qu'il y a une particularité. Pour le reste des services, il n'y a rien de particulier.

From:  
<https://wiki.ouieuhoutca.eu/> - **kilsufi de noter**

Permanent link:  
[https://wiki.ouieuhoutca.eu/supervision\\_nagios\\_snmp](https://wiki.ouieuhoutca.eu/supervision_nagios_snmp)

Last update: **2021/01/21 21:42**

