

Supervision de services sous Linux et Windows



OBJECTIF :

Pouvoir réaliser (et donc superviser) des 'checks' sur les services actifs sur tous les OS (Windows, Linux et même MacOS).

Il sera donc possible de vérifier en temps réel si tous nos services sont bien actifs.

REDACTION : février 2020

VERSION : 1.0

PAR : GUENAT Lilian (Stagiaire)

Sommaire

1	INSTALLATION	3
1.1	Avec Windows	3
1.2	Avec Linux (ici CentOS 7)	4
2	CONFIGURATION.....	5

Un service actif, quel que soit l'OS doit pouvoir être supervisé. Pour se faire nous allons utiliser dans cette documentation le logiciel/service NCPA qui nous permet de réaliser cette tâche. Il sera donc possible de vérifier si nos services sont bien en fonctionnement sur les différentes machines de notre réseau.

Remarque : NCPA n'est pas le seul agent actif pouvant fonctionner avec Nagios. Ci-dessous un tableau comparatif des différents outils existant dans la sphère de Nagios.

Agent Comparison:

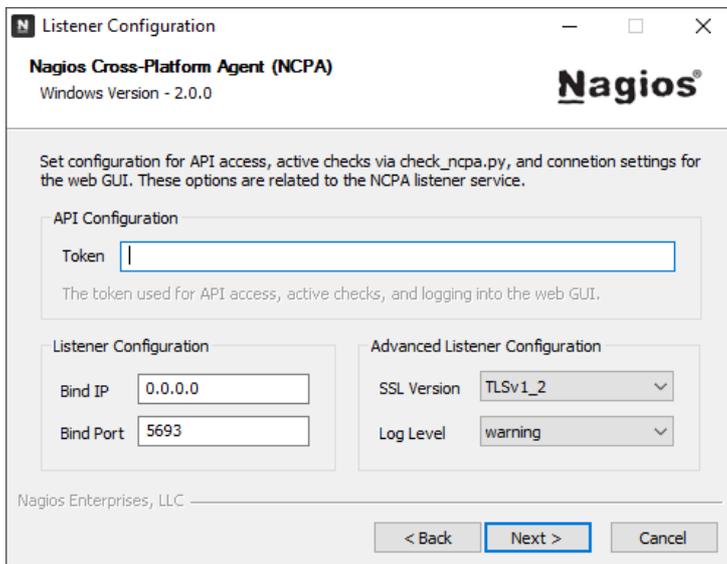
Features \ Edition	NCPA	NRDS Agent	NSClient ++	NRPE
Installs on Linux	✓	✓		✓
Installs on Windows	✓	✓	✓	
Installs on Mac OSX	✓	✓		✓
Graphical User Interface	✓			
Active Check Metrics	✓		✓	✓
Passive Check Capabilities	✓	✓	✓	
Flexible API Access	✓			
Seamless Integration with Nagios XI	✓		✓	✓
Integration with Nagios Core via NRDP	✓	✓	✓	
Official Nagios Enterprises Monitoring Agent	✓			
Pre-configured Monitoring Metrics	✓		✓	✓
Integration with NRDS Configuration Protocol	✓	✓		

1 INSTALLATION

1.1 Avec Windows

Pour commencer, nous allons devoir télécharger l'exécutable sur le site de Nagios, section NCPA juste ici : <https://www.nagios.org/nca/#downloads>.

Nous avons besoin d'indiquer une clé à un seul endroit, alors passez toutes les étapes jusqu'ici :



A cet endroit, indiquez dans la 1^{ère} case le token/clé que vous souhaitez. Retenez-le bien nous en aurons besoin pour notre commande plus tard.

Dans notre cas, le token est : **EONAxBBKey**, il s'agit du même que nos workers afin qu'on ait qu'un seul token à retenir.

Ensuite ne touchez rien d'autre, puis cliquez sur 'Installer'.

1.2 Avec Linux (ici CentOS 7)

Sous Linux, c'est un peu différent, le package NCPA n'est pas disponible dans les repos par défaut. Pour se faire, ajoutez le repo avec cette commande :

- ▶ `rpm -Uvh https://repo.nagios.com/nagios/7/nagios-repo-7-4.el7.noarch.rpm`

A présent, vous pouvez installer le package avec la commande suivante :

- ▶ `yum install ncpa`

NCPA est maintenant bien installé sur votre machine Linux, rien de très compliqué. Il faut maintenant mettre notre clé/token comme sous Windows. Ici, il se situe dans le fichier de configuration suivant :

- ▶ `vi /usr/local/ncpa/etc/ncpa.cfg`

Maintenant, cherchez la ligne suivante, « **community_string=** », normalement elle se situe sous la petite balise **[api]** qui est également dans le fichier. Effacez la valeur déjà présente puis indiquez le token que vous avez choisi. Pour nous ce sera toujours **EONAxBBKey**.

Sauvegardez votre fichier (:x ou :wq) puis il faut à présent redémarrer le service pour bien que notre token soit pris en compte.

- ▶ `service ncpa_listener restart` (ou `/etc/init.d/ncpa_listener restart`)

Notre NCPA est opérationnel. Passons maintenant à la configuration.

2 CONFIGURATION

Nos services NCPA sont fonctionnels sous Windows comme sous Linux. A présent, retour sur notre worker afin d'ajouter la commande qui lui permettra de lancer des checks.

Rendez-vous dans le répertoire de Nagios, à l'endroit où se situe les différentes commandes :

- ▶ `cd /srv/eyesofnetwork/nagios/plugins`

A présent, vous pouvez télécharger la dernière version du script ici :

- ▶ `wget https://raw.githubusercontent.com/HookDonn/Axians-Scripts/master/check_ncpa.py`

Donnez-lui les permissions nécessaires :

- ▶ `chmod 755 check_ncpa.py`
- ▶ `chown nagios:eyesofnetwork check_ncpa.py`

Le script est maintenant bien installé et opérationnel. Passons maintenant à la configuration sur EyesOfNetwork.

Nous allons commencer par créer la commande, pour se faire rendez-vous dans « Administration », « Configurations Nagios » puis cliquez sur « Nagios Commands » :

Il faut maintenant cliquer sur le bouton

Add A New Command

afin de créer notre nouvelle commande.

Remplissez la commande comme ceci :

Command Name:
check_ncpa ⓘ

Command Line:
\$USER1\$/check_ncpa.py -H \$HOSTADDRESS\$ \$ARG1\$ ⓘ

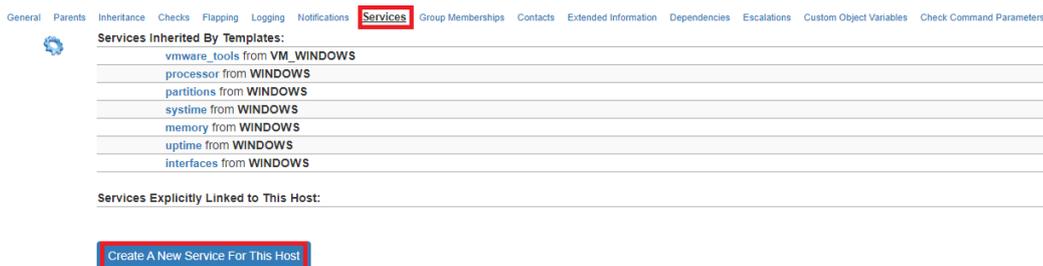
Command Description:
Check services Windows & Linux ⓘ

Cliquez sur le bouton pour sauvegarder. Nous pouvons maintenant commencer à mettre en place le 'check' de service sur nos hôtes.

Retournez donc sur la page d'accueil « Eonweb Configurator » dans « Administration » -> « Configuration Nagios ».

Maintenant, cliquez sur « Equipements » puis sur « Lister » en haut à droite. Rendez-vous sur un hôte dont vous voulez superviser un service (Linux ou Windows, comme vous voulez, la commande fonctionne exactement de la même façon peu importe l'OS). Puis, cliquez sur l'onglet « Services ».

Host Editor



General Parents Inheritance Checks Flapping Logging Notifications **Services** Group Memberships Contacts Extended Information Dependencies Escalations Custom Object Variables Check Command Parameters

Services Inherited By Templates:

- vmware_tools from VM_WINDOWS
- processor from WINDOWS
- partitions from WINDOWS
- systime from WINDOWS
- memory from WINDOWS
- uptime from WINDOWS
- interfaces from WINDOWS

Services Explicitly Linked to This Host:

Create A New Service For This Host

A présent, cliquez sur "Create A New Service For This Host" pour ajouter un nouveau service à notre hôte.

Puis complétez comme ici en adaptant le service que vous souhaitez check. Ici, nous faisons un test sur le service DNS.

Service Editor

Search:

Service Description:

Display Name: (Optional)

Service Templates To Inherit From (Top to Bottom):
Delete GENERIC_SERVICE

Add Template To Inherit From: Add Template

Check Command: Provide Value

Check Command Parameters:
Delete \$ARG1\$: -t 'EONAxBBKey' -P 5693 -M 'services' -q 'service=Dnscache,status=running'

Value for \$ARG2\$: Add Parameter

Add Service Cancel

Pensez bien à ajouter un template, la commande « check_ncpa » ainsi que l’argument 1 :

▶ -t 'EONAxBBKey' -P 5693 -M 'services' -q 'service=Dnscache,status=running'

Changez bien le ‘Dnscache’ par le service que vous voulez.

La ligne d’arguments se décompose comme suit :

- ▶ « -t 'EONAxBBKey' » : la community_string renseignée dans le fichier de config de l’agent sur le serveur.
- ▶ « -P 5693 » : le port utilisé par l’agent NCPA, en l’occurrence le port 5693, par défaut.
- ▶ « -M 'services' » : le Metric à interroger => dans ce cas un service.
- ▶ « -q 'service=Dnscache,status=running' » : le nom du service à intérogger, ici ‘Dnscache’ ainsi que le status qu’il doit avoir, ici ‘running’ donc, en marche.

Le statut peut être « running », comme ci-dessus, ou « stopped ».

Si le status a la valeur « stopped », le plugin renverra Ok si le service est arrêté, et Critical s’il est démarré.

A présent cliquez sur « Add Service ». Il faut maintenant bien penser à appliquer la configuration afin que les ajouts soient pris en compte.

Si vous avez tout bien suivi, vous devriez à présent voire le status du service que vous avez ajouté :

DNS	OK	17:02:25	0d 1h 29m 49s	1/4	OK: Dnscache is running
-----	-----------	----------	---------------	-----	-------------------------

Vous pouvez faire de même avec vos autres hôtes.