Table des matières

[1. Installation et configuration de CentCTL 1](#_Toc92570683)

[2. Test de la configuration 3](#_Toc92570684)

[3. Automatiser le déploiement 3](#_Toc92570685)

# 1. Installation et configuration de CentCTL

Pour installer et configurer CentCTL nous nous aidons du site d’Aukfood :

<https://www.aukfood.fr/supervision-tour-dhorizon-avec-centctl-pour-centreon-open-source/>

**/ ! \ L’installation se fait avec un utilisateur normal (pas en root)**

**Installation :**

Nous devons commencer par créer un répertoire nommé centctl et nous installons l’outil dedans :

mkdir $HOME/centctl

cd $HOME/centctl

wget https://github.com/YPSI-SAS/centctl/releases/download/v2.5/centctl\_linux\_amd64 -O centctl

Il nous faut rendre le fichier exécutable :

chmod +x centctl

Il faut maintenant que nous modifions le fichier .bashrc (situé dans /home/$USER/.bashrc) :

**Attention à bien remplacer le $USER par votre nom d’utilisateur**

export CENTCTL\_CONF=/home/$USER/centctl/centctl.yaml

source /usr/share/bash-completion/bash\_completion

Nous devons aussi installer le fichier bash-completion si ce n’est pas fait :

sudo apt install bash-completion

On vérifie aussi que la complétion se fasse :

source ~/.bashrc

type \_init\_completion

La commande doit nous ressortir quelque chose commençant comme ça :

debian@debian10:~/centctl$ type \_init\_completion

\_init\_completion is a function

\_init\_completion ()

{

local exclude= flag outx errx inx OPTIND=1;

while getopts "n:e:o:i:s" flag "$@"; do

case $flag in

n)

exclude+=$OPTARG

… … … … … … … …

… … … … … … … …

… … … … … … … …

… … … … … … … …

Nous ajoutons maintenant le script de complétion dans le répertoire /etc/bash\_completion.d **(attention à remplacer le $USER par l’utilisateur de la machine)** :

sudo su

/home/$USER/centctl/centctl completion bash > /etc/bash\_completion.d/centctl

Si le répertoire n’existe pas il faut le créer :

mkdir /etc/bash\_completion.d

Nous repassons en utilisateur normal :

exit

**Configuration de CentCTL:**

Nous devons maintenant construire un fichier de configuration .yaml pour déclarer notre serveur **(attention il faut créer le fichier dans /home/$USER/centctl/)**  :

debian@debian10:~/centctl$ nano centctl.yaml

Dans ce fichier nous mettons ce qui suit :

servers:

- server: "NOM DU SERVEUR DONNE LORS DE LA CREATION"

url: "http://[IP\_SERVER]/centreon"

login: "LOGIN LORS DE LA CREATION DU COMPTE"

password: "MOT DE PASSE DONNE LORS DE LA CREATION"

version: "v1"

# 2. Test de la configuration

Maintenant que CentCTL est bien installé et configuré nous pouvons tester quelques commandes pour vérifier son bon fonctionnement :

Lister les hôtes sur le serveur :

debian@debian10:~/centctl$ ./centctl --server=centreon list host --output csv

ID,Name,Alias,Address,Activate

2,debian,Debian10,192.168.7.1,1

29,Windows10,,192.168.6.1,1

Nous retrouvons bien les hôtes et les IP qui sont sur notre serveur

Il existe plusieurs commandes que nous pouvons passer par exemple pour exporter ou importer une configuration. Nous pouvons les retrouver sur le site donné en début de rapport

# 3. Automatiser le déploiement

Dans cette partie nous devons automatiser le déploiement d’un nouveau service sur un ensemble d’hôtes. Pour se faire nous devons sur CentCTL :

* Créer une liste d’hôtes (en récupérant nos hôtes crées dans les précédents TP)
* Créer une nouvelle commande/check
* Créer des templates de services/hôtes grâce à CentCTL
* Créer un script de déploiement

**Création d’une liste d’hôte :**

Pour la création d’une liste d’hôtes nous allons simplement récupérer les hôtes crées dans les TP précédents grâce à la commande suivante :

debian@debian10:~/centctl$ ./centctl --server=centreon export host -r ^.\* -f hosts.csv

Nous pouvons vérifier que le fichier est bien présent :

debian@debian10:~/centctl$ ls

centctl centctl.yaml hosts.csv

**Création d’une nouvelle commande/check :**

On peut créer une commande de check de la façon suivante :

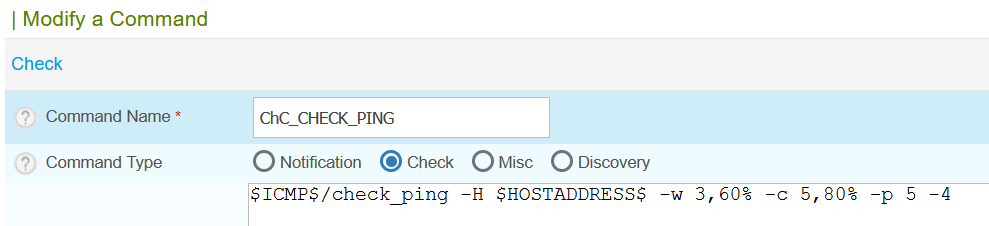
debian@debian10:~/centctl$ ./centctl --server=centreon add command -l "\$ICMP$/check\_ping -H \$HOSTADDRESS$ -w 3,60% -c 5,80% -p 5 -4" -n ChC\_CHECK\_PING -t check

Syntaxe de la commande shell :

1. -n, --name string To define the name of the command
2. -t, --type string To define the type of the command (check, notif, misc or discovery)
3. -l, --line string To define the line of the command

Il faut mettre un antislash devant toutes les Macro sinon le shell va l’interpréter comme une variable vide et dans centreon nous aurons une erreur

Nous pouvons vérifier dans notre serveur centreon que la commande a bien été créée :



**Création des templates de services :**

Maintenant que la commande est créée nous pouvons créer un template de services que l’on va lier à notre commande :

debian@debian10:~/centctl$ ./centctl --server=centreon add template service -a SFT\_ChC\_CHECK\_PING -n ST\_ChC\_CHECK\_PING

INFO: The STPL ST\_ChC\_CHECK\_PING is created

Il nous faut aussi modifier les paramètres généraux :

debian@debian10:~/centctl$ ./centctl --server=centreon modify template service -n ST\_ChC\_CHECK\_PING -p check\_command -v ChC\_CHECK\_PING

INFO: The parameter check\_command of the STPL ST\_ChC\_CHECK\_PING is modified

debian@debian10:~/centctl$ ./centctl --server=centreon modify template service -n ST\_ChC\_CHECK\_PING -p check\_period -v 24x7

INFO: The parameter check\_period of the STPL ST\_ChC\_CHECK\_PING is modified

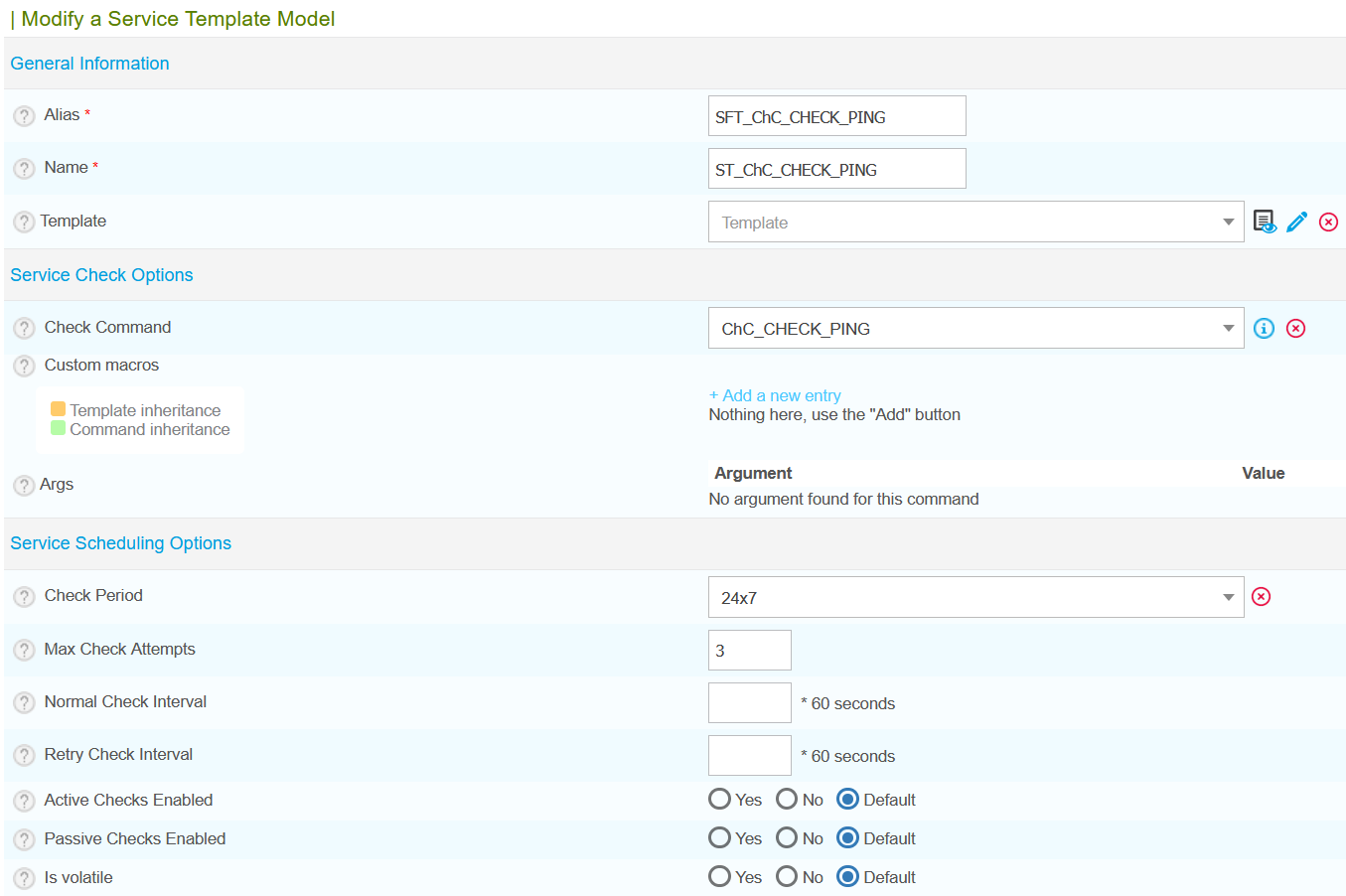
debian@debian10:~/centctl$ ./centctl --server=centreon modify template service -n ST\_ChC\_CHECK\_PING -p max\_check\_attempts -v 3

INFO: The parameter max\_check\_attempts of the STPL ST\_ChC\_CHECK\_PING is modified

-p : paramètre à modifier

-v : valeur que l’on veut mettre au paramètre

Nous pouvons vérifier que le template est bien crée dans centreon :



**Création des templates d’hôtes :**

Maintenant que le template de services est créé nous pouvons créer un template d’hôtes qui va regrouper les différents services crées et qui sera lié à notre hôte (pour le moment nous n’avons qu’un seul service à des fins de test) :

debian@debian10:~/centctl$ ./centctl --server=centreon add template host -a HFT\_DEBIAN\_WEB\_SERVER -n HT\_DEBIAN\_WEB\_SERVER

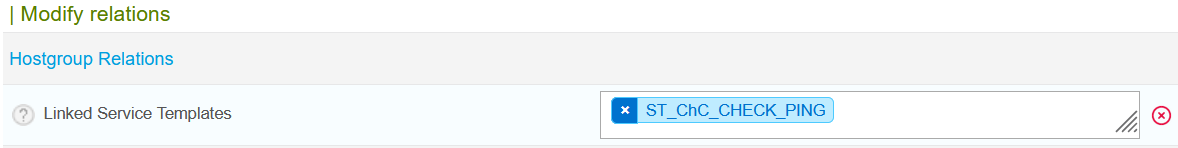
INFO: The HTPL HT\_DEBIAN\_WEB\_SERVER is created

Maintenant que le template est créé il faut le lier à tous les templates de services que nous avons crée et avons besoin pour un hôte. Ici nous n’avons crée que le ping mais si nous avions plusieurs services pour un hôte Linux par exemple il faudrait lier tous les templates de services Linux au template d’hôte et donc répéter la commande suivante pour chaque service en modifiant la syntaxe du -v :

debian@debian10:~/centctl$ ./centctl --server=centreon modify template host -n HT\_DEBIAN\_WEB\_SERVER -p linkedservice -v ST\_ChC\_CHECK\_PING

INFO: The parameter linkedservice of the STPL HT\_DEBIAN\_WEB\_SERVER is modified

Nous pouvons vérifier dans centreon que le template de service ping est bien lié à notre service d’hôte :



**Création d’un hôte :**

Il ne nous reste plus qu’à créer un hôte avec un nom et une IP puis le lier à notre template d’hôte qui regroupe les différents services (nous pourrions au lieu de créer un hôte, lier le template à un hôte déjà existant) :

debian@debian10:~/centctl$ ./centctl --server=centreon add host -i 192.168.7.1 -a debian\_test\_ping -n debian\_test\_ping -p Central -t HT\_DEBIAN\_WEB\_SERVER

INFO: The host debian\_test\_ping is created

INFO: The template of the host debian\_test\_ping is applied

-i : IP de l’hôte

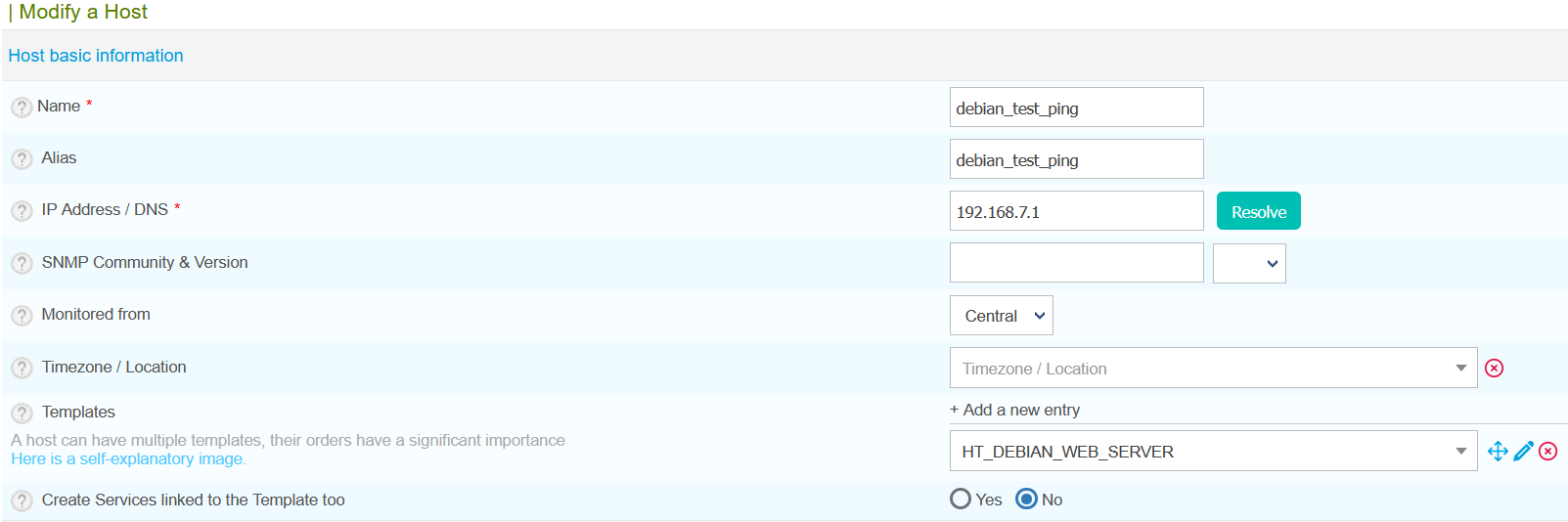
-a : Alias au nom de l’hôte

-n : Nom de l’hôte

-p : Serveur qui supervise cet hôte

-t : Template à appliquer sur l’hôte

Nous pouvons vérifier dans centreon que l’hôte est bien créé et lié au template d’hôte :



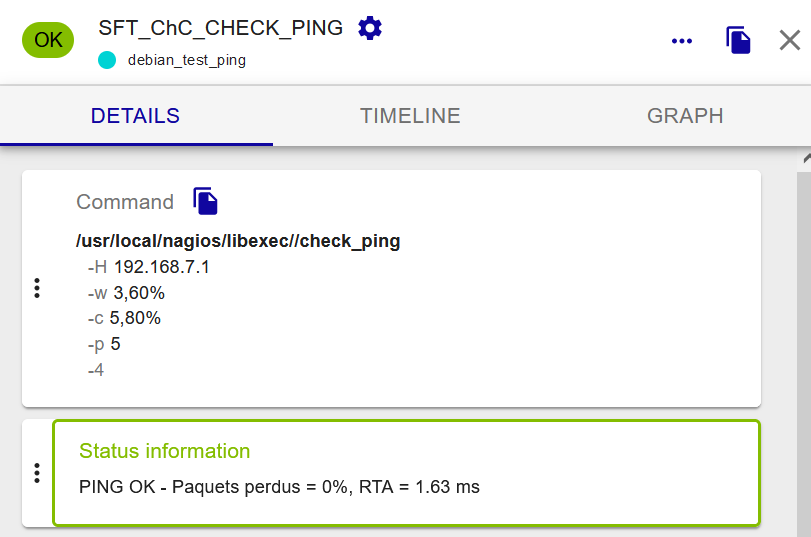
**Appliquer les modifications sur le serveur :**

Pour appliquer les modifications sur le serveur il faut exporter le fichier de configuration de la façon suivante :

debian@debian10:~/centctl$ ./centctl --server=centreon apply -n Central

INFO: The configuration of the poller Central is exported

Nous pouvons maintenant vérifier dans l’Onglet Monitoring que notre hôte est bien supervisé au niveau du service ping :



**Création d’un script de déploiement :**

Pour créer un script de déploiement nous devons commencer par récupérer les hôtes de notre serveur et ne garder que leur nom :

debian@debian10:~/centctl$ ./centctl --server=centreon list host --output csv |cut -d, -f 2 > hosts.csv

Le cut permet de ne garder que le 2ème champ de notre fichier ce qui correspond au nom des hôtes

Le "> hosts.csv" permet d’écrire le résultat de la commande dans le fichier hosts.csv et d’écraser les anciennes données s’il y en avait

Il faut éditer le fichier hosts.csv et supprimer la 1ère ligne avec "Name"

Maintenant nous avons notre fichier hosts de la façon suivante :

Windows10

debian

Nous pouvons donc maintenant créer un script qui va récupérer le nom des hôtes dans le fichier hosts.csv et qui va créer différentes commandes de supervision en fonction de l’hôte sélectionné

Ici dans notre script nous avons créé 3 commandes : 1 en NRPE, 1 en SSH et 1 en SNMP

Le script va créer pour chaque hôte une commande, un template de services, un template d’hôte puis va lier le template d’hôte à l’hôte

Pour terminer le script va exporter la configuration dans le serveur principal afin de mettre à jour les données

Le script est donc le suivant :

#!/bin/bash

#Déclare la liste d'hôte

listhost="$(<hosts.csv)"

#Boucle sur tout les hôtes

for host in $listhost; do

#Création d'une supervision NRPE sur la machine Windows

if [ "$host" == "Windows10" ]; then

echo "Création de la commande NRPE pour l'hôte $host :"

./centctl --server=centreon add command -l "\$USER1$/check\_centreon\_nrpe3 -H \$HOSTADDRESS$ -c check\_updates" -n check\_updates\_nr$

echo "Commande NRPE créée"

echo "Création d'un template de service pour le NRPE"

./centctl --server=centreon add template service -a updates\_nrpe -n updates\_nrpe

./centctl --server=centreon modify template service -n updates\_nrpe -p check\_command -v check\_updates\_nrpe

./centctl --server=centreon modify template service -n updates\_nrpe -p check\_period -v 24x7

./centctl --server=centreon modify template service -n updates\_nrpe -p max\_check\_attempts -v 3

echo "Template de services NRPE créée"

echo "Création d'un template d'hôtes"

./centctl --server=centreon add template host -a template\_windows\_nrpe -n template\_windows\_nrpe

./centctl --server=centreon modify template host -n template\_windows\_nrpe -p linkedservice -v updates\_nrpe

echo "Template d'hôte créée"

echo "Liaison du template sur l'hôte"

./centctl --server=centreon modify host -n $host -p template -v template\_windows\_nrpe

fi

#Création d'une supervison by SSH sur la machine debian

if [ "$host" == "debian" ]; then

echo "Création de la commande by SSH pour l'hôte $host :"

./centctl --server=centreon add command -l "\$LOCAL$/check\_by\_ssh -H \$HOSTADDRESS$ -i ""\$SSH$"" -l \$HOSTNAME$ -C '\$LOCAL$/che$

echo "Commande SSH créée"

echo "Création d'un template de service pour le SSH"

./centctl --server=centreon add template service -a check\_ssh\_apache -n check\_ssh\_apache

./centctl --server=centreon modify template service -n check\_ssh\_apache -p check\_command -v check\_by\_ssh

./centctl --server=centreon modify template service -n check\_ssh\_apache -p check\_period -v 24x7

./centctl --server=centreon modify template service -n check\_ssh\_apache -p max\_check\_attempts -v 3

echo "Template de services SSH créée"

echo "Création d'un template d'hôtes"

./centctl --server=centreon add template host -a template\_debian -n template\_debian

./centctl --server=centreon modify template host -n template\_debian -p linkedservice -v check\_ssh\_apache

echo "Template d'hôte créée"

#Création d'une supervision SNMP sur la machine debian

echo "Création de la commande SNMP pour l'hôte $host :"

./centctl --server=centreon add command -l "\$CENTREONPLUGINS$/centreon\_linux\_snmp.pl --plugin=os::linux::snmp::plugin --mode=cpu$

echo "Commande SNMP créée"

echo "Création d'un template de service pour le SNMP"

./centctl --server=centreon add template service -a check\_snmp -n check\_snmp

./centctl --server=centreon modify template service -n check\_snmp -p check\_command -v check\_cpu\_snmp

./centctl --server=centreon modify template service -n check\_snmp -p check\_period -v 24x7

./centctl --server=centreon modify template service -n check\_snmp -p max\_check\_attempts -v 3

echo "Template de services SNMP créée"

echo "Association au template actuel pour l'hôte"

./centctl --server=centreon modify template host -n template\_debian -p linkedservice -v check\_snmp

echo "Association terminée"

echo "Liaison du template sur l'hôte"

./centctl --server=centreon modify host -n $host -p template -v template\_debian

fi

done

#Exporte la configuration sur le poller selectionné et le recharge

./centctl --server=centreon apply -n Central

echo "Le déploiement de la supervision est terminé"

echo "Vous pouvez ouvrir le serveur dans un navigateur web à l'adresse suivante : "

echo "http://IP\_Serveur/centreon"

echo "Vérifiez le bon fonctionnement du script en regardant l'onglet Monitoring"

Nous pouvons donc lancer le script :

debian@debian10:~/centctl$ ./autodeploy.sh

Création de la commande NRPE pour l'hôte Windows10 :

INFO: The CMD check\_updates\_nrpe is created

Commande NRPE créée

Création d'un template de service pour le NRPE

INFO: The STPL updates\_nrpe is created

INFO: The parameter check\_command of the STPL updates\_nrpe is modified

INFO: The parameter check\_period of the STPL updates\_nrpe is modified

INFO: The parameter max\_check\_attempts of the STPL updates\_nrpe is modified

Template de services NRPE créée

Création d'un template d'hôtes

INFO: The HTPL template\_windows\_nrpe is created

INFO: The parameter linkedservice of the STPL template\_windows\_nrpe is modified

Template d'hôte créée

Liaison du template sur l'hôte

INFO: The parameter template of the host Windows10 is modified

INFO: The template of the host Windows10 is applied

Création de la commande by SSH pour l'hôte debian :

INFO: The CMD check\_by\_ssh is created

Commande SSH créée

Création d'un template de service pour le SSH

INFO: The STPL check\_ssh\_apache is created

INFO: The parameter check\_command of the STPL check\_ssh\_apache is modified

INFO: The parameter check\_period of the STPL check\_ssh\_apache is modified

INFO: The parameter max\_check\_attempts of the STPL check\_ssh\_apache is modified

Template de services SSH créée

Création d'un template d'hôtes

INFO: The HTPL template\_debian is created

INFO: The parameter linkedservice of the STPL template\_debian is modified

Template d'hôte créée

Création de la commande SNMP pour l'hôte debian :

INFO: The CMD check\_cpu\_snmp is created

Commande SNMP créée

Création d'un template de service pour le SNMP

INFO: The STPL check\_snmp is created

INFO: The parameter check\_command of the STPL check\_snmp is modified

INFO: The parameter check\_period of the STPL check\_snmp is modified

INFO: The parameter max\_check\_attempts of the STPL check\_snmp is modified

Template de services SNMP créée

Association au template actuel pour l'hôte

INFO: The parameter linkedservice of the STPL template\_debian is modified

Association terminée

Liaison du template sur l'hôte

INFO: The parameter template of the host debian is modified

INFO: The template of the host debian is applied

INFO: The configuration of the poller Central is exported

Le déploiement de la supervision est terminé

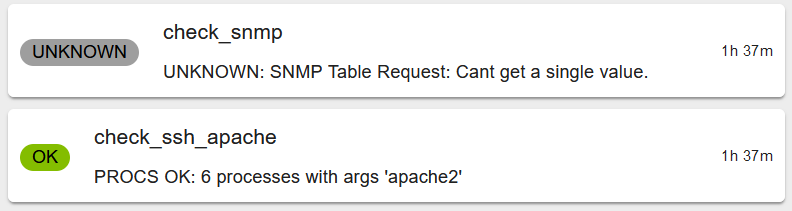
Vous pouvez ouvrir le serveur dans un navigateur web à l'adresse suivante :

http://IP\_Serveur/centreon

Vérifiez le bon fonctionnement du script en regardant l'onglet Monitoring

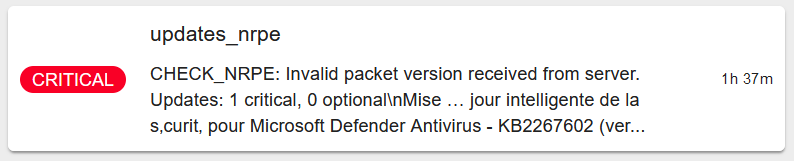
Notre script a bien fonctionné et nous pouvons voir dans l’onglet "Monitoring que nos services sont mis en place :

**Pour l’hôte debian :**



La commande de check SNMP est fonctionnelle sur le serveur mais comme le SNMP n’est pas configuré sur ma machine le serveur nous remonte un statut « UNKNOWN »

**Pour l’hôte Windows10 :**



Nous aurions pu mettre des arguments dans nos commandes mais nous n’avons pas trouvé la commande pour définir un argument dans le template de service

Création d’une commande avec argument (attention aux antislash) :

debian@debian10:~/centctl$ ./centctl --server=centreon add command -l "\$USER1$/check\_centreon\_nrpe3 -H \$HOSTADDRESS$ -c check\_cpu -a filter=none 'warn=load > \$ARG1$' 'crit=load > \$ARG2$'" -n cpu\_nrpe -t check

INFO: The CMD cpu\_nrpe is created

Modification d’une commande avec argument (attention aux antislash) :

debian@debian10:~/centctl$ ./centctl --server=centreon modify command -p command\_line -v "\$USER1$/check\_centreon\_nrpe3 -H \$HOSTADDRESS$ -c check\_cpu -a filter=none 'warn=load > \$ARG1$' 'crit=load > \$ARG2$'" -n cpu\_nrpe

INFO: The parameter command\_line of the CMD cpu\_nrpe is modified