

Université IBM i 2018

16 et 17 mai

IBM Client Center Paris



Session S53 – Monitoriez votre IBM i avec Nagios

Christian Massé
cmasse@volubis.fr

Pascal Ruckebusch
pruckebusch@m81.fr

Plan de la présentation



- Principe de fonctionnement de Nagios
- Les plugins Nagios pour IBM i
- La solution basée sur l'utilisation d'un agent

Principe de fonctionnement de Nagios

Surveiller l'infrastructure



Historique de Nagios

- Logiciel de supervision de réseaux
- Créé en 1999 par Ethan Galstad
- Considéré comme la référence des solutions de supervision Open Source
- Outil très complet
 - Permet de s'adapter à n'importe quel type d'utilisation
 - Possibilités de configuration très poussées
 - Grande modularité et forte communauté
- Permet aujourd'hui de superviser pratiquement n'importe quelle ressource.

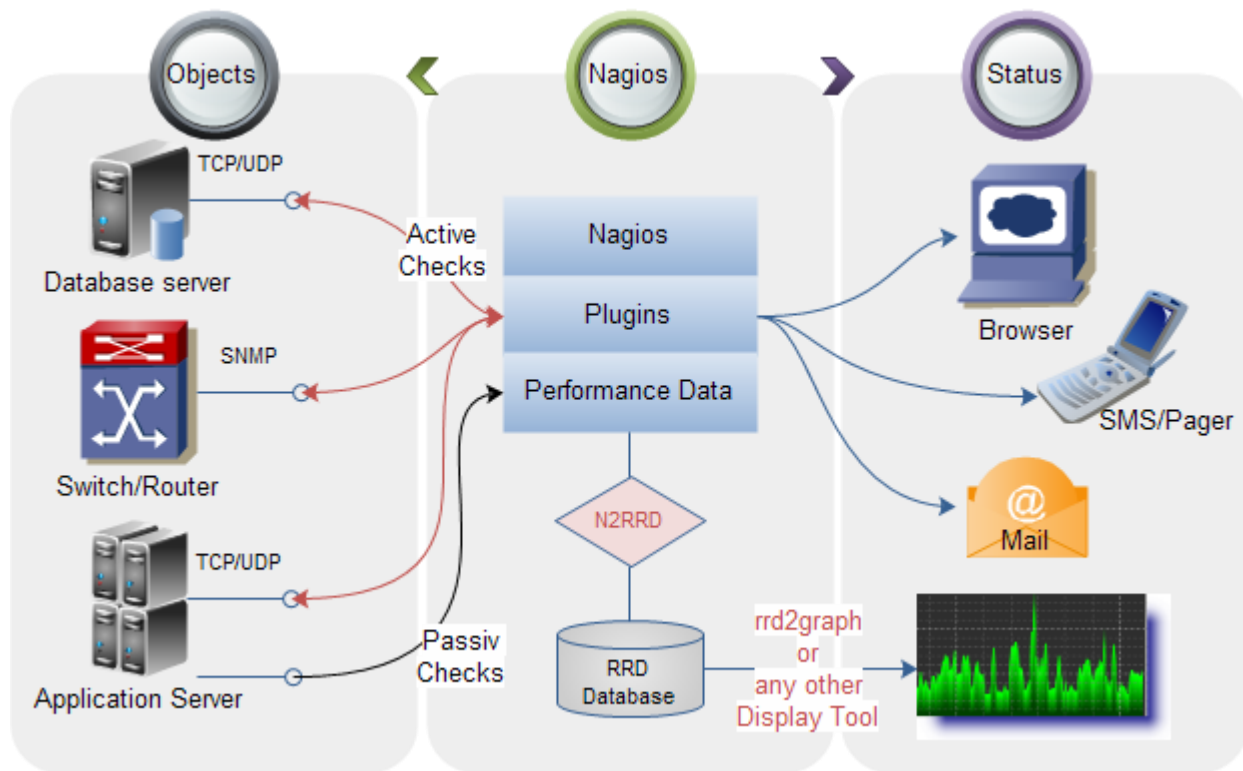
(Sources Wikipédia)

Le fonctionnement

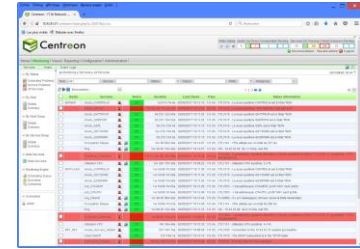
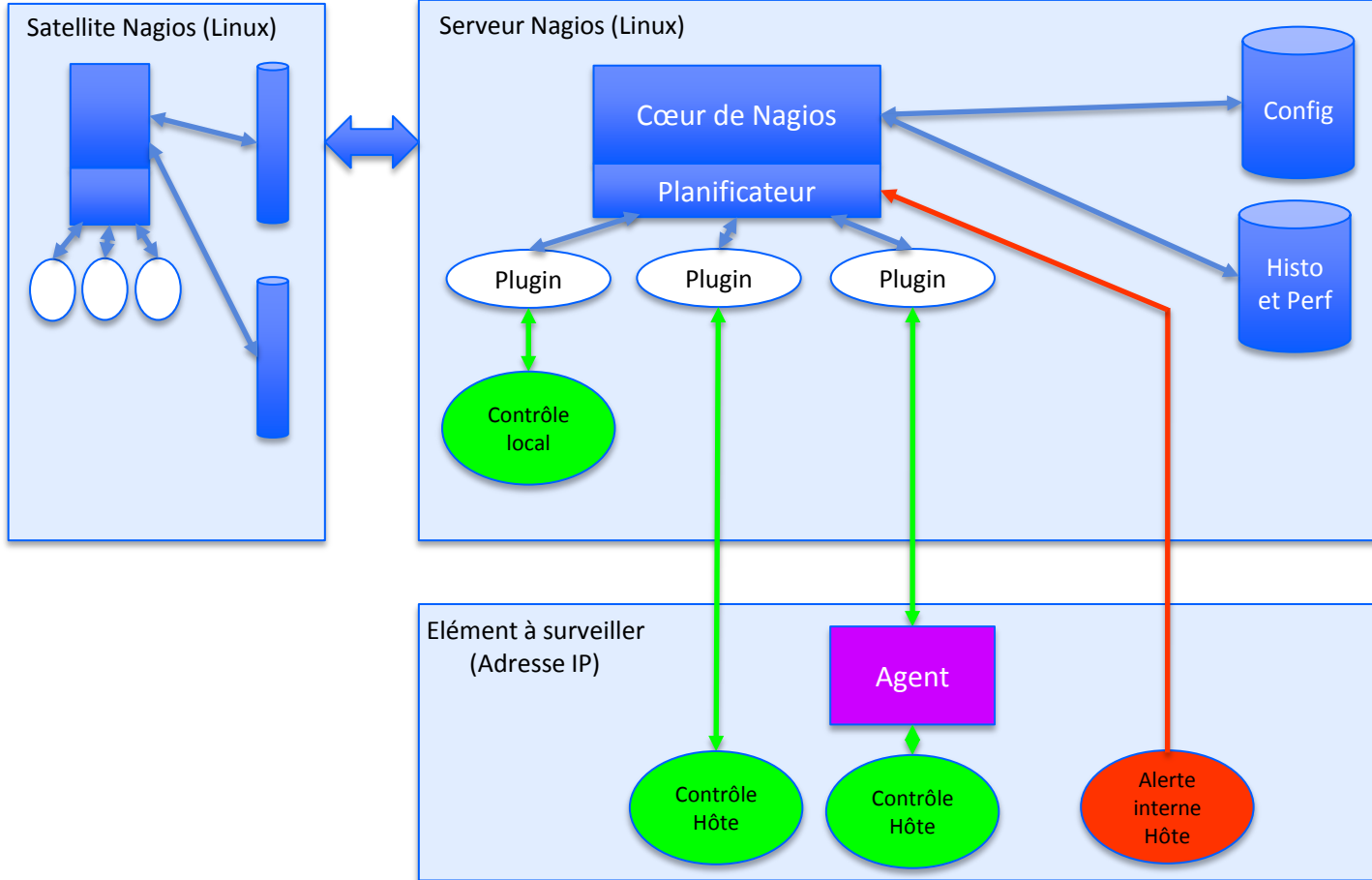


- Nagios apporte les fonctionnalités suivantes
 - Surveillance de tous les éléments dans le réseau (basé sur les adresses IP)
 - Contrôles
 - Actifs (pilotés par Nagios)
 - Passifs (pilotés par l'élément surveillé)
 - Définition des plages horaires pour les contrôles
 - Suspension temporaire ou permanente des contrôles
 - Visualisation des résultats
 - Envoi d'alertes (mails, sms, ...)
 - Regroupement des ressources (groupe d'hôtes, groupes de services) pour une meilleure visibilité
 - Possibilité de hiérarchiser les ressources

Le fonctionnement



Le fonctionnement



Le fonctionnement

- Les plugins
 - Programmes situés sur le serveur Nagios (sous Linux)
 - Développés dans des langages tels que java, C, C++, perl, PHP, ...
 - Utilisent un protocole pour se connecter à l'élément à surveiller
 - icmp (Ping)
 - snmp (utilisé le plus souvent)
 - ssh
 - ftp
 - telnet
 - Spécifique lors de la présence d'un agent sur l'élément à surveiller
 - Exemple
 - `check_icmp -H 10.55.1.3 -n 5 -w 200,20% -c 400,50%`

Le fonctionnement

Nagios

- Beaucoup d'autres produits sont basés sur le même principe que Nagios
- Ou même sont des ajouts à Nagios



Etc ...

Visualisation des résultats


















By Status

- Services
- Hosts
- Services Grid
- Services by Hostgroup
- Services by Servicegroup
- Hostgroups Summary

Monitoring > Status Details > Services

Service Status	Status	Poller			
<input type="text" value="All"/>	<input type="text"/>	<input type="text"/>			
Host	Service	Hostgroup	Servicegroup	Output	
<input type="text" value="f"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

More actions...    1 2 3 4 5 6 7 8 9 10 11 > >> 100

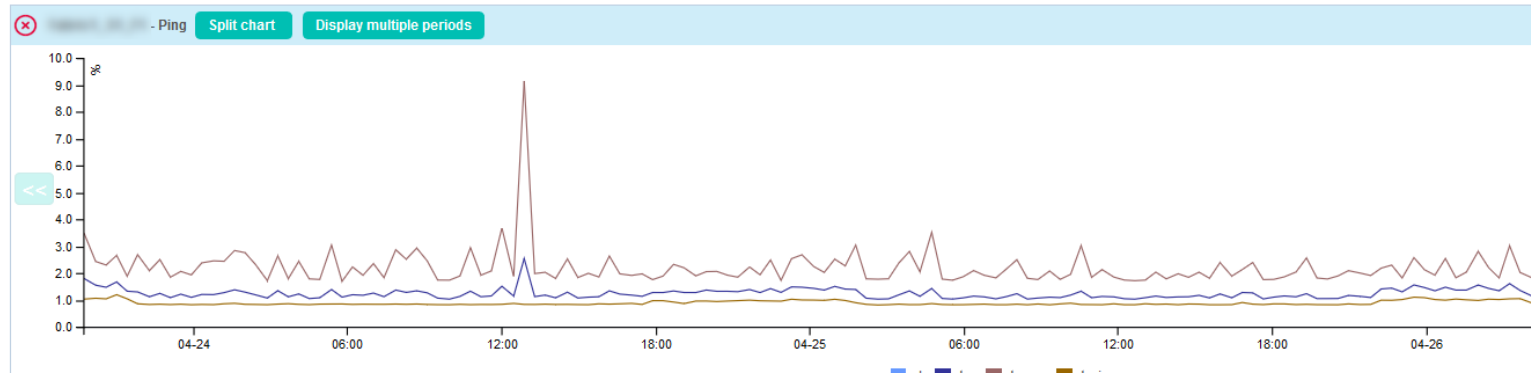
<input type="checkbox"/>	Hosts ^	Services	Status	Duration	Last Check	Tries	Status information
<input type="checkbox"/>	192.168.1.1	Ping	 OK	1w 2d	37s	1/3 (H)	OK - 109.0.86.11: rta 1,117ms, lost 0%
<input type="checkbox"/>	192.168.1.1	Ping	 OK	1w 2d	1m 3s	1/3 (H)	OK - 109.0.86.21: rta 1,406ms, lost 0%
<input type="checkbox"/>	192.168.1.1	Hardware	 OK	1w 4d	10s	1/3 (H)	OK: All 6 components are ok [5/5 sensors, 1/1 switch].
<input type="checkbox"/>	192.168.1.1	Ping	 OK	6M 3w	41s	1/3 (H)	OK - 172.16.88.80: rta 0,113ms, lost 0%
<input type="checkbox"/>	192.168.1.1	Traffic-Global	 OK	1w 1d	1m 24s	1/3 (H)	OK: All interfaces are ok
<input type="checkbox"/>	192.168.1.1	Hardware	 OK	11h 53m	33s	1/3 (H)	OK: All 6 components are ok [5/5 sensors, 1/1 switch].
<input type="checkbox"/>	192.168.1.1	Ping	 OK	6M 3w	1m 45s	1/3 (H)	OK - 172.16.84.1: rta 0,512ms, lost 0%
<input type="checkbox"/>	192.168.1.1	Traffic-Global	 OK	56m 37s	32s	1/3 (H)	OK: All interfaces are ok
<input type="checkbox"/>	192.168.1.1	Hardware	 OK	3d 22h	10s	1/3 (H)	OK: All 6 components are ok [5/5 sensors, 1/1 switch].
<input type="checkbox"/>	192.168.1.1	Ping	 OK	4M 1w	34s	1/3 (H)	OK - 172.16.88.81: rta 0,142ms, lost 0%
<input type="checkbox"/>	192.168.1.1	Traffic-Global	 OK	3d 12h	1m 24s	1/3 (H)	OK: All interfaces are ok
<input type="checkbox"/>	192.168.1.1	Hardware	 OK	11h 53m	34s	1/3 (H)	OK: All 6 components are ok [5/5 sensors, 1/1 switch].
<input type="checkbox"/>	192.168.1.1	Ping	 OK	6M 3w	1m 44s	1/3 (H)	OK - 172.16.84.2: rta 0,420ms, lost 0%
<input type="checkbox"/>	192.168.1.1	Traffic-Global	 OK	1h 28m	32s	1/3 (H)	OK: All interfaces are ok
<input type="checkbox"/>	192.168.1.1	Ping	 OK	2d 16h	44s	1/3 (H)	OK - 172.16.91.12: rta 0,344ms, lost 0%
<input type="checkbox"/>	192.168.1.1	Cpu	 CRITICAL	7m 25s	1m 25s	3/3 (H)	CRITICAL: 2 CPU(s) average usage is: 92.50%
<input type="checkbox"/>	192.168.1.1	Disk-Global	 OK	16h 27m	51s	1/3 (H)	OK: Storage 'C:' Usage Total: 273.24 GB Used: 51.78 GB (18.95%) Free: 221.46 GB (81.05%)

Différents niveaux et options de reporting

Monitoring > Performances > Graphs

Chart Period From

Filter by Host Filter by Hostgroup Filter by Servicegroup

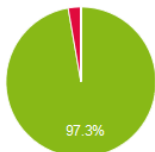


Différents niveaux et options de reporting



Reporting > Dashboard > Hosts

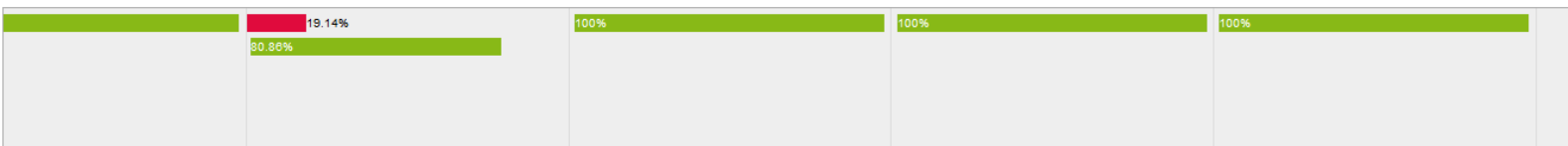
Host: From 19/04/2018 00:00 to 26/04/2018 00:00 Reporting Period: Last 7 Days From to



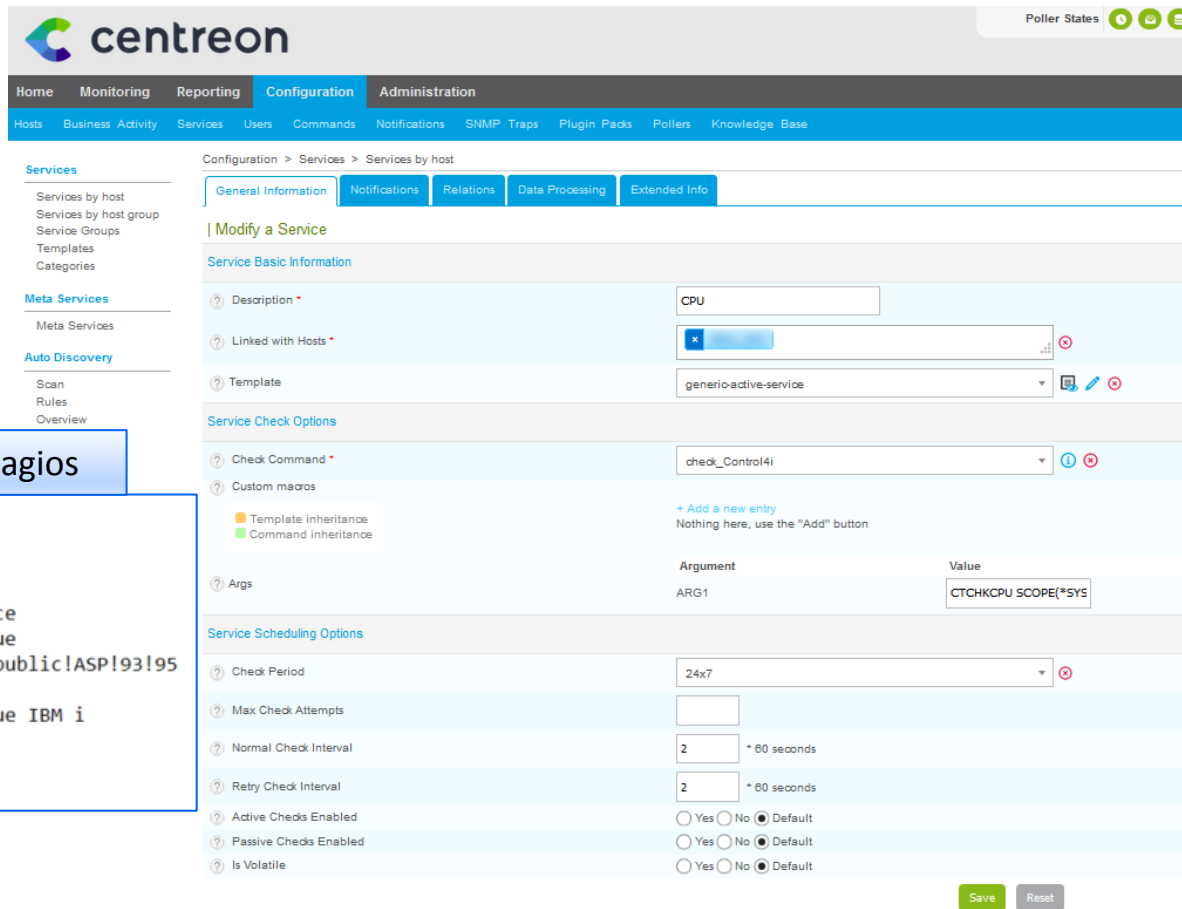
■ Up
 ■ Down
 ■ Unreachable
 ■ Downtime
 ■ Undetermined

State	Duration	Total Time	Mean Time	Alerts
UP	6d 19h 24m 25s	97.27%	97.27%	1
DOWN	4h 35m 35s	2.73%	2.73%	1
UNREACHABLE		0%	0%	0
SCHEDULED DOWNTIME		0%		
UNDETERMINED		0%		
Total	7d			2

Service	OK		Warning		Critical		Unknown		Scheduled downtime		Undetermined
	%	Alert	%	Alert	%	Alert	%	Alert	%	%	
Control4I_CPU	95.97% (95.97%)	32	0% (0%)	0	4.03% (4.03%)	32	0% (0%)	0	0%	0%	0%
CPU	97.24% (97.24%)	1	0% (0%)	0	2.76% (2.76%)	1	0% (0%)	0	0%	0%	0%
Espace_Disque	97.24% (97.24%)	1	0% (0%)	0	2.76% (2.76%)	1	0% (0%)	0	0%	0%	0%
Messages_sous_systeme_du_Systeme	83.86% (83.86%)	6	0% (0%)	0	16.14% (16.14%)	6	0% (0%)	0	0%	0%	0%
Ping	97.23% (97.23%)	1	0.02% (0.02%)	1	2.76% (2.76%)	1	0% (0%)	0	0%	0%	0%
Problemes_materiel	97.23% (97.23%)	1	0% (0%)	0	2.77% (2.77%)	1	0% (0%)	0	0%	0%	0%
Sous_systeme_QINTER	58.8% (58.8%)	6	0% (0%)	0	41.2% (41.2%)	13	0% (0%)	0	0%	0%	0%
Sous_systeme_QINTER1	97.23% (97.23%)	1	0% (0%)	0	2.77% (2.77%)	1	0% (0%)	0	0%	0%	0%
Average	90.6% (90.6%)	49	0% (0%)	1	9.4% (9.4%)	30	0% (0%)	0	0%	0%	0%



Configuration



Configuration > Services > Services by host

General Information | Notifications | Relations | Data Processing | Extended Info

Modify a Service

Service Basic Information

Description * CPU

Linked with Hosts * [x] [red circle]

Template generic-active-service [red circle]

Service Check Options

Check Command * check_Control4i [red circle]

Custom macros

Template inheritance (orange square) Command inheritance (green square)

+ Add a new entry
Nothing here, use the "Add" button

Argument	Value
ARG1	CTCHKCPU SCOPE(*SYS)

Service Scheduling Options

Check Period 24x7 [red circle]

Max Check Attempts [input]

Normal Check Interval 2 * 60 seconds

Retry Check Interval 2 * 60 seconds

Active Checks Enabled Yes No Default

Passive Checks Enabled Yes No Default

Is Volatile Yes No Default

Save Reset

Définition en mode texte pour Nagios

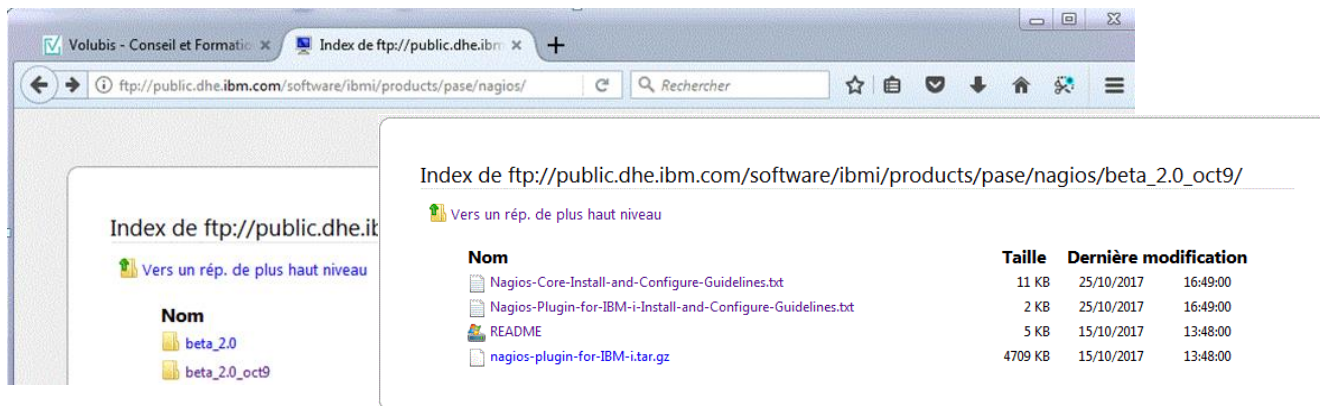
```
#####  
# Occupation disque.  
#####  
define service{  
    use                local-service  
    name               Espace disque  
    check_command      check_disk!public!ASP!93!95  
    hostgroup_name     IBM_i  
    service_description Espace disque IBM i  
    check_period       24x7  
}
```

Les plugins Nagios pour IBM i

Plugin Nagios pour IBMi

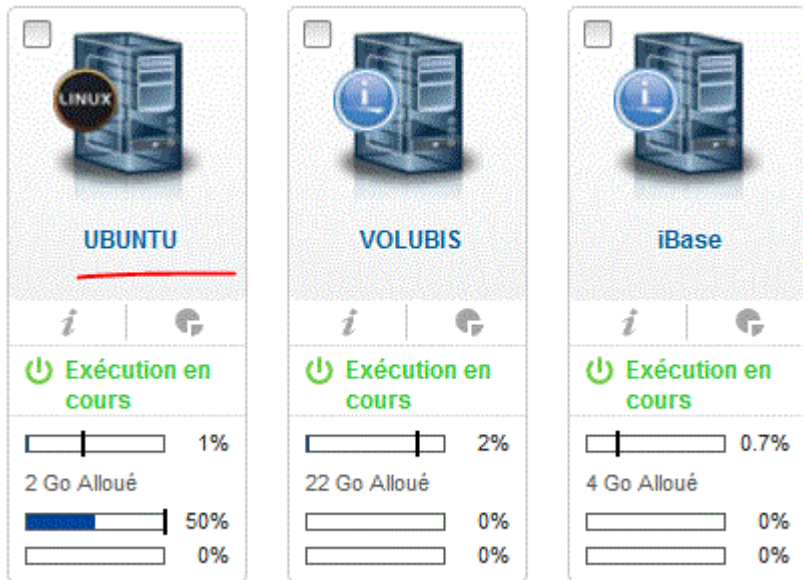
- Le plugin fourni par IBM :

- Est en Béta
- Est documenté sur <https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/IBM%20i%20Technology%20Updates/page/Nagios%20plugin%20support%20for%20IBMi>
- Est pour l'instant disponible à <ftp://public.dhe.ibm.com/software/ibmi/products/pase/nagios/>



Plugin Nagios pour IBMi

- Nous l'avons installé sur une partition Ubuntu for Power



Plugin Nagios pour IBMi



- Installation standard

```
=====
=====Install the Nagios plugin for i=====
=====
```

1. Extract the Nagios for i plugin to the directory: /usr/local/nagios/

```
# tar xzf nagios-plugin-for-IBM-i.tar.gz /usr/local/nagios/
```

2. Execute the below command:

```
# /bin/bash /usr/local/nagios/libexec/ibmi_init.sh.
```

It will initialize the IBM i related commands, services and templates into the configuration

3. Issue below command to add the user name and password of the system that you would like to monitor (you can use -h option to see the help):

```
# /bin/bash /usr/local/nagios/libexec/host_config.sh -i [host | sst]
```

4. Start the server of plugin for i and restart the nagios service

```
# /bin/bash /usr/local/nagios/libexec/server_start.sh
# service nagios restart
```

Plugin Nagios pour IBMi

- Quelques remarques pour une installation sur Ubuntu

(en effet l'installation part du principe que nagios est dans /usr/local/nagios , or sur Ubuntu c'est dans /etc/nagios3)

- Après avoir restauré le fichier .tar
- passez **ibmi_init.sh -p /etc/nagios3/objects.**

- Puis passez les commandes
 - **host_config.sh -i host.**
pour enregistrer un profil(et son mot de passe) pour accéder à votre IBM i.
 - **host_config.sh -i sst**
pour enregistrer un profil SST (et son mot de passe) pour accéder aux informations disques de votre IBM i.

Plugin Nagios pour IBMi

- Quelques remarques pour une installation sur Ubuntu

```
cm@Ubuntu: /etc/nagios3/libexec
Using username "cm".
cm@10.3.1.59's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic ppc64le)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
Last login: Wed Nov  8 14:47:05 2017 from 10.3.1.51
cm@Ubuntu:~$ cd /etc/nagios3/libexec
cm@Ubuntu:/etc/nagios3/libexec$ sudo ./host_config.sh -i host
[sudo] password for cm:
Input the host address(IP):10.5.0.1
Input the userID:CM
Input the password:
Confirm the password:
Insert the item successfully
Host: 10.5.0.1    User: CM
cm@Ubuntu:/etc/nagios3/libexec$ █
```

les informations sont enregistrées cryptées dans `/user/local/nagios/Nagios.host.java.config.ser`
Vérifiez que `/usr/local/nagios` existe et que vous avez le droit d'écriture.

Plugin Nagios pour IBMi

- `as400.cfg` (*dans notre exemple*) : liste des machines IBM i

```
define host{
    use generic-host
    host_name AS400
    alias ibmi
    address 10.3.1.1
}
define hostgroup{
    hostgroup_name as400group1
    alias group1
    members AS400
}
```

Plugin Nagios pour IBMi

- services.cfg : liste des services à surveiller
(font références à des commandes)

```
define service{
  use generic-service, srv-pnp
  hostgroup_name as400group1
  service_description CPU utilization
  check_command check-ibmi-cpu-utilization!CPU!60!80
}
define service{
  use generic-service, srv-pnp
  hostgroup_name as400group1
  service_description Active job number
  check_command check-ibmi-active-job-num!ActiveJobs
}
define service{
```

Etc....

Plugin Nagios pour IBMi

- commands.cfg (liste des commandes, utilisées par les services)

```
#####  
#  
# IBM i STATUS CHECK COMMANDS  
#  
#####  
# 'check-cpu-utilization' command definition  
define command{  
    command_name check-ibmi-cpu-utilization  
    command_line /bin/bash $USER1$/check_ibmi_status.sh -m $ARG1$ -H $HOSTADDRESS$ -w $ARG2$ -c $ARG3$  
}  
# 'check-active-job' command definition  
define command{  
    command_name check-ibmi-active-job-num  
    command_line /bin/bash $USER1$/check_ibmi_status.sh -m $ARG1$ -H $HOSTADDRESS$  
}  
# 'check-disk-config' command definition  
define command{  
    command_name check-ibmi-disk-config  
    command_line /bin/bash $USER1$/check_ibmi_status.sh -m $ARG1$ -H $HOSTADDRESS$  
}  
# 'check-disk-usage' command definition  
define command{  
    command_name check-ibmi-disk-usage  
    command_line /bin/bash $USER1$/check_ibmi_status.sh -m $ARG1$ -H $HOSTADDRESS$ -w $ARG2$ -c $ARG3$  
}
```

Plugin Nagios pour IBMi

- les commandes font référence à \$USER1\$ qui est défini dans resource.cfg

```
# Sets $USER1$ to be the path to the plugins
$USER1$=/usr/lib/nagios/plugins
.../...
```

Nous avons donc copié *check_ibmi_status.sh* et *check_daemon_status.sh* dans */usr/lib/nagios/plugins* et nous les avons modifié pour qu'ils fassent référence à notre répertoire nagios (pour nous */etc/nagios3*)

```
#!/bin/bash
java -cp /etc/nagios3/libexec/jt400.jar:/etc/nagios3/libexec/server.jar:/etc/nagios3/libexec/ CheckIBMiStatus $*
```

Remarquez que cela va utiliser *jt400.jar*, pour en fait passer des requêtes **SQL**

Plugin Nagios pour IBMi

- il faut ensuite (re)démarrer nagios et le démon spécifique à IBM i
dans notre cas nous avons remplacé :
- les répertoires (à nouveau `/etc/nagios3`)
- et `service/nagios start | stop`
par `/etc/init.d/nagios3 start | stop`

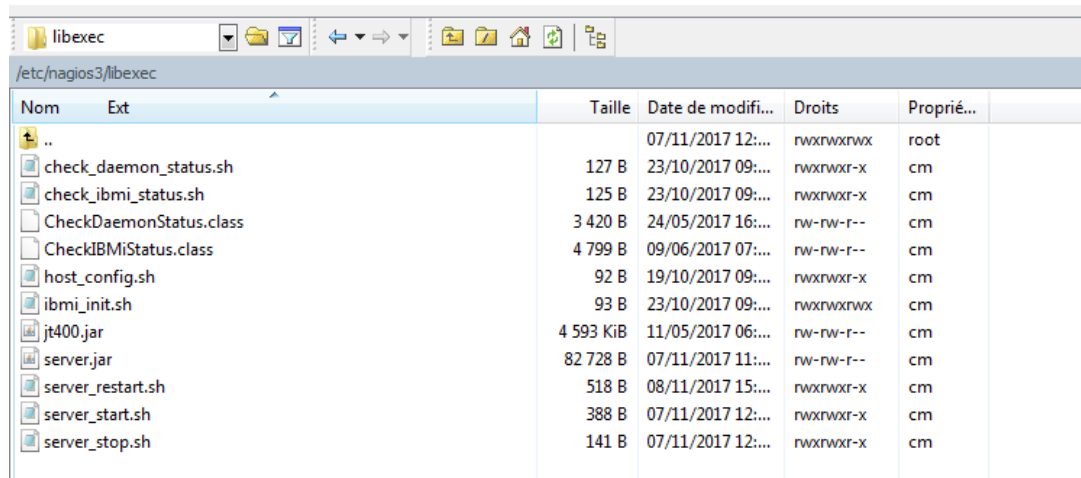
Puis Exécuter `server_start.sh` qui va lancer un démon **server.jar**.

C'est lui qui va se connecter à l'iBMi pour passer les requêtes
(en mode « SQL as a service »)

Plugin Nagios pour IBMi

- Lancement du démon

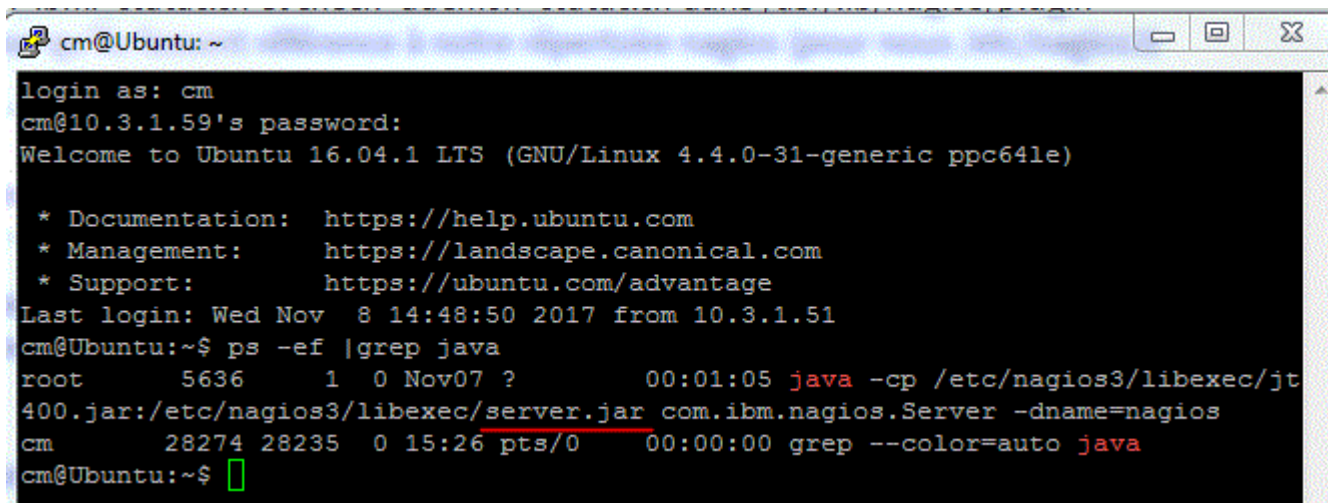
```
root@Ubuntu:/usr/lib/nagios3# cd /etc/nagios3/libexec
root@Ubuntu:/etc/nagios3/libexec# ./server_start.sh
Starting Nagios Server
nohup: redirecting stderr to stdout
Starting nagios3 (via systemctl): nagios3.service.
Nagios Service Started
root@Ubuntu:/etc/nagios3/libexec#
```



Nom	Ext	Taille	Date de modifi...	Droits	Proprié...
..			07/11/2017 12:...	rw-rwxrwx	root
check_daemon_status.sh	.sh	127 B	23/10/2017 09:...	rw-rwxr-x	cm
check_ibmi_status.sh	.sh	125 B	23/10/2017 09:...	rw-rwxr-x	cm
CheckDaemonStatus.class	.class	3 420 B	24/05/2017 16:...	rw-rw-r--	cm
CheckIBMiStatus.class	.class	4 799 B	09/06/2017 07:...	rw-rw-r--	cm
host_config.sh	.sh	92 B	19/10/2017 09:...	rw-rwxr-x	cm
ibmi_init.sh	.sh	93 B	23/10/2017 09:...	rw-rwxrwx	cm
jt400.jar	.jar	4 593 KiB	11/05/2017 06:...	rw-rw-r--	cm
server.jar	.jar	82 728 B	07/11/2017 11:...	rw-rw-r--	cm
server_restart.sh	.sh	518 B	08/11/2017 15:...	rw-rwxr-x	cm
server_start.sh	.sh	388 B	07/11/2017 12:...	rw-rwxr-x	cm
server_stop.sh	.sh	141 B	07/11/2017 12:...	rw-rwxr-x	cm

Plugin Nagios pour IBMi

- Vérifiez qu'il s'exécute par `ps -ef |grep java`

A screenshot of a terminal window on an Ubuntu system. The window title is 'cm@Ubuntu: ~'. The terminal shows a login sequence for user 'cm' on IP '10.3.1.59'. After the login, the user runs the command 'ps -ef |grep java'. The output shows two processes: one running as root (PID 5636) which is the Nagios server process, and another running as the user 'cm' (PID 28274) which is the 'grep' command itself. The Nagios process line is: 'root 5636 1 0 Nov07 ? 00:01:05 java -cp /etc/nagios3/libexec/jt400.jar:/etc/nagios3/libexec/server.jar com.ibm.nagios.Server -dname=nagios'. The 'grep' process line is: 'cm 28274 28235 0 15:26 pts/0 00:00:00 grep --color=auto java'. The terminal prompt is currently at 'cm@Ubuntu:~\$' with a cursor.

```
cm@Ubuntu: ~
login as: cm
cm@10.3.1.59's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic ppc64le)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
Last login: Wed Nov  8 14:48:50 2017 from 10.3.1.51
cm@Ubuntu:~$ ps -ef |grep java
root      5636      1  0 Nov07 ?        00:01:05 java -cp /etc/nagios3/libexec/jt
400.jar:/etc/nagios3/libexec/server.jar com.ibm.nagios.Server -dname=nagios
cm        28274    28235  0 15:26 pts/0    00:00:00 grep --color=auto java
cm@Ubuntu:~$
```

Plugin Nagios pour IBMi

- et voilà

Host Status Details For All Host Groups

Limit Results: 100

Host 	Status 	Last Check 	Duration 	Status Information
AS400 	UP	2017-11-08 15:30:24	2d 10h 3m 0s	PING OK - Packet loss = 0%, RTA = 0.40 ms
localhost 	UP	2017-11-08 15:27:44	40d 23h 51m 31s	PING OK - Packet loss = 0%, RTA = 0.08 ms

Results 1 - 2 of 2 Matching Hosts

Plugin Nagios pour IBMi

- détail

Service Status Details For Host 'AS400'

Limit Results:

Host	Service	Status	Last Check	Duration	Attempt	Status Information
AS400	Active job number	OK	2017-11-08 15:31:06	1d 3h 5m 12s	1/4	Active Jobs Num : 353
	Asp usage	OK	2017-11-08 15:26:31	1d 3h 4m 47s	1/4	ASP Usage: 46.22%
	Basic information	OK	2017-11-08 15:26:56	1d 3h 4m 22s	1/4	System Info: V7R3M0
	CPU overload jobs num	OK	2017-11-08 15:27:21	1d 3h 3m 57s	1/4	CPU overload job num: 0 (Warning: 60% Critical: 80%)
	CPU utilization	OK	2017-11-08 15:28:46	0d 15h 22m 32s	1/4	CPU Utilization: 0.58%
	Current log on user number	OK	2017-11-08 15:28:11	1d 3h 3m 7s	1/4	Currently Sign On Users Num: 0
	Disk configuration	OK	2017-11-08 15:28:36	1d 3h 2m 42s	1/4	Disk Status: OK
	Disk usage	OK	2017-11-08 15:29:01	1d 3h 2m 17s	1/4	Disk Usage status: OK
	Disk utilization	WARNING	2017-11-08 15:26:26	1d 2h 59m 52s	4/4	Disk Utilitize Status: WARN
	Long run sql	OK	2017-11-08 15:30:51	0d 21h 30m 27s	1/4	Long Run SQL Status: OK
	Message	WARNING	2017-11-08 15:27:16	1d 3h 4m 2s	4/4	Message Num in type 'INQUIRY','ESCAPE','REPLY': 1
	Page faults	OK	2017-11-08 15:30:41	0d 3h 5m 37s	1/4	Page Faults Status: OK
	Specific job CPU	OK	2017-11-08 15:26:19	1d 3h 4m 59s	1/4	Job: ADMIN2 CPU: 0.0%
	Specific message	OK	2017-11-08 15:28:44	1d 3h 2m 34s	1/4	Status: OK
	Subsystem jobs	OK	2017-11-08 15:27:09	1d 3h 4m 9s	1/4	42 jobs in subsystem QHTTPSVR
	Temporary storage top N jobs	OK	2017-11-08 15:27:34	1d 3h 3m 44s	1/4	10 jobs retrieved from endpoint

Results 1 - 16 of 16 Matching Services

Plugin Nagios pour IBMi

- Cde mail

```
10.3.1.29 - PuTTY
root@Ubuntu:~# mail
Mail version 8.1.2 01/15/2001. Type ? for help
"/var/mail/root": 1208 messages 1
>U 1 nagios@as400.volu Mon Oct
U 2 nagios@as400.volu Mon Oct
U 3 nagios@as400.volu Mon Oct
U 4 nagios@as400.volu Mon Oct
U 5 nagios@as400.volu Mon Oct
U 6 nagios@as400.volu Mon Oct
U 7 nagios@as400.volu Mon Oct
U 8 nagios@as400.volu Mon Oct
U 9 nagios@as400.volu Mon Oct
U 10 nagios@as400.volu Mon Oct
U 11 nagios@as400.volu Mon Oct
U 12 nagios@as400.volu Mon Oct
U 13 nagios@as400.volu Tue Oct
U 14 nagios@as400.volu Tue Oct
U 15 nagios@as400.volu Tue Oct
U 16 nagios@as400.volu Tue Oct
U 17 nagios@as400.volu Tue Oct
U 18 nagios@as400.volu Tue Oct

Message 66:
From nagios@as400.volubis.intra Sat Oct 28 00:09:37 2017
X-Original-To: root@localhost
To: root@localhost
Subject: ** PROBLEM Service Alert: ibmi/Page faults is WARNING **
Date: Sat, 28 Oct 2017 00:09:37 +0200 (CEST)
From: nagios@as400.volubis.intra

***** Nagios *****

Notification Type: PROBLEM

Service: Page faults
Host: ibmi
Address: 10.3.1.1
State: WARNING

Date/Time: Sat Oct 28 00:09:37 CEST 2017

Additional Info:

Page Faults Status: WARN
```

Plugin Nagios pour IBMi

- *Quelques exemples de requêtes capturées par un STRDBMON.*

Statement

```
SELECT AVERAGE_CPU_UTILIZATION FROM QSYS2.SYSTEM_STATUS_INFO
SELECT SYSTEM_ASP_USED, SYSTEM_ASP_STORAGE, CURRENT_TEMPORARY_STORAGE FROM QSYS2.SYSTEM_STATUS_INFO
SELECT ACTIVE_JOBS_IN_SYSTEM FROM QSYS2.SYSTEM_STATUS_INFO
SELECT MESSAGE_ID, MESSAGE_TEXT, MESSAGE_TIMESTAMP FROM QSYS2.MESSAGE_QUEUE_INFO WHERE MESSAGE_ID LIKE ?
WITH ACTIVE_USER_JOBS (Q_JOB_NAME, CPU_TIME, RUN_PRIORITY) AS (SELECT JOB_NAME, CPU_TIME, RUN_PRIORITY FROM TABLE (QSYS2.ACTI...
SELECT MESSAGE_TYPE, MESSAGE_TEXT, SEVERITY, MESSAGE_TIMESTAMP FROM QSYS2.MESSAGE_QUEUE_INFO WHERE MESSAGE_QUEUE_LIBRARY ...
SELECT ELAPSED_CPU_PERCENTAGE FROM TABLE(QSYS2.ACTIVE_JOB_INFO(? , ? , ? , ?)) X
SELECT SUBSTR(JOB_NAME,8,POSSTR(SUBSTR(JOB_NAME,8),'/')-1) AS JOB_USER, SUBSTR(SUBSTR(JOB_NAME,8),POSSTR(SUBSTR(JOB_NAME,8),'/')...
SELECT SUBSTR(JOB_NAME,8,POSSTR(SUBSTR(JOB_NAME,8),'/')-1) AS JOB_USER, SUBSTR(SUBSTR(JOB_NAME,8),POSSTR(SUBSTR(JOB_NAME,8),'/')...
SELECT SUBSTR(JOB_NAME,8,POSSTR(SUBSTR(JOB_NAME,8),'/')-1) AS JOB_USER, SUBSTR(SUBSTR(JOB_NAME,8),POSSTR(SUBSTR(JOB_NAME,8),'/')...
SELECT ASP_NUMBER, UNIT_NUMBER, UNIT_TYPE, UNIT_STORAGE_CAPACITY, UNIT_SPACE_AVAILABLE, PERCENT_USED FROM QSYS2.SYSDISKSTAT
SELECT POOL_NAME, ELAPSED_TOTAL_FAULTS FROM QSYS2.POOL_INFO
```




La solution basée sur l'utilisation d'un agent

Comment aller plus loin ?

- Pourquoi ne pas suivre réellement TOUT ce qui se passe dans les partitions IBM i (OS/400) ?
 - Système
 - Y a-t-il un incident matériel ?
 - Puis-je remonter des alertes de performances ?
 - Exploitation
 - Y a t'il un travail planté ou en attente de réponse ?
 - Les sauvegardes ont-elles été réalisées correctement ?
 - Les travaux de nuit ont-ils tournés et se sont-ils bien terminés ?
 - Haute disponibilité
 - La réplication fonctionne t'elle correctement ?
 - N'y a t'il pas de retard ou d'objets bloqués ?
 - Base de données
 - Puis je aller chercher une info essentielle dans la base de données ?
 - Applications
 - L'ERP est-il bien démarré avec tous ses composants ?
 - L'EDI a bien transféré les commandes à telle heure ?
 - etc, etc, ...

Surveiller tout l'IBM i en temps réel

Le produit «  CONTROL »
for 

permet de surveiller une partition IBM i

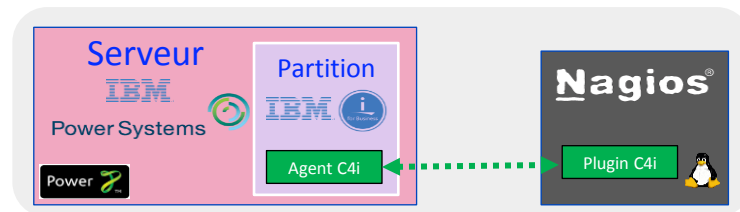
en utilisant des commandes développées sur IBM i

- Commandes naturelles pour les spécialistes IBM i
- Mise au point facile sur IBM i avant de l'intégrer à Nagios
- Possibilité d'adapter des contrôles déjà existants
- Paramétrage ultra simple dans Nagios

Surveiller tout l'IBM i en temps réel

- Le produit  **CONTROL** est basé sur l'utilisation d'un agent 

- Un agent doit être présent et démarré sur chaque partition IBM i
 - Constitué d'un sous-système et 2 travaux
 - Déploiement et installation très simple
 - L'agent attend les commandes venant du Plugin
- Un plugin unique sur le serveur Nagios
 - Très simple d'utilisation
 - En paramètre : la commande IBM i à appeler
 - `check_Control4i -H 10.55.1.3 -c « CTCHKSBS SBSD(QBATCH) »`
 - Sécurité gérée par le produit
 - Pas de profil / mot de passe



Surveiller tout l'IBM i en temps réel



■ Exemple de commande

```
Vérif. état de sous-système (CTCHKSBS)

Indiquez vos choix, puis appuyez sur ENTREE.

Nom sous-système à contrôler . . . > QBATCH          Nom
Bibliothèque . . . . .                *ANY          Nom, *ANY
Etat sous-système à contrôler . . . . *RUN          *RUN, *STOP
Nbr mini travaux actifs . . . . .     *NOCTL       Nombre, *NOCTL
Liste des travaux obligatoires > JOB01          Nom, *NONE
      + si autres valeurs > AUTOMATE
Niveau de criticité . . . . .         *CRITICAL    *CRITICAL, *WARNING

F3=Exit  F4=Invite  F5=Réafficher  F12=Annuler  F13=Mode d'emploi invite
F24=Autres touches
```

```
CTCHKSBS SBSD(QBATCH) LSTJOB(JOB01 AUTOMATE)
Sous-système QBATCH, Il manque 2 travail(travaux). Les 10 premiers sont
JOB01, AUTOMATE, , , , , , , , , .
```

Surveiller tout l'IBM i en temps réel

Exemple : appel de la commande CTCHKSBS

Home | Monitoring | Views | Reporting | Configuration | Administration

Hosts | Services | Users | Commands | Notifications | SNMP Traps | Monitoring Engines | Centreon

>> Configuration > Services > Services by host

Service Configuration | Relations | Data Processing | Service Extended Info

Add a Service

General Information

Description * QBATCH et 2 travaux

Service Template generic-service

Service State

Is Volatile Yes No Default

Check Period * 24x7

Check Command * check_Control4i

Args

Argument	Value	Example
Commande IBM i	CTCHKSBS SBSD(QBAT	

Max Check Attempts *

Normal Check Interval * * 60 seconds

Retry Check Interval * * 60 seconds

Active Checks Enabled Yes No Default

Passive Checks Enabled Yes No Default

Macros

Custom macros	Macro name	Macro value	
			+

Notification

Les contrôles fournis en standard



- Surveiller les sous-systèmes et travaux
 - CTCHKSBS
 - CTCHKJOB
 - CTCHKBCH
- Surveiller qu'aucun travail n'est bloqué
 - CTCHKMSGW
 - CTCHKLCKW
- Surveiller les files d'attentes
 - CTCHKJOBQ, CTCHKOUTQ, CTCHKDTAQ
- Surveiller l'occupation disque, ou l'utilisation CPU d'un travail/sous-système
 - CTCHKDSK
 - CTCHKCPU *SYSTEM/*SBS/*JOB
- Vérifier qu'il n'y a pas de problème hardware
 - CTCHKPRB
- Vérifier que les sauvegardes de la nuit étaient correctes
 - CTCHKSAV
 - CTCHKBRM, CTCHKBRMEX
- Vérifier que la réplication n'est pas en retard
 - CTCHKEDH
 - CTCHKMMXAG, CTCHKMMXDG, CTCHKMMXDS
- Faire un Ping à partir de la partition IBM i
 - CTCHKPING
- Vérifier l'état des profils utilisateurs
 - CTCHKUSR
- Vérifier l'ERP (exemple M3)
 - CTM3HSTSTS, Contrôle de l'état des hôtes
 - CTM3NODSTS, Contrôle de l'état des nodes
 - CTM3APPSTS, Contrôle de l'état des applications
 - CTM3NBAJ, Contrôle du nombre de travaux Asynchrones ou AutoJobs
 - CTM3JOBSTS, Contrôles sur les jobs batchs
- Et bien d'autres à venir ...

Possibilité de réaliser des contrôles spécifiques

En effet, tout programme existant qui répond :

- Oui / Non
- 0 / 1
- ça marche / Il y a un problème

peut être transformé en sonde Nagios en moins de 30 minutes

- Des 'templates' sont fournis afin que vous développiez vos nouvelles sondes

```
PDM - Gestion des membres M81DEV
Fichier . . . . . CTEXAMPLE
Bibliothèque : CTL4I Afficher à partir de . . . . .

Opt  Membre  Type  Texte
_    EXAMPLE_01  CMD   Ctl4i, Example command for specific check
_    EXAMPLE_02  CLLE  Ctl4i, Example program for specific check
```

- il est donc possible de réutiliser des contrôles déjà développés

Merci de votre attention

**N'oubliez pas de remplir
le questionnaire
de satisfaction !**