Création commandes et services

Documentation qui traite de la création des commandes et services Nagios nécessaires à l'interrogation des éléments.

Idée générale

L'idée générale est de créer des commandes qui fonctionnent, ensuite de créer un service qui s'appuie sur cette commande et de faire en sorte que ce service fonctionne sur tout un groupe de machine. Le terme important est la **généricité**. Il faut créer des commandes qui fonctionnent sur tout les serveurs d'une même catégorie, mais éviter le plus possible des commandes du genre services pour machines SERVEUR1 et SERVEUR2, un autre pour SERVEUR3… Plus on créé de services individuels plus il est difficile de maintenir et de comprendre comment sont supervisés les hôtes.

Le fait de faire ca fonctionne mais devient ingérable, c'est pour ca qu'il faut créer des commandes génériques qui traitent le plus d'hôtes possibles voir même avec réellement **la totalité des hôtes**.

Pour créer les commandes, cela n'est pas aussi difficile que cela puisse paraître. Il est beaucoup plus facile de commencer à créer la commande en ligne de commande. Ensuite seulement, après avoir validé le fonctionnement du script et les bonnes options, on créera les commandes et les services à travers l'interface Centreon pour commencer la supervision.

Pour créer les commandes je conseille fortement de se connecter sur la machine avec une session commande et de tester les options des commandes à la main sur une machine spécifiée. Il existe une autre possibilité à travers l'interface web. On peut dans l'écran de modification d'une commande tout simplement l'exécuter. Elle offre tout de même moins de souplesse que la session commande.

Création de commande en console

Chaque commande propose une aide avec l'option `-h`, ce qui fournit la liste complète des possibilités de la commande avec une description de celle-ci. Toutes les commandes se situent dans le dossier `/usr/lib/nagios/plugins du serveur`.

Il faut savoir que la majorité des problèmes intervient sur la supervision des interfaces réseaux. Pour ce qui est du CPU, de la mémoire, de l'Uptime ou de la gestion des disques, la configuration est la même pour tout les Windows et tout les Linux. Il y a simplement une commande séparé pour chaque système d'exploitation car ils sont sensiblement différents.

Je prends donc un exemple avec le plugin d'interface réseau.

```
/usr/lib/nagios/plugins/check_centreon_snmp_traffic -h
```

L'option -h fonctionne avec tout les scripts (plugins) présents dans le dossier `/usr/lib/nagios/plugins/`.

Pour la configuration réseau, il est intéressant d'avoir la liste des interfaces réseaux d'un serveur ou d'un switch. De cette manière on valide que le SNMP est bien activé sur l'élément et on peut prendre connaissance de la liste de toutes les interfaces afin de simplement pouvoir sélectionner celles qui nous intéressent.

Commande qui permet de lister la liste des interfaces réseaux de l'élément avec l'adresse IP @IP

```
/usr/lib/nagios/plugins/check_centreon_snmp_traffic -H @IP -C
COMMUNAUTE_SERVEUR -v 2 -s
```

Pour sélectionner une interface par son nom et récupérer les informations, il faut ajouter `-i “nom interface” -n`

```
./check_centreon_snmp_traffic -H @IP -C COMMUNAUTE_SERVEUR -v 2 -i "nom
interface" -n
```

Pour d'autres commandes ou services, vous pouvez vous fier aux commandes déjà créées présentent dans Centreon ou dans le référentiel la fin de ce document.

Exemple concret

Exemple de création de la supervision des interfaces réseaux des serveurs en machine virtuelle.

J'ai listé les interfaces réseaux de quelques machines à travers le script check_snmp_netint.pl et je me suis rendu compte que le nom des interfaces étaient les mêmes : “VMware…”. Certains serveurs notamment les Windows 2008 disposent de cartes Intel et donc ne s'appellent pas Vmware mais Intel.

Un paramètre dans le script `check_snmp_netint.pl` permet d'effectuer une recherche sur le nom (description de l'interface réseau).

L'idée est de créé une expression régulière qui vérifie qu'il y a soit un nom VMware soit un nom Intel.

C'est ce qui a été fait pour pouvoir lister uniquement les interfaces réseaux que nous souhaitons. Cette configuration s'applique évidemment pour les serveurs Windows. Pour les serveurs Linux, les interfaces se nomment ethX.

Nous devons utiliser des expressions régulières pour une seule raison : Si on ne limite pas sur l'interface qui doit être active et qu'on en sélectionne des inactives, on aura une erreur critique dans Nagios qui les considérera comme DOWN. On a la possibilité d'inverser la sélection, c'est-à-dire de considérer normal que les interfaces soient DOWN mais cela s'applique à toutes les interfaces. On a le choix entre tout UP ou tout DOWN mais pas 3 interfaces UP et une DOWN par exemple.

Commande en SNMP pure avec check_snmp

Lorsqu'un élément support SNMP, on peut récupérer n'importe quelle valeur. Pour récupérer ces valeur, il faut utiliser le plugin check_snmp et connaître les OID exacts des valeurs de MIB à interroger (celles qui nous intéressent).

Pour interroger la MIB v2 d'un élément, afin de trouver un OID, il faut télécharger la MIB sur le site du

constructeur de l'élément (ou parfois sur l'élément lui-même) pour comprendre à quoi correspond les valeurs qu'on interroge et il faut taper une commande de ce type.

```
snmpwalk -v 2c -c COMMUNAUTE_SECU @IP:161  1.3.6.1.4
```

L'outil sous Linux est `snmpwalk`, `snmpget`... pour lire les valeurs et pour récupérer les informations. Sous Windows, utiliser **MIB Browser** (en graphique) est une bonne solution. Il faut ouvrir le fichier de MIB préallablement téléchargé, cliquer sur la valeur qu'on veut dans la MIB, et simplement lui donner l'adresse IP et la communauté SNMP de l'élément à interroger. En faisant un “get” ou un “Go” on obtient la valeur du champ et **l'OID exact à positionner dans la commande `check_snmp`**.

Avec snmpwalk, la succession de chiffres 1.3.6.1.4 limite l'interrogation à la MIB v2. Si on enlève les derniers chiffres, on a encore plus d'information.

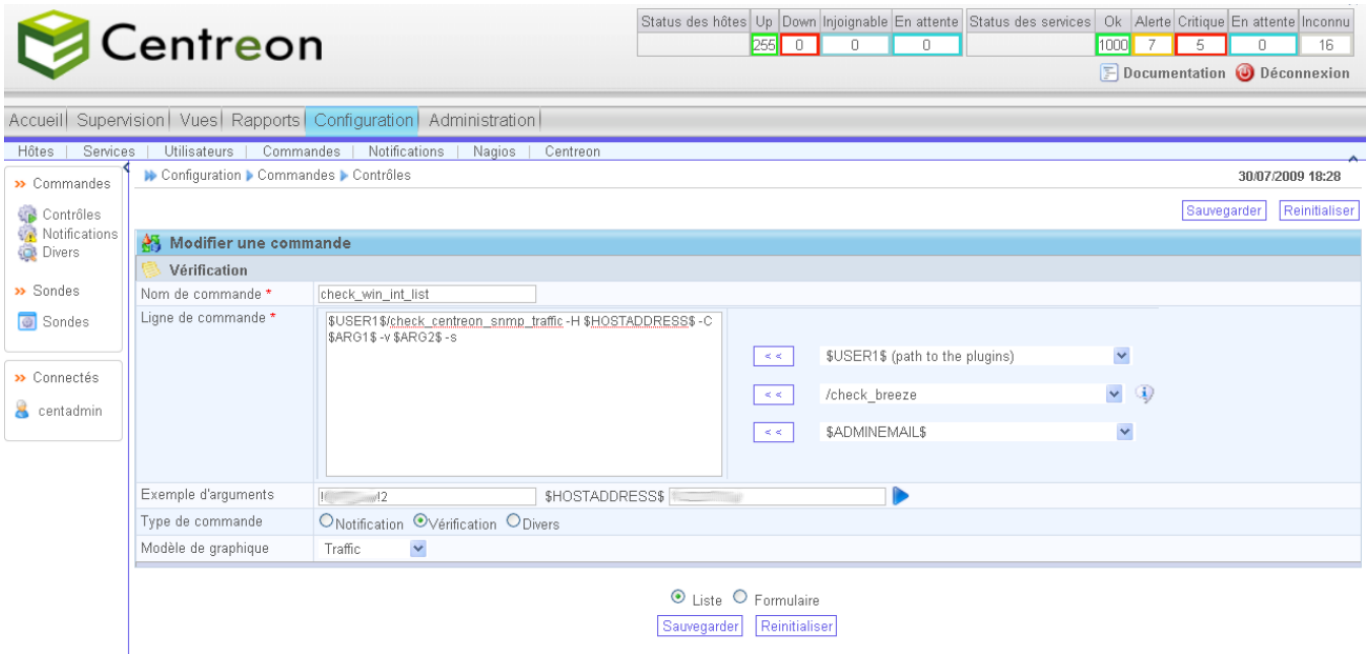
Autre exemple avec la librairie de sauvegarde.

```
snmpwalk -v 2c -c COMMUNAUTE_RESEAU @IP:161  1.3.6.1.4

snmpwalk -v 2c -c COMMUNAUTE_RESEAU @IP:161 -Os enterprises.3764.1.10.10.1
snmpwalk -v 2c -c COMMUNAUTE_RESEAU @IP:161 -Os
enterprises.3764.1.10.10.1.1.0
```

Création de commande en graphique sous Centreon

Pour faire la même chose en graphique sous Centreon, se rendre dans la configuration des commandes. Configuration > Commandes > Contrôles. Éditer la commande `check_win_int_list` si c'est un windows ou `check_lin_int_list` si c'est un Linux.



Dans cette écran on peut lister les interfaces réseaux de l'adresse IP qu'on spécifie dans HOSTADDRESS. Puis on exécute la commande avec le triangle.

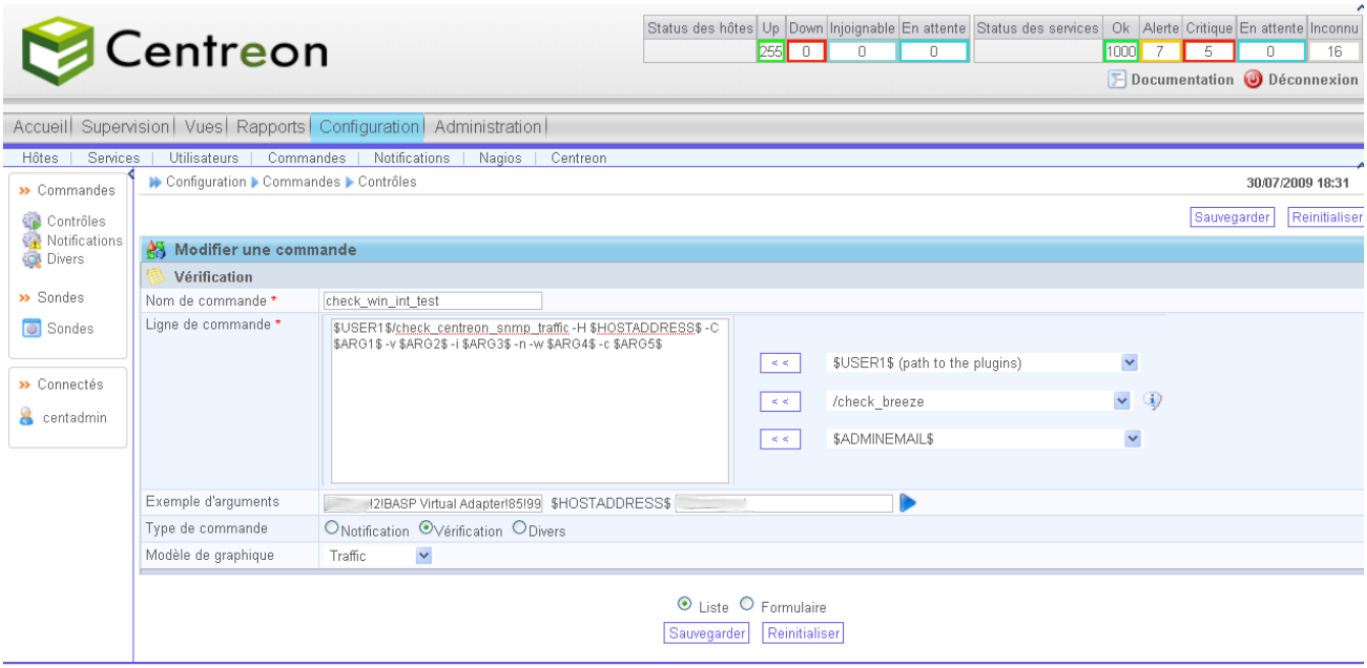
On obtient la liste des interfaces.

Test de la sonde

Test de la sonde

Ligne de commande	/usr/lib/nagios/plugins/check_centreon_snmp_traffic -H [redacted] -C '[redacted]' -v '2' -s
Sortie	Interface 1 :: MS TCP Loopback interface :: up Interface 65539 :: Intel(R) PRO/1000 PT Dual Port Server Adapter :: down Interface 65540 :: Intel(R) PRO/1000 PT Dual Port Server Adapter #2 :: down Interface 65541 :: BASP Virtual Adapter :: up
Statut	OK

Pour créer une nouvelle commande en testant des paramètres, sélectionner la commande nommée `check_win_int_test` afin de tester la remonté d'alerte sur le nom de la carte réseau obtenue dans la liste précédente.



On a la commande avec les arguments, séparés par des « ! » et enfin l'adresse IP de la machine. L'argument principal est le nom de la carte réseau. Ici c'est `BASP Virtual Adapter`.

Remarque : dans les paramètres, il ne faut pas mettre le nom de l'interface entre “”. Par contre la commande le nécessite quand même pour fonctionner.

Cliquer sur le triangle bleu pour exécuter la commande. Elle s'exécute dans une nouvelle fenêtre.

On obtient les informations de la carte réseau nommé.

Test de la sonde

Test de la sonde

Ligne de commande	/usr/lib/nagios/plugins/check_centreon_snmp_traffic -H [redacted] -C '[redacted]' -v '2' -i 'BASP Virtual Adapter' -n -w '85' -c '99'
Sortie	Traffic In : 16.71 kb/s (0.0 %), Out : 611.39 b/s (0.0 %) - Total RX Bits In : 19.54 GB, Out : 15.66 Gb|traffic_in=16707,6Bits/s;0;1000000000 traffic_out=611,4Bits/s;0;1000000000
Statut	OK

Référentiel des commandes utilisées

Se positionner dans le dossier `/usr/lib/nagios/plugins` pour exécuter les commandes suivantes. Je rappelle que chacune des commandes disposent d'une aide avec l'option `-h`.

Windows

CPU

```
./check_snmp_load -H @IP -C COMMUNAUTE_SERVEUR --v2c -T stand -w 85 -c 95 -f
```

RAM

```
./check_snmp_storage -H @IP -C COMMUNAUTE_SERVEUR --v2c -m "^Virtual
Memory$" -w 85 -c 99 -f
```

Disque

```
./check_snmp_storage.pl -H @IP -C COMMUNAUTE_SERVEUR --v2c -m
^[CDEFGHIJKLMNOPQRSTUVXYZ]: -w 85 -c 95 -f
```

Réseau

```
./check_centreon_snmp_traffic -H @IP -C COMMUNAUTE_SERVEUR  -v 2 -i "BASP
Virtual Adapter" -n -w 85 -c 100
```

Uptime

```
./check_snmp -H @IP -C COMMUNAUTE_SERVEUR  -P 2c -o sysUpTime.0
```

Services

Exemple avec service C2O serveur

```
./check_snmp_win.pl -H @IP -C COMMUNAUTE_SERVEUR -n "C2O"
```

Oracle

Exemple avec srv-dbigda

```
./check_snmp_win.pl -H @IP -C COMMUNAUTE_SERVEUR -n
"OracleOraDb10g_home1TNSListener$,OracleOraDb10g_home1TNSListenerBASE1,Oracl
eOraDb10g_home1TNSListenerBASE2,OracleOraDb10g_home1TNSListenerBASE3,OracleO
raDb10g_home1TNSListenerBASE4,OracleOraDb10g_home1TNSListenerBASE5,OracleOra
Db10g_home1TNSListenerBASE6,OracleOraDb10g_home1TNSListenerBASE7"
```

Exemple avec srv-ap09

```
./check_snmp_win.pl -H @IP -C COMMUNAUTE_SERVEUR -n
"OracleDBConsoleBASE1,OracleDBConsoleBASE2,OracleDBConsoleBASE3,OracleDBCons
oleBASE4,OracleDBConsoleBASE5,OracleDBConsoleBASE6,OracleDBConsoleBASE7,Orac
leDBConsoleBASE8,OracleOra102TNSListener,OracleServiceBASE1,OracleServiceBAS
E2,OracleServiceBASE3,OracleServiceBASE4,OracleServiceBASE5,OracleServiceBAS
E6,OracleServiceBASE7,OracleServiceBASE8"
```

Linux

Load

```
./check_snmp_load.pl -H @IP -C COMMUNAUTE_SERVEUR --v2c -T netsl -w
2,1.5,1.5 -c 3,2,2 -f
```

RAM

```
/check_snmp_mem.pl -H @IP -C COMMUNAUTE_SERVEUR --v2c -N -w 95,20 -c 99,70 -
f
```

Disque

```
./check_snmp_storage.pl -H @IP -C COMMUNAUTE_SERVEUR --v2c -m
"^/$|tmp|usr|var|data" -w 85 -c 95 -f
```

Réseau

```
./check_centreon_snmp_traffic -H @IP -C COMMUNAUTE_SERVEUR -v 2 -i "bond0" -
n -w 85 -c 99
```

Processus

Exemple avec processus C2O linux.

```
./check_snmp_process.pl -H @IP -C COMMUNAUTE_SERVEUR -2 -n "c2ondmgr"  -w
2,10 -c 0,10
./check_snmp_process.pl -H @IP -C COMMUNAUTE_SERVEUR -2 -n "c2o" -w 2,7 -c
0,8
```

Uptime

```
./check_snmp -H @IP -C COMMUNAUTE_SERVEUR -P 2c -o sysUpTime.0
```

ESX

Load

```
./check_snmp_load.pl -H @IP -C COMMUNAUTE_SERVEUR --v2c -T netsl -w 2,1,1 -c
3,2,2 -f
```

RAM

```
./check_snmp_mem.pl -H @IP -C COMMUNAUTE_SERVEUR –v2c -N -w 95,20 -c 99,70 -
f
```

Disque

```
./check_snmp_storage.pl -H @IP -C COMMUNAUTE_SERVEUR --v2c -m ^/$|var/log -w
85 -c 95 -f
```

Réseau

```
./check_centreon_snmp_traffic -H @IP -C COMMUNAUTE_SERVEUR -v 2 -r
"vmnic[0-3]" -w 80 -c 90
```

Uptime

```
./check_snmp -H @IP -C COMMUNAUTE_SERVEUR -P 2c -o sysUpTime.0
```

Routeurs

Ping

```
./check_centreon_ping -H @IP -n 1 -w 80,20% -c 150,60%
```

Température et ventilateur

Voici ce que récupère la commande 'check_snmp_env'' avec le type cisco.

```
cisco : All Cisco equipements : voltage,temp,fan,power supply
 (will try to check everything in the env mib)
```

Ces commandes vérifient notamment la température des alimentations ainsi que leur fonctionnement et la vitesse des ventilateurs. Cela s'applique sur du matériel Cisco.

```
./check_snmp_env.pl -H @IP -C COMMUNAUTE_RESEAU -T cisco -F -c 60
./check_snmp_env.pl -H @IP -C COMMUNAUTE_RESEAU -T cisco -F -c 60
./check_snmp_env.pl -H @IP -C COMMUNAUTE_RESEAU -T cisco -F -c 60
```

Réseau

```
./check_snmp_netint.pl -H @IP -C COMMUNAUTE_RESEAU --v2c -n
"FastEthernet0/0|FastEthernet0$|Ethernet1/0" -f -k -Y -w 600,600 -c
1000,1000
```

Robot de sauvegarde

Exemple avec le valeurs de MIB récupérable sur un robot de sauvegarde Quantum Scalar i500.

libraryGlobalStatus

DESCRIPTION “Current status of the entire library system (including all attached drive).”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.1.8.0 -s 1 -l "LibraryGlobalStatus :"
```

overallPhDriveReadinessStatus

DESCRIPTION “Overall Drives readiness.”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.11.1.0 -s 1 -l "OverallPhDriveReadinessStatus :"
```

phDriveRasStatus

DESCRIPTION “Drive health status.”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.11.3.1.11.1 -s 1 -l "PhDriveRasStatus :"
```

phDriveNeedsCleaning

DESCRIPTION “Cleaning status of the Drive.”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.11.3.1.12.1 -s 2 -l "PhDriveNeedsCleaning :"
```

powerStatus

DESCRIPTION “Indicates overall power supply Status”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.12.1.0 -s 1 -l "PowerStatus :"
```

coolingStatus

DESCRIPTION “Indicates overall cooling fans Status.”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.12.2.0 -s 1 -l "CoolingStatus :"
```

roboticsStatus

DESCRIPTION “Indicates overall robotics Status”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.12.5.0 -s 1 -l "RoboticsStatus :"
```

driveStatus

DESCRIPTION “Indicates overall Drives Status”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.12.7.0 -s 1 -l "DriveStatus :"
```

physicalLibraryState

DESCRIPTION “This indicates Physical Library's overall Online Status”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.14.1.0 -s 1 -l "PhysicalLibraryState :"
```

robotState

DESCRIPTION “Device SCSI state.”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.14.30.2.0 -s 1 -l "RobotState :"
```

mediaStatus

DESCRIPTION “Indicates overall media Status”

```
./check_snmp -H @IP -C COMMUNAUTE_RESEAU -P2c -o
.1.3.6.1.4.1.3764.1.10.10.12.6.0 -s 1 -l "MediaStatus :"
```

Sécurité

Checkpoint

Status (actif ou standby) de deux pare-feu Checkpoint.

```
./check_snmp_cpfw.pl -H @IP -2 -C COMMUNAUTE_SECU -a -ws
./check_snmp_cpfw.pl -H @IP -2 -C COMMUNAUTE_SECU -a standby -ws
```

Interfaces réseaux des VLAN dans lesquels passe le trafic.

```
./check_snmp_netint.pl -H @IP -C COMMUNAUTE_SECU -2 -n "eth3.10[014]" -f -k
-w 15000,15000 -c  25000,25000
./check_snmp_netint.pl -H @IP -C COMMUNAUTE_SECU -2 -n "eth3.10[014]" -f -k
-w 15000,15000 -c  25000,25000

./check_snmp_netint.pl -H @IP -C COMMUNAUTE_SECU -2 -n "eth4.10[23]" -f -k -
w 15000,15000 -c  25000,25000
./check_snmp_netint.pl -H @IP -C COMMUNAUTE_SECU -2 -n "eth4.10[23]" -f -k -
w 15000,15000 -c  25000,25000
```

Espace de /var pour superviser la taille des logs.

```
./check_snmp_storage.pl -H @IP -m "^/var$" -C COMMUNAUTE_SECU -w 80 -c 90
```

Iron Port

CPU

```
./check_snmp -H @IP -o .1.3.6.1.4.1.15497.1.1.1.1.0 -w 0:80 -c 0:90 -C
COMMUNAUTE_SECU -P2c -l "%CPU="
```

Power Supply

```
./check_snmp_env.pl -H @IP -C COMMUNAUTE_SECU -T iron -F -c 60
./check_snmp_env.pl -H @IP -C COMMUNAUTE_SECU -T iron -F -c 60
```

Interface

```
./check_snmp_int.pl -H @IP -2 -C COMMUNAUTE_SECU -n Data
./check_snmp_int.pl -H @IP -2 -C COMMUNAUTE_SECU -n Data
```

Status de la queue

```
./check_snmp -H @IP -o .1.3.6.1.4.1.15497.1.1.1.5.0 -w 0:80 -c 0:90 -C
COMMUNAUTE_SECU -P2c -l "%QueueUtilization="
```

Utilisation de la queue.

```
./check_snmp -H @IP -o .1.3.6.1.4.15497.1.1.1.1.4.0 -w 0:80 -c 0:90 -C
COMMUNAUTE_SECU -P2c -l "%QueueUtilization="
```

Bluecoat

Température, alimentations, ventilateurs, nombre de disques en ligne.

```
./check_snmp_env.pl -H @IP -C COMMUNAUTE_SECU -T bc
```

IWSA

```
./check_snmp_netint.pl -H @IP -C COMMUNAUTE_SECU -n eth0
```

Port particulier sur un switch qui contient tout le trafic Web

```
./check_snmp_netint.pl -H @IP -C COMMUNAUTE_SECU --v2c -n 19 -f -k -Y -w 6,6
-c 8,8
```

Avec l'interface Centreon

Lancer l'interface web

```
http://serveur.domaine.local/centreon
```

Aller dans Configuration>Commandes et Configuration>Services pour ajouter les commandes et services testées. Il ne restera plus qu'à les appliquer sur des groupes d'hôtes ou des machines en particuliers pour que les services commencent à checker et grapher un peu plus tard.

Ne pas oublier après avoir ajouté un hôte d'exporter la configuration Nagios dans Configuration>Nagios>Generate.

Tout cocher et redémarrer Nagios.