

Nagios : sensorProbe2

© 2008 [Yoann LAMY](#)

1) Présentation

Le boîtier sensorProbe2 d'AKCP est un équipement réseau de surveillance environnementale. Il est notamment possible de mesurer la température et l'humidité d'une salle serveur en lui associant des capteurs. Un capteur de température/humidité est branché sur le premier port du boîtier.



Ce boîtier embarque un serveur Web et un agent SNMP (version 1).

Par défaut, le boîtier est configuré en DHCP. Si aucun serveur DHCP n'est trouvé, l'adresse IP *192.168.0.100* lui est attribuée. Dans notre exemple, le boîtier AKCP possède l'adresse IP : *192.168.0.4*.

AKCP		sensorProbe2		Admin Log Off									
Location: Salle serveurs		Current System Time: 22/7/08 11:03:22											
Summary		Sensors		Traps		Mail		Network		System		Help	
Network Settings													
IP Address		192.168.0.4											
Subnet Mask		255.255.255.0											
Default Gateway		192.168.0.254											
Domain Name Server		[Redacted]											
Use DHCP		Do Not Use DHCP											
Ethernet MAC ID		[Redacted]											
Ethernet Duplex Mode		10 Mbps Half Duplex											
		Save		Reset									
System Date & Time Settings													
RTC Battery Status		Good											
Time Zone		(GMT+01:00, DST observed) Amsterdam, Berlin, Paris, Rome, Stockholm, Vienna											
Date and Time Settings		[Redacted] (dd/mm/yy hh:mm:ss ex. 22/07/08 11:02:33)											
Use NTP		Once an hour											
NTP Server 1		ntp.mondomaine.com											
NTP Server 2		[Redacted]											
		Save		Reset									

Le nom de la communauté par défaut est *public*. Si vous décidez de restreindre l'accès à l'interface Web d'administration du sensorProbe2 en spécifiant un mot de passe pour l'utilisateur *User*, alors le nom de la communauté correspondra à ce mot de passe (en lecture).

Afin de récupérer les informations (température et humidité) dans Nagios, nous allons détailler deux solutions :

- Plugin *check_snmp*
- Trappes SNMP

2) Plugin *check_snmp*

La première solution pour récupérer les informations mesurées dans Nagios à partir d'un boîtier sensorProbe2 est l'utilisation du plugin actif *check_snmp*. Le principe est plutôt simple. Nagios planifie l'exécution régulière de ce plugin pour récupérer les informations mesurées par le boîtier sensorProbe2 à travers son agent SNMP.

Le plugin *check_snmp* est fourni avec les [plugins officiels de Nagios](#). Nous allons l'utiliser pour superviser la température et l'humidité.

- Ajoutez deux nouvelles commandes : *vi /etc/nagios/objets/commands.cfg*

```
#Temperature (Port 1)
define command {
    command_name check_sensorprobe2_temperature
    command_line $USER1$/check_snmp -H $HOSTADDRESS$ -o .
1.3.6.1.4.1.3854.1.2.2.1.16.1.3.0 -w $ARG2$: $ARG1$ -c $ARG4$: $ARG3$ -C $ARG5$ -l
Temperature -u °C
}

#Humidite (Port 1)
define command {
    command_name check_sensorprobe2_humidite
    command_line $USER1$/check_snmp -H $HOSTADDRESS$ -o .
1.3.6.1.4.1.3854.1.2.2.1.17.1.3.0 -w $ARG2$: $ARG1$ -c $ARG4$: $ARG3$ -C $ARG5$ -l Humidite
-u %
}
```

A noter que les OIDs peuvent être différents selon le port du sensorProbe2 utilisé.

- Ajoutez un nouvel hôte : *vi /etc/nagios/objetscs/hosts/sensor1.cfg*

```
define host {
    host_name sensor1
    use generic-host
    alias Salle serveurs
    address 192.168.0.4
}
```

- Ajoutez les deux nouveaux services supervisés : *vi /etc/nagios/objetscs/services/sensor1.cfg*

```

define service {
    use generic-service
    host_name sensor1
    service_description Temperature
    check_command check_sensorprobe2_temperature!25!30!31!99!public
}

define service {
    use generic-service
    host_name sensor1
    service_description Humidite
    check_command check_sensorprobe2_humidite!60!65!66!100!public
}

```

Le service *Temperature* indiquera un état d'avertissement lorsque la température mesurée sera comprise entre 25°C et 30°C et un état critique pour une température comprise entre 31°C et 99°C. Le service *Humidite* indiquera un état d'avertissement lorsque l'humidité relevée sera comprise entre 60% et 65% et un état critique pour une humidité comprise entre 66% et 100%.

Le modèle *generic-service* possède l'attribut *active_checks_enabled* à 1.

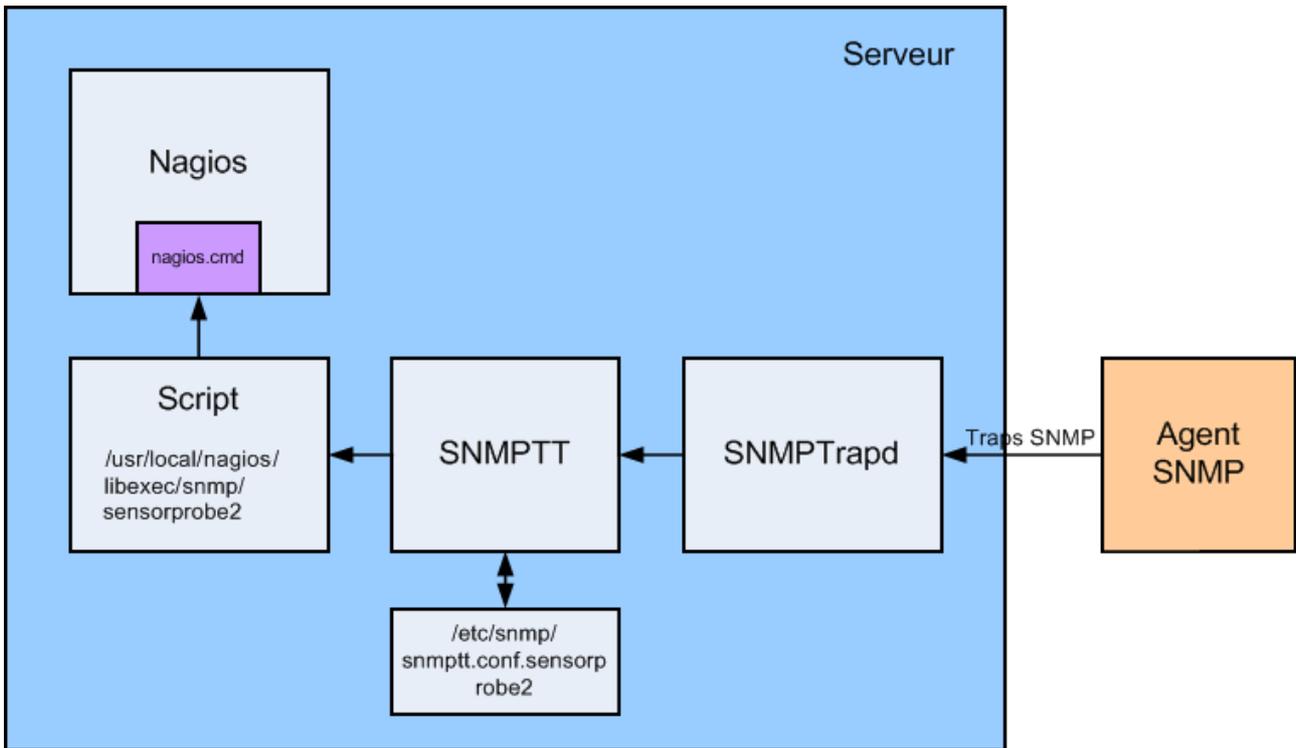
- Redémarrez le service Nagios : *service nagios restart*

3) Trappes SNMP

Une autre solution, plus complexe à mettre en œuvre, consiste à récupérer les trappes SNMP. En cas de problème, selon les seuils fixés, le boîtier AKCP envoie des trappes SNMP par l'intermédiaire de l'agent. Il est alors nécessaire de configurer le serveur de supervision pour récupérer les trappes SNMP sur le port UDP 162.

L'avantage de cette solution est que vous êtes immédiatement prévenu lors d'un dépassement des seuils.

Le schéma ci-dessous décrit le principe de fonctionnement pour récupérer les trappes SNMP et les exploiter dans Nagios.



Pour commencer, il faut configurer le boîtier sensorProbe2 pour activer l'envoi de trappes SNMP vers le serveur de supervision.

AKCP		sensorProbe2		Admin	Log Off
Location: Salle serveurs			Current System Time: 22/7/08 10:55:22		
Summary	Sensors	Traps	Mail	Network	System
Trap Settings					
Trap 1					
Send Trap	On ▾				
Destination IP	192.168.0.104				
Community	public				
Send Keep Alive Trap	Off ▾				
Keep Alive Trap Resend Interval (mins)	20	20 mins			
<input type="button" value="Save"/> <input type="button" value="Reset"/>					

192.168.0.104 correspond à l'adresse IP du serveur de supervision.

Vous devez également vérifier que le boîtier sensorProbe2 va bien envoyer une trappe SNMP lorsque l'état d'un capteur revient à un seuil normal.

Admin Log Off

AKCP sensorProbe2

Location: Salle serveurs Current System Time: 2/8/08 11:54:46

[Summary](#) | [Sensors](#) | [Traps](#) | [Mail](#) | [Network](#) | [System](#) | [Help](#)

Sensor Settings

Environmental	Trap/Email Filters	
Temperature	Port	1
Humidity	Sensor Type	Temperature
Water Detector	Sensor Description	Temperature1 Description
Airflow Sensor		
Contacts & Drivers	Send Trap/Email when Status "Normal"	Yes ▾
Dry Contacts & Drivers	Continuous time sensor is normal to report (secs)	<input type="text" value="0"/> 0 secs
4-20 mAmp	Continuous time sensor is warning/error to report (secs)	<input type="text" value="0"/> 0 secs
Dry Contacts (3 - 12)		
Power	Minimum time between each Trap/Email	<input type="text" value="0"/> minutes

Terminé

Ensuite, côté serveur de supervision, la première étape est de récupérer les trappes SNMP provenant du boîtier sensorProbe2. SNMPTrapd est le programme que nous allons utiliser pour réaliser cette tâche. Nous supposons que [Net-SNMP](#) a été installé au préalable.

- Éditez le fichier de configuration de SNMPTrapd : `vi /etc/snmp/snmptrapd.conf`

```
traphandle default /usr/sbin/snmpptt --ini=/etc/snmp/snmpptt.ini
snmpTrapdAddr udp:192.168.0.104:162
authCommunity log,execute public
```

Le paramètre `traphandle` permet de lancer un programme (en l'occurrence SNMPTT) selon un OID. Le paramètre `snmpTrapdAddr` permet de spécifier les interfaces réseaux en écoute. `authCommunity` autorise les trappes de la communauté spécifiée à transmettre les trappes au programme spécifié par le paramètre `traphandle`. La journalisation est activée temporairement pour vérifier que le serveur de supervision reçoit les trappes SNMP provenant du boîtier sensorProbe2.

Si vous disposez d'un pare-feu sur votre serveur de supervision, n'oubliez pas d'autoriser le port UDP 162 en entrée.

- Éditez le fichier des règles du pare-feu (IPTables) : `vi /etc/sysconfig/iptables`

```
-A RH-Firewall-1-INPUT -m state --state NEW -m udp -p udp --dport 162 -j ACCEPT
```

- Redémarrez le pare-feu : `service iptables restart`

- Éditez le script de lancement de SNMPTrapd : `vi /etc/rc.d/init.d/snmptrapd`

```
OPTIONS="-On -Lsd -p /var/run/snmptrapd.pid"
```

L'argument `-On` spécifie que les OID sont laissés sous forme numérique. L'argument `-p` indique l'emplacement du fichier contenant le numéro du processus. Les arguments `-Lsd` indiquent l'utilisation

du fichier de journalisation du système (*/var/log/messages*) lorsque cette fonctionnalité est activée.

- Lancez le service SNMPTrapd au démarrage de la machine : *chkconfig snmptrapd on*
- Lancez le service SNMPTrapd : *service snmptrapd start*

En parcourant le fichier de journalisation du système du serveur de supervision, vous pouvez alors vérifier que les trappes SNMP sont bien reçues.

```
Jul 30 17:19:40 supervision snmptrapd[1665]: 2008-07-30 17:19:40 192.168.0.4 (via
UDP: [192.168.0.4]:162) TRAP, SNMP v1, community public#012#011.1.3.6.1.4.1.385
4.1 Enterprise Specific Trap (5) Uptime: 1:54:28.50#012#011.1.3.6.1.4.1.3854.1.2
.2.1.3.0 = STRING: "00-0B-DC-00-86-A1"#011.1.3.6.1.4.1.3854.1.7.8.0 = INTEGER: 1
#011.1.3.6.1.4.1.3854.1.7.4.0 = INTEGER: 0#011.1.3.6.1.4.1.3854.1.7.2.0 = INTEGE
R: 28#011.1.3.6.1.4.1.3854.1.7.8.0 = INTEGER: 3#011.1.3.6.1.4.1.3854.1.7.4.0 = I
NTEGER: 0#011.1.3.6.1.4.1.3854.1.7.2.0 = INTEGER: 60
```

Les OIDs des trappes SNMP sont pour l'instant sous forme numérique. Le script [SNMPTT](#) va permettre dans un premier temps de les traduire et dans un deuxième temps d'exécuter un autre script pour récupérer les informations mesurées dans Nagios.

Nous allons installer SNMPTT.

- Placez-vous dans le répertoire suivant : *cd /usr/local/src/*
- Installez ce pré-requis : *yum install perl-Config-IniFiles*
- Téléchargez la dernière version de SNMPTT avec la commande *wget*.
- Décompressez l'archive : *tar -xvzf snmptt_version.tar.gz*
- Placez-vous dans le répertoire suivant : *cd snmptt_version*
- Copiez le fichier suivant : *cp snmptt /usr/sbin/*
- Rendez le script exécutable : *chmod +x /usr/sbin/snmptt*
- Copiez le fichier suivant : *cp snmptthandler /usr/sbin/*
- Rendez le script exécutable : *chmod +x /usr/sbin/snmptthandler*
- Copiez le fichier suivant : *cp snmptt.ini /etc/snmp/*
- Copiez le fichier suivant : *cp snmpttconvert /usr/bin/*
- Rendez le script exécutable : *chmod +x /usr/bin/snmpttconvert*
- Copiez le fichier suivant : *cp snmpttconvertmib /usr/bin/*
- Rendez le script exécutable : *chmod +x /usr/bin/snmpttconvertmib*

SNMPTT sera utilisé en mode *stand alone*. Cela signifie que SNMPTrapd fera appel à lui à chaque réception de trappes SNMP.

- Éditez le fichier de configuration de SNMPTT : *vi /etc/snmp/snmptt.ini*

```

[General]
snmptt_system_name =
mode = standalone
multiple_event = 1
dns_enable = 0
strip_domain = 1
strip_domain_list = <<END
mondomaine.com
END

resolve_value_ip_addresses = 0
net_snmp_perl_enable = 1
net_snmp_perl_best_guess = 0
translate_log_trap_oid = 0
translate_value_oids = 1
translate_enterprise_oid_format = 1
translate_trap_oid_format = 1
translate_varname_oid_format = 1
translate_integers = 1
wildcard_expansion_separator = " "
allow_unsafe_regex = 0
remove_backslash_from_quotes = 0
dynamic_nodes = 0
description_mode = 0
description_clean = 1
threads_enable = 0
threads_max = 10

[Logging]
stdout_enable = 0
log_enable = 0
log_file = /var/log/snmptt.log
log_system_enable = 0
log_system_file = /var/log/snmpttssystem.log
unknown_trap_log_enable = 0
unknown_trap_log_file = /var/log/snmpttunknown.log
statistics_interval = 0
syslog_enable = 0
syslog_facility = local0
syslog_level_debug = <<END
END
syslog_level_info = <<END
END
syslog_level_notice = <<END
END
syslog_level_warning = <<END
END
syslog_level_err = <<END
END
syslog_level_crit = <<END

```

```

syslog_level_alert = <<END
END

syslog_level = info
syslog_system_enable = 0
syslog_system_facility = local0
syslog_system_level = warning

[Exec]
exec_enable = 1
pre_exec_enable = 0
unknown_trap_exec =
unknown_trap_exec_format =
exec_escape = 1

[Debugging]
DEBUGGING = 0
DEBUGGING_FILE =
DEBUGGING_FILE_HANDLER =

[TrapFiles]
snmptt_conf_files = <<END
END

```

L'agent SNMP envoie les trappes SNMP en spécifiant l'adresse IP du boîtier. Nagios se basant sur l'attribut *host_name*, il nous faut donc convertir l'adresse IP en nom d'hôte.

Pour cela, il y a deux possibilités :

- Si vous disposez d'un serveur DNS, vous pouvez mettre à *1* le paramètre *dns_enable* du fichier */etc/snmp/snmptt.ini* pour activer la résolution inversée. Le serveur DNS retournera alors le nom complet de l'hôte (par exemple, *sensor1.mondomaine.com*). Pour ne garder que le nom (*sensor*), on peut activer le paramètre *strip_domain*. Le paramètre *strip_domain_list* indique les noms de domaines à ne pas prendre en compte (par exemple, *mondomaine.com*).
- L'autre possibilité est de réaliser cette correspondance à l'aide d'un script. Nous allons retenir cette solution.

Pour que SNMPTT puisse traduire les OIDs, un fichier de correspondance doit lui être spécifié. SNMPTT ne reconnaît pas les fichiers MIB (*Management Information Base*). Il nous faut donc convertir le fichier MIB fourni par AKCP pour le boîtier sensorProbe2 vers un format de fichier que SNMPTT pourra comprendre. Cette opération s'effectue à l'aide du script *snmpttconvertmib*.

- Placez-vous dans le répertoire suivant : *cd /usr/share/snmp/mibs/*

Le fichier MIB du boîtier sensorProbe2 se trouve sur le CD-ROM de support.

- Effectuez la conversion : `snmpttconvertmib --in=/usr/share/snmp/mibs/sp.mib --out=/etc/snmp/snmptt.conf.sensorprobe2`

```
Done
Total translations:      99
Successful translations: 99
Failed translations:     0
```

Il ne manque plus qu'à spécifier l'emplacement du fichier converti dans le fichier de configuration de SNMPTT.

- Éditez à nouveau le fichier suivant : `vi /etc/snmp/snmptt.ini`

```
...
[TrapFiles]
snmptt_conf_files = <<END
/etc/snmp/snmptt.conf.sensorprobe2
END
```

Une fois les trappes SNMP récupérées, il faut envoyer celles concernant la température et l'humidité, à Nagios. Nagios possède un fichier de commandes externes (*nagios.cmd*) qui est lu régulièrement. Les informations mesurées seront placées dans ce fichier pour être intégrées dans Nagios. Le script `/usr/local/nagios/libexec/snmp/sensorprobe2` est en charge de ces opérations.

- Éditez le fichier suivant : `vi /etc/snmp/snmptt.conf.sensorprobe2`

```
EVENT spTemperature1Status .1.3.6.1.4.1.3854.1.0.101 "Status Events" MAJOR
FORMAT Temperature sensor trap $*
EXEC /usr/local/nagios/libexec/snmp/sensorprobe2 $ar Temperature $1 $2
SDESC
Major: Temperature sensor trap
Variables:
1: spSensorStatus
2: spSensorValue
3: spSensorLevelExceeded
4: spSensorIndex
5: spSensorName
6: spSensorDescription
EDESC

EVENT spAnalogue1Status .1.3.6.1.4.1.3854.1.0.201 "Status Events" MAJOR
FORMAT Analogue Sensor Type $*
EXEC /usr/local/nagios/libexec/snmp/sensorprobe2 $ar Humidite $1 $2
SDESC
Analogue Sensor Type
Variables:
1: spSensorStatus
2: spSensorValue
3: spSensorLevelExceeded
4: spSensorIndex
```

5: *spSensorName*
6: *spSensorDescription*
EDESC

La variable *\$ar* correspondant à l'adresse IP du boîtier sensorProbe2. *\$1* indique l'état du capteur (*spSensorStatus*) et *\$2* indique la valeur mesurée du capteur (*spSensorValue*).

Le fichier de commandes externes de Nagios exige un format particulier. Le script */usr/local/nagios/libexec/snmp/sensorprobe2* doit donc s'y conformer. Il est nécessaire également de traduire l'état du capteur avec les codes de retour qu'attend Nagios.

Dans le fichier MIB du sensorProbe2, on peut trouver les différents états correspondant à la variable *spSensorStatus*.

Numéro	État du capteur	Code de retour dans Nagios
1	Inconnu (<i>nostatus</i>)	<i>UNKNOWN</i> (3)
2	Normal (<i>normal</i>)	<i>OK</i> (0)
3	Avertissement haut (<i>highWarning</i>)	<i>WARNING</i> (1)
4	Critique haut (<i>highCritical</i>)	<i>CRITICAL</i> (2)
5	Avertissement bas (<i>lowWarning</i>)	<i>WARNING</i> (1)
6	Critique bas (<i>lowCritical</i>)	<i>CRITICAL</i> (2)
7	Erreur du capteur (<i>sensorError</i>)	<i>CRITICAL</i> (2)
8	Capteur activé (<i>turnOn</i>)	<i>OK</i> (0)
9	Capteur éteint (<i>turnOff</i>)	<i>WARNING</i> (1)

- Créez le répertoire suivant : *mkdir /usr/local/nagios/libexec/snmp/*
- Créez le script suivant : *vi /usr/local/nagios/libexec/snmp/sensorprobe2*

```
#!/bin/bash

#Fichier de commandes externes de Nagios
NAGIOS_CMD_FILE='/var/nagios/rw/nagios.cmd'

#Fichier de correspondance adresse IP/Nom
SNMP_HOSTNAME='/etc/nagios/hostname.snmp'

#Correspondance entre l'adresse IP et le nom d'hote Nagios
HOST=`/bin/cat $SNMP_HOSTNAME | /bin/grep $1 | /bin/awk '{ print $2}'`

#Nom du service (propriete 'service_description' dans Nagios)
SERVICE_DESC=$2

#Traduction du code de retour (sensorProbe2 vers Nagios)
case $3 in
  1) CODE=3 ;;
  2|8) CODE=0 ;;
  3|5) CODE=1 ;;
  4|6|7) CODE=2 ;;
esac

#Description de la trap SNMP ('Status Information' dans l'interface Web de Nagios)
case $2 in
  Temperature) INFORMATION="La temperature est de ${4}°C" ;;
  Humidite) INFORMATION="L'humidite est de ${4}%" ;;
esac

#Recupere la date en secondes
DATE=`date +%s`

#Ajoute la ligne dans le fichier $NagiosCmdFile
/bin/echo "[${DATE}] PROCESS_SERVICE_CHECK_RESULT;${HOST};${SERVICE_DESC};${CODE};
${INFORMATION}" >> $NAGIOS_CMD_FILE
```

- Rendez le script exécutable : `chmod +x /usr/local/nagios/libexec/snmp/sensorprobe2`

Nous devons créer le fichier de correspondance entre les adresses IP et les noms d'hôtes.

- Créez le fichier de correspondance : `vi /etc/nagios/hostname.snmp`

192.168.0.4 sensor1

Dans la configuration de Nagios, vous devez vérifier que les commandes externes et que les plugins passifs sont autorisés.

- Éditez le fichier de configuration de Nagios : `vi /etc/nagios/nagios.cfg`

```
check_external_commands=1
...
accept_passive_service_checks=1
```

Pour terminer, il ne vous reste plus qu'à définir l'hôte et les services supervisés (température et humidité).

- Ajoutez l'hôte si ce n'est pas déjà fait : *vi /etc/nagios/objetcs/hosts/sensor1.cfg*

```
define host {
    host_name sensor1
    use generic-host
    alias Salle serveurs
    address 192.168.0.4
}
```

- Éditez le fichier suivant : *vi /etc/nagios/objetcs/services/templates.cfg*

```
define service {
    name generic-service-passif
    active_checks_enabled 0
    passive_checks_enabled 1
    parallelize_check 1
    obsess_over_service 0
    check_freshness 0
    notifications_enabled 1
    event_handler_enabled 0
    flap_detection_enabled 1
    process_perf_data 0
    retain_status_information 1
    retain_nonstatus_information 1
    is_volatile 0
    check_period 24x7
    retry_interval 1
    max_check_attempts 3
    check_interval 10
    notification_options w,u,c,r
    notification_interval 360
    notification_period 24x7
    contact_groups administrateurs
    check_command check_dummy!0
    register 0
}
```

Le modèle *generic-service-snmptrap* active l'utilisation des plugins passifs à l'aide de l'attribut *passive_checks_enabled*.

- Éditez le fichier suivant : *vi /etc/nagios/objetcs/services/sensor1.cfg*

```
define service {
    use generic-service-passif
    host_name sensor1
    service_description Temperature
}

define service {
    use generic-service-passif
    host_name sensor1
    service_description Humidite
}
```

- Redémarrez le service Nagios : *service nagios restart*