Dossiers importants

Mémento rapide
NAGIOS/CENTREON

La définition des machines, des groupes, des services ... à superviser est faite grâce à des fichiers ***.cfg*** stockés dans le dossier :

/usr/local/nagios/etc/objects/ ***(si Nagios est installé seul)***
/etc/nagios ***(si Nagios est installé avec FAN)***

Les icônes utilisables sont ceux disponibles dans le dossier *logos :*

/usr/local/nagios/share/images/logos/ ***(si Nagios est installé seul)***
/usr/share/nagios/images/logos/ ***(si Nagios est installé avec FAN)***

Les plugins Nagios sont stockés dans le dossier

/usr/local/nagios/libexec/

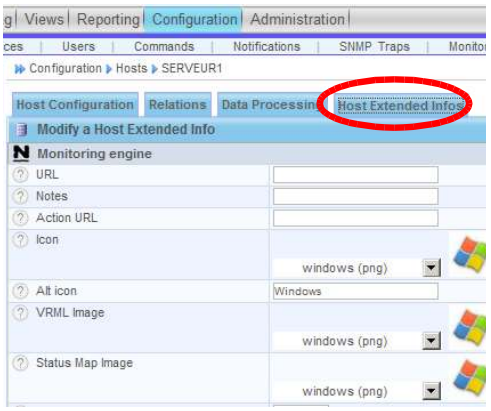
Beaucoup de commandes sont déjà définies dans :

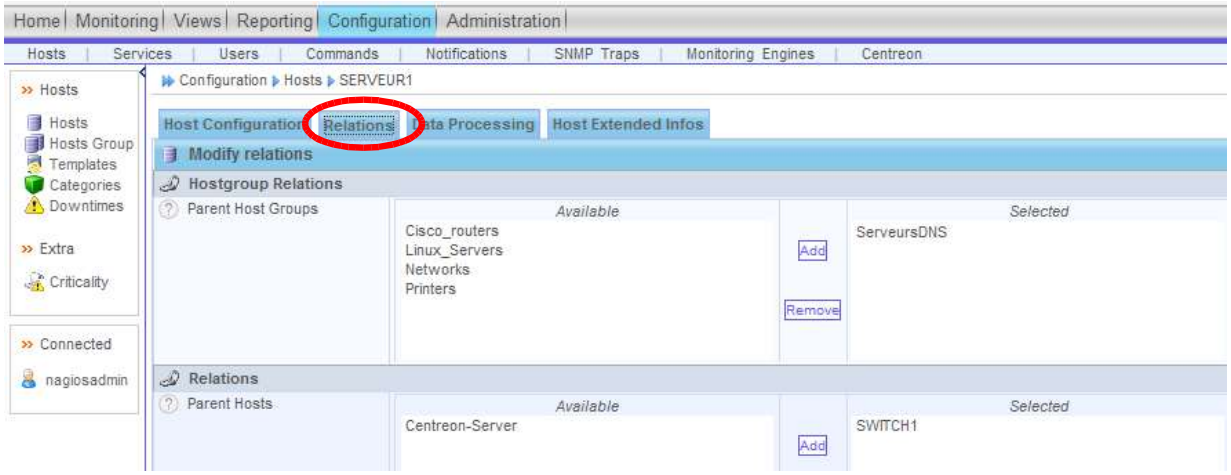
le fichier ***/usr/local/nagios/etc/objects/commands.cfg*** ***(si Nagios est installé seul)***
le dossier ***/etc/nagios*** ***(si Nagios est installé avec FAN)***

Définition d'un hôte

Lien de référence :
https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/objectdefinitions.html

```
define host {
        use             generic-host
        host_name       SERVEUR1
        alias           Serveur de domaine Windows
        address         192.168.1.21
        parents         SWITCH1
        icon_image      win40.png
        icon_image_alt  Windows
        vrml_image      win40.png
}
```

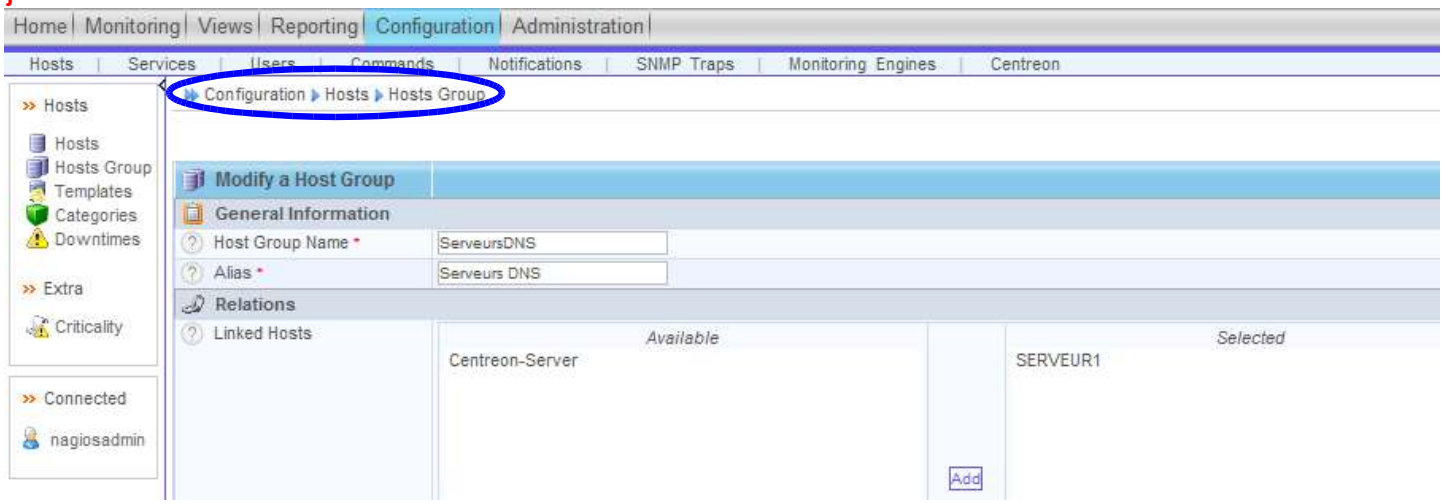




L'hôte *localhost* est déjà défini dans */usr/local/nagios/etc/objects/localhost.cfg*

Définition d'un groupe d'hôtes

define hostgroup {
hostgroup_name *ServeursDNS*
alias *Serveurs DNS*
members *SERVEUR1*
}

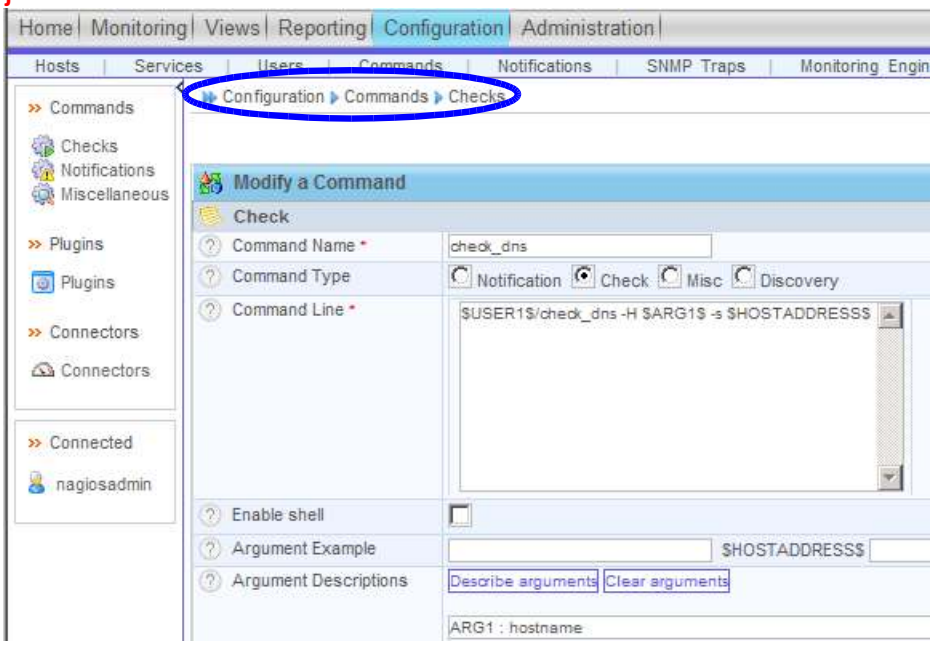


Plusieurs groupes d'hôtes (*linux-servers*, *windows-servers*) sont déjà définis respectivement dans */usr/local/nagios/etc/objects/localhost.cfg* et dans */usr/local/nagios/etc/objects/windows.cfg*.

localhost est membre de *linux-servers*.

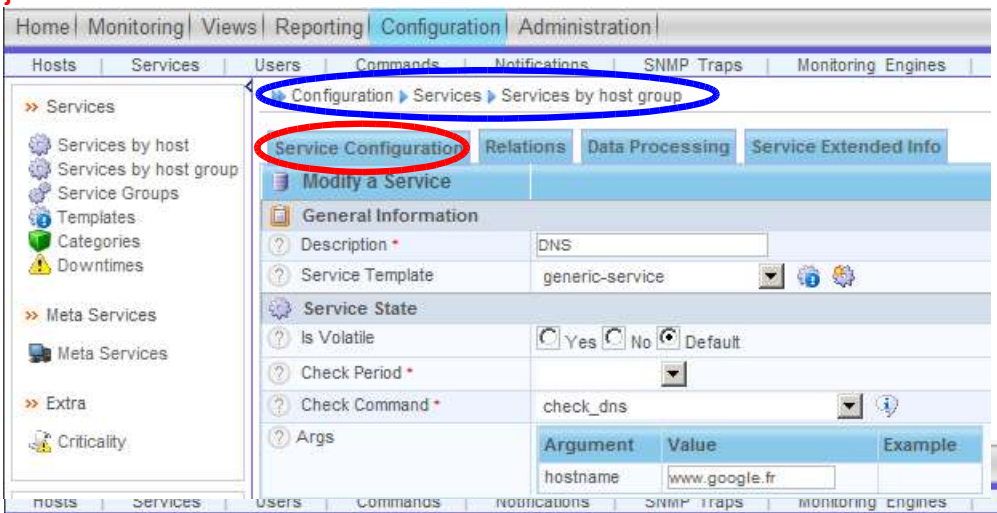
Définition d'une commande

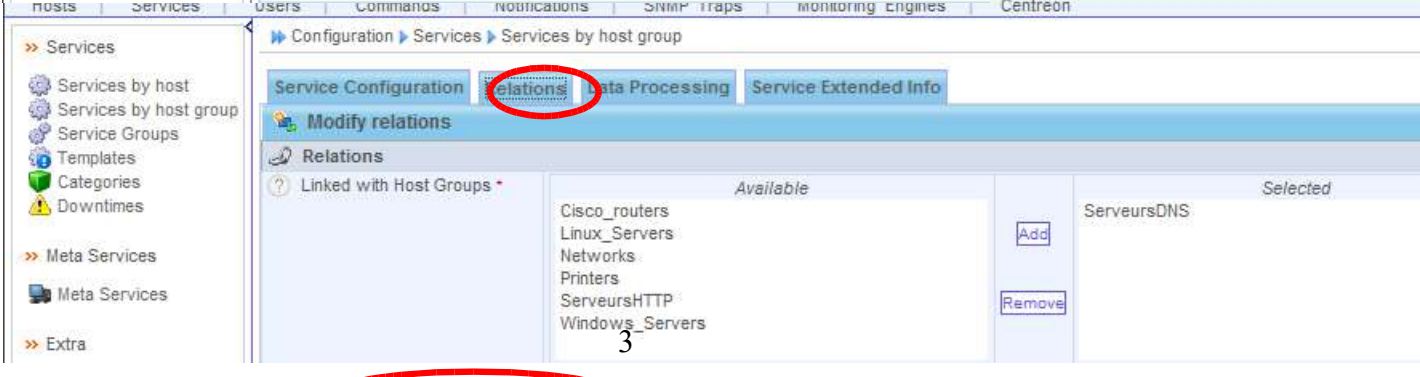
```
define command {
        command_name        check_dns
        command_line        $USER1$/check_dns -H $ARG1$ -s $HOSTADDRESS$
}
```



Définition d'un service

```
define service {
        use                 generic-service
        hostgroup_name      ServeursDNS
        service_description DNS
        check_command       check_dns!www.google.fr
}
```





Notice d'utilisation du plugin check_http

./check_http -h
check_http v1.4.16 (nagios-plugins 1.4.16)
Copyright (c) 1999 Ethan Galstad <nagios@nagios.org>
Copyright (c) 1999-2011 Nagios Plugin Development Team
<nagiosplug-devel@lists.sourceforge.net>

Ce plugin teste le service HTTP sur l'hôte spécifié. Il peut tester les
serveurs normaux (http) et sécurisés (https), suivre les redirections, rechercher des
chaînes de caractères et expressions rationnelles, vérifier le temps de réponse
et rapporter la date d'expiration du certificat.

Utilisation:
check_http -H <vhost> | -I <IP-address> [-u <uri>] [-p <port>]
[-w <warn time>] [-c <critical time>] [-t <timeout>] [-L] [-a auth]
[-b proxy_auth] [-f <ok|warning|critcal|follow|sticky|stickyport>]
[-e <expect>] [-s string] [-l] [-r <regex> | -R <case-insensitive regex>]
[-P string] [-m <min_pg_size>:<max_pg_size>] [-4|-6] [-N] [-M <age>]
[-A string] [-k string] [-S <version>] [--sni] [-C <warn_age>[,<crit_age>]]
[-T <content-type>] [-j method]
NOTE: les paramètres -H et -I peuvent être spécifiés

Options:
-h, --help
Afficher l'aide détaillée
-V, --version
Afficher les informations relative à la version
-H, --hostname=ADDRESS
IP address or name of HTTP server (use numeric address if possible to bypass DNS lookup)
Append a port to include it in the header (eg: example.com:5000)
-I, --IP-address=ADDRESS
IP address or name (use numeric address if possible to bypass DNS lookup).
-p, --port=INTEGER
Numéro du port (défaut: 80)
-4, --use-ipv4
Utiliser une connection IPv4
-6, --use-ipv6
Utiliser une connection IPv6
-S, --ssl=VERSION
Connect via SSL. Port defaults to 443. VERSION is optional, and prevents
auto-negotiation (1 = TLSv1, 2 = SSLv2, 3 = SSLv3).
--sni
Enable SSL/TLS hostname extension support (SNI)
-C, --certificate=INTEGER
Nombre de jours minimum pour que le certificat soit valide. Port par défaut 443
(when this option is used the URL is not checked.)

-e, --expect=STRING
Liste the chaines de charactères séparées par des virgules, au moins une d'elles
est attendue dans la première ligne de réponse du serveur (défaut: HTTP/1.)
Si spécifié, surpasse toute autre logique de status (ex: 3xx, 4xx, 5xx)
-s, --string=STRING
Chaîne de caractère attendue dans le contenu
-u, --url=PATH
URL de la ressource Internet à superviser (défaut: /)
-P, --post=STRING

URL encoded http POST data
-j, --method=STRING (for example: HEAD, OPTIONS, TRACE, PUT, DELETE)
Set HTTP method.
-N, --no-body
Ne pas attendre pour le corps du document: arrêter de lire après les entêtes
(Veuillez noter qu'un HTTP GET ou POST est effectué, pas un HEAD.)
-M, --max-age=SECONDS
Warn if document is more than SECONDS old. the number can also be of
the form "10m" for minutes, "10h" for hours, or "10d" for days.
-T, --content-type=STRING
specify Content-Type header media type when POSTing

-l, --linespan
Allow regex to span newlines (must precede -r or -R)
-r, --regex, --ereg=STRING
Search page for regex STRING
-R, --eregi=STRING
Search page for case-insensitive regex STRING
--invert-regex
Return CRITICAL if found, OK if not

-a, --authorization=AUTH_PAIR
Username:password on sites with basic authentication
-b, --proxy-authorization=AUTH_PAIR
Username:password on proxy-servers with basic authentication
-A, --useragent=STRING
String to be sent in http header as "User Agent"
-k, --header=STRING
Any other tags to be sent in http header. Use multiple times for additional headers
-L, --link
Wrap output in HTML link (obsoleted by urlize)
-f, --onredirect=<ok|warning|critical|follow|sticky|stickyport>
How to handle redirected pages. sticky is like follow but stick to the
specified IP address. stickyport also ensures port stays the same.
-m, --pagesize=INTEGER<:INTEGER>
Minimum page size required (bytes) : Maximum page size required (bytes)
-w, --warning=DOUBLE
Temps de réponse résultant en un état d'avertissement (secondes)
-c, --critical=DOUBLE
Temps de réponse résultant en un état critique (secondes)
-t, --timeout=INTEGER
Délais de connection en secondes (défaut: 10)
-v, --verbose
Affiche les informations de déboguage en ligne de commande (Nagios peut tronquer la sortie)

Exemple : check_http -H 192.168.211.181

Notice d'utilisation du plugin check_dns

./check_dns -h
check_dns v1.4.16 (nagios-plugins 1.4.16)
Copyright (c) 1999 Ethan Galstad <nagios@nagios.org>
Copyright (c) 2000-2008 Nagios Plugin Development Team
<nagiosplug-devel@lists.sourceforge.net>

Ce plugin utilise le programme nslookup pour obtenir l'adresse IP de l'hôte/domaine à interroger.
Un serveur DNS à utiliser peut être indiqué.
Si aucun serveur DNS n'est spécifié, les serveurs spécifiés dans /etc/resolv.conf seront utilisé.

Utilisation:
check_dns -H host [-s server] [-a expected-address] [-A] [-t timeout] [-w warn] [-c crit]

Options:
-h, --help
Afficher l'aide détaillée
-V, --version
Afficher les informations relative à la version
-H, --hostname=HOST
Hostname you want to resolve (translate) to IP address
-s, --server=HOST
IP address of DNS Server you want to use for the lookup
-a, --expected-address=IP-ADDRESS|HOST
Adresse IP que le serveur DNS doit retourner. Les hôtes doivent se terminer avec un point (.). Cette option peut être répétée (Retourne OK si une valeur correspond). Si plusieurs adresses sont retournées en même temps, vous devrez toutes les inscrire séparées pas des virgules (en ordre alphabétique)
-A, --expect-authority
Serveur DNS qui doit normalement être autoritaire pour la recherche
-w, --warning=seconds
Renvoie une alerte si le temps écoulé dépasse la valeur indiquée. Désactivé par défaut
-c, --critical=seconds
Renvoie critique si le temps utilisé dépasse la valeur indiquée. Désactivé par défaut
-t, --timeout=INTEGER
Délais de connection en secondes (défaut: 10)

Exemple : check_dns -H www.google.fr -s 192.168.211.181

Notice d'utilisation du plugin check_dhcp

./check_dhcp -h
check_dhcp v1.4.16 (nagios-plugins 1.4.16)
Copyright (c) 2001-2004 Ethan Galstad (nagios@nagios.org)
Copyright (c) 2001-2007 Nagios Plugin Development Team
<nagiosplug-devel@lists.sourceforge.net>

Ce plugin teste la disponibilité de serveurs DHCP dans un réseau.

Utilisation:
check_dhcp [-v] [-u] [-s serverip] [-r requestedip] [-t timeout]
[-i interface] [-m mac]

Options:
-h, --help
Afficher l'aide détaillée
-V, --version
Afficher les informations relative à la version
-v, --verbose
Affiche les informations de déboguage en ligne de commande (Nagios peut tronquer la sortie)
-s, --serverip=IPADDRESS
IP address of DHCP server that we must hear from
-r, --requestedip=IPADDRESS
IP address that should be offered by at least one DHCP server
-t, --timeout=INTEGER
Seconds to wait for DHCPOFFER before timeout occurs
-i, --interface=STRING
Interface to to use for listening (i.e. eth0)
-m, --mac=STRING
MAC address to use in the DHCP request
-u, --unicast
Unicast testing: mimic a DHCP relay, requires -s

Exemple : check_dhcp -s 192.168.211.181

Notice d'utilisation du plugin check_ftp

./check_ftp -h

```
check_ftp v2.2.git (monitoring-plugins 2.2)
Copyright (c) 1999-2008 Monitoring Plugins Development Team

This plugin tests FTP connections with the specified host (or unix socket).

Usage:
check_ftp -H host -p port [-w <warning time>] [-c <critical time>] [-s <send string>]
[-e <expect string>] [-q <quit string>][-m <maximum bytes>] [-d <delay>]
[-t <timeout seconds>] [-r <refuse state>] [-M <mismatch state>] [-v] [-4|-6] [-j]
[-D <warn days cert expire>[,<crit days cert expire>]] [-S <use SSL>] [-E]

Options:
 -h, --help                    Print detailed help screen
 -V, --version                 Print version information
 --extra-opts=[section][@file]
   Read options from an ini file. See  https://www.monitoring-plugins.org/doc/extra-opts.html for usage and examples.
-H, --hostname=ADDRESS
   Host name, IP Address, or unix socket (must be an absolute path)
 -p, --port=INTEGER
   Port number (default: none)
 -4, --use-ipv4
   Use IPv4 connection
 -6, --use-ipv6
   Use IPv6 connection
 -E, --escape
   Can use \n, \r, \t or \\ in send or quit string. Must come before send or quit option ;
   Default: nothing added to send, \r\n added to end of quit
 -s, --send=STRING
   String to send to the server
 -e, --expect=STRING
   String to expect in server response (may be repeated)
 -A, --all
   All expect strings need to occur in server response. Default is any
 -q, --quit=STRING
   String to send server to initiate a clean close of the connection
 -r, --refuse=ok|warn|crit
   Accept TCP refusals with states ok, warn, crit (default: crit)
 -M, --mismatch=ok|warn|crit
   Accept expected string mismatches with states ok, warn, crit (default: warn)
 -j, --jail
   Hide output from TCP socket
 -m, --maxbytes=INTEGER
   Close connection once more than this number of bytes are received
 -d, --delay=INTEGER
   Seconds to wait between sending string and polling for response
 -D, --certificate=INTEGER[,INTEGER]
   Minimum number of days a certificate has to be valid.     1st is #days for warning, 2nd is critical (if not specified - 0).
 -S, --ssl
   Use SSL for the connection.
 -w, --warning=DOUBLE
   Response time to result in warning status (seconds)
 -c, --critical=DOUBLE
   Response time to result in critical status (seconds)
 -t, --timeout=INTEGER
   Seconds before connection times out (default: 10)
 -v, --verbose             Show details for command-line debugging (output may be truncated by  the monitoring system)
```

Notice d'utilisation du plugin check_snmp

./check_snmp -h
check_snmp v1.4.16 (nagios-plugins 1.4.16)
Copyright (c) 1999-2007 Nagios Plugin Development Team
<nagiosplug-devel@lists.sourceforge.net>

Vérifie l'état des machines distantes et obtient l'information système via SNMP

Utilisation:
check_snmp -H <ip_address> -o <OID> [-w warn_range] [-c crit_range]
[-C community] [-s string] [-r regex] [-R regexi] [-t timeout] [-e retries]
[-l label] [-u units] [-p port-number] [-d delimiter] [-D output-delimiter]
[-m miblist] [-P snmp version] [-L seclevel] [-U secname] [-a authproto]
[-A authpasswd] [-x privproto] [-X privpasswd]

Options:
-h, --help
Afficher l'aide détaillée
-V, --version
Afficher les informations relative à la version
-H, --hostname=ADDRESS
Nom d'hôte, Adresse IP, ou socket UNIX (doit être un chemin absolu)
-p, --port=INTEGER
Numéro de port (défaut: 161)
-n, --next
Utiliser SNMP GETNEXT au lieu de SNMP GET
-P, --protocol=[1|2c|3]
Version du protocole SNMP
-L, --seclevel=[noAuthNoPriv|authNoPriv|authPriv]
Niveau de sécurité SNMPv3 (securityLevel)
-a, --authproto=[MD5|SHA]
Protocole d'authentification SNMPv3
-x, --privproto=[DES|AES]
SNMPv3 priv proto (default DES)
-C, --community=STRING
Communauté optionnelle pour la communication SNMP (défaut: "public")
-U, --secname=USERNAME
Nom d'utilisateur SNMPv3
-A, --authpassword=PASSWORD
Mot de passe d'authentification SNMPv3
-X, --privpasswd=PASSWORD
Mot de passe de confidentialité SNMPv3
-o, --oid=OID(s)
Object identifier(s) or SNMP variables whose value you wish to query
-m, --miblist=STRING
List of MIBS to be loaded (default = none if using numeric OIDs or 'ALL'
for symbolic OIDs.)
-d, --delimiter=STRING
Delimiter to use when parsing returned data. Default is "="
Any data on the right hand side of the delimiter is considered
to be the data that should be used in the evaluation.
-w, --warning=THRESHOLD(s)
Valeurs pour le seuil d'avertissement
-c, --critical=THRESHOLD(s)
Valeurs pour le seuil critique
--rate
Enable rate calculation. See 'Rate Calculation' below

--rate-multiplier
  Converts rate per second. For example, set to 60 to convert to per minute
-s, --string=STRING
  Return OK state (for that OID) if STRING is an exact match
-r, --ereg=REGEX
  Return OK state (for that OID) if extended regular expression REGEX matches
-R, --eregi=REGEX
  Return OK state (for that OID) if case-insensitive extended REGEX matches
--invert-search
  Invert search result (CRITICAL if found)
-l, --label=STRING
  Prefix label for output from plugin
-u, --units=STRING
  Units label(s) for output data (e.g., 'sec.').
-D, --output-delimiter=STRING
  Separates output on multiple OID requests
-t, --timeout=INTEGER
  Délais de connection en secondes (défaut: 10)
-e, --retries=INTEGER
  Le nombre d'essai pour les requêtes
-O, --perf-oids
  Label performance data with OIDs instead of --label's
-v, --verbose
  Affiche les informations de déboguage en ligne de commande (Nagios peut tronquer la sortie)

This plugin uses the 'snmpget' command included with the NET-SNMP package.
Si vous n'avez pas le programme installé, vous devrez le télécharger depuis
http://net-snmp.sourceforge.net avant de pouvoir utiliser ce plugin.

Notes:
- Multiple OIDs (and labels) may be indicated by a comma or space-delimited
  list (lists with internal spaces must be quoted). Maximum: 8 OIDs.
- Voir:
http://nagiosplug.sourceforge.net/developer-guidelines.html#THRESHOLDFORMAT
pour le format et examples des seuils (THRESHOLD).
- When checking multiple OIDs, separate ranges by commas like '-w 1:10,1:,:20'
- Note that only one string and one regex may be checked at present
- All evaluation methods other than PR, STR, and SUBSTR expect that the value
  returned from the SNMP query is an unsigned integer.

Rate Calculation:
In many places, SNMP returns counters that are only meaningful when
calculating the counter difference since the last check. check_snmp
saves the last state information in a file so that the rate per second
can be calculated. Use the --rate option to save state information.
On the first run, there will be no prior state - this will return with OK.
The state is uniquely determined by the arguments to the plugin, so
changing the arguments will create a new state file.

Envoyez un email à nagios-users@lists.sourceforge.net si vous avez des questions
reliées à l'utilisation de ce logiciel. Pour envoyer des patches ou suggérer des
améliorations, envoyez un email à nagiosplug-devel@lists.sourceforge.net

Exemple : check_snmp -H 127.0.0.1 -C public -o .1.3.6.1.2.1.1.1.0

check_snmp_storage.pl
check_manubulon_snmp_storage.pl

Description :

This scripts checks by snmp (V1 and v3) disks, memory, swap, everthing in hrStorage table.
Storages selection can be done :
- by perl regexp on description or index (*-m*)
- and (optional) by storage type (*-q*) : Other, Ram, VirtualMemory, FixedDisk, RemovableDisk, FloppyDisk, CompactDisk, RamDisk, FlashMemory, NetworkDisk

One or multiple storages can be selected.
It is also possible to sum all selected storages (*-s*)

Warning and critical levels can be checked based on :
- Percent of disk used
- Percent of disk left
- MB left
- MB used

Output of check_snmp_storage.pl -h

```
SNMP Disk Monitor for Nagios version 1.3.3
(c)2004-2007 Patrick Proy

Usage: check_snmp_storage [-v] -H <host> -C <snmp_community> [-2] | (-l login -x passwd [-X pass -L <authp>,<privp>]) [-p
<port>] -m <name in desc_oid> [-q storagetype] -w <warn_level> -c <crit_level> [-t <timeout>] [-T pl|pu|bl|bu ] [-r -s -i -G] [-
e] [-S 0|1[,1,<car>]] [-o <octet_length>] [-R <% reserved>]
By default, plugin will monitor %used on drives :
warn if %used > warn and critical if %used > crit
-v, --verbose
print extra debugging information (and lists all storages)
-h, --help
print this help message
-H, --hostname=HOST
name or IP address of host to check
-C, --community=COMMUNITY NAME
community name for the host's SNMP agent (implies SNMP v1)
-2, --v2c
Use snmp v2c
-l, --login=LOGIN ; -x, --passwd=PASSWD
Login and auth password for snmpv3 authentication
If no priv password exists, implies AuthNoPriv
-X, --privpass=PASSWD
Priv password for snmpv3 (AuthPriv protocol)
-L, --protocols=<authproto>,<privproto>
<authproto> : Authentication protocol (md5|sha : default md5)
<privproto> : Priv protocole (des|aes : default des)
-x, --passwd=PASSWD
Password for snmpv3 authentication
-p, --port=PORT
SNMP port (Default 161)
-m, --name=NAME
Name in description OID (can be mounpoints '/home' or 'Swap Space'...)
This is treated as a regexp : -m /var will match /var , /var/log, /opt/var ...
Test it before, because there are known bugs (ex : trailling /)
No trailing slash for mountpoints !
If multiple storage are selected, the worse condition will be returned
i.e if one disk is critical, the return is critical
example :
Test drive C,F,G,H,I on Windows : -m ^[CFGHI]:
Test all mounts containing /var : -m /var
Test all mounts under /var : -m ^/var
Test only /var : -m /var -r
Test all swap spaces : -m ^Swap
Test all but swap spaces : -m ^Swap -e-
q, --storagetype=[Other|Ram|VirtualMemory|FixedDisk|RemovableDisk|FloppyDisk
CompactDisk|RamDisk|FlashMemory|NetworkDisk]
Also check the storage type in addition of the name
It is possible to use regular expressions ( "FixedDisk|FloppyDisk" )
-r, --noregexp
Do not use regexp to match NAME in description OID
```

-s, --sum
Add all storages that match NAME (used space and total space)
THEN make the tests.
-i, --index
Parse index table instead of description table to select storage
-e, --exclude
Select all storages except the one(s) selected by -m
No action on storage type selection
-T, --type=TYPE
pl : calculate percent left
pu : calculate percent used (Default)
bl : calculate MegaBytes left
bu : calculate MegaBytes used
-w, --warn=INTEGER
percent / MB of disk used to generate WARNING state
you can add the % sign
-c, --critical=INTEGER
percent / MB of disk used to generate CRITICAL state
you can add the % sign
-R, --reserved=INTEGER
% reserved blocks for superuser
For ext2/3 filesystems, it is 5% by default
-G, --gigabyte
output, warning & critical levels in gigabytes
-f, --perfparse
Perfparse compatible output
-S, --short=<type>[,<where>,<cut>]
<type>: Make the output shorter :
0 : only print the global result except the disk in warning or critical
ex: "< 80% : OK"
1 : Don't print all info for every disk
ex : "/ : 66 %used (< 80) : OK"
<where>: (optional) if = 1, put the OK/WARN/CRIT at the beginning
<cut>: take the <n> first caracters or <n> last if n<0
-o, --octetlength=INTEGER
max-size of the SNMP message, usefull in case of Too Long responses.
Be carefull with network filters. Range 484 - 65535, default are
usually 1472,1452,1460 or 1440.
-t, --timeout=INTEGER
timeout for SNMP in seconds (Default: 5)
-V, --version
prints version number
Note :
with T=pu or T=bu : OK < warn < crit
with T=pl ot T=bl : crit < warn < OK

example :
Browse storage list : <script> -C <community> -H <host> -m <anything> -w 1 -c 2 -v

Output options (*-S*):

-S can have 3 options : <type>[,<where>,<cut>]
<*type*> : Make the output shorter :
0 : only print the global result except the disk in warning or critical
1 : Don't print all info for every disk
<*where*> : (optional) if = 1, put the OK/WARN/CRIT at the beginning
<*cut*> : take the <n> first caracters or <n> last if n<0

With the following disks : /home : 51% used and / : 90% used

Option used	Warning	Output
	99%	/home: 51%used(1012MB/1969MB) /: 90%used(5781MB/6390MB) (<99%) : OK
	90%	/home: 51%used(1012MB/1969MB) /: 90%used(5781MB/6390MB) (>90%) : WARNING
-S0	90%	All selected storages (<99%) : OK
-S0	99%	/: 90%used(5781MB/6390MB) (>90%) : WARNING
-S1	90%	/home: 51% /: 90% (<99%) : OK
-S1	99%	/home: 51% /: 90%**used(5781MB/6390MB)** (>90%) : WARNING
-S0,1	90%	**OK** : (<99%) All selected storages
-S0,1	90%	**WARNING** : (>90%) /: 90%used(5781MB/6390MB)
-S1,,**2**	99%	**/h**: 51% /: 90% (<99%) : OK
-S1,,**-2**	99%	**me**: 51% /: 90% (<99%) : OK
-S0,1,-2	30%	WARNING : (>30%) me: 51%used(1012MB/1969MB) /: 90%used(5781MB/6390MB)

Performance output (*-f* option)

The performance will remove any weird caracters (**`~!$%^&*'"<>|?,(=)**) from the drive name.

Msg size option (*-o* option)

In case you get a "ERROR: running table : Message size exceeded maxMsgSize" error, you may need to adjust the maxMsgSize, i.e. the maximum size of snmp message with the -o option. Try a value with the -o AND the -v option : the script will output the actual value so you can add some octets to it with the -o option.

SNMP Login

See snmp info page

Vérifie par snmp v1 ou v3 l'occupation des disques, mais aussi de la swap, de la mémoire, etc.. tout ce qui est disponible en snmp par la table hrStorage.
Les disques sont sélectionnables par expression régulière compatible Perl

Requirements :

- Perl in /usr/bin/perl - or just run 'perl script'
- Net::SNMP
- file 'utils.pm' in plugin diretory (/usr/local/nagios/libexec)

Configurations examples

Changelog **: On CVS repository on sourceforge : http://nagios-snmp.cvs.sourceforge.net/nagios-snmp/plugins/.**

Examples :

All examples below are considering the script is local directory. Host to be checked is 127.0.0.1 with snmp community "public".

Get help	./check_snmp_storage.pl **-h**
List all storage	./check_snmp_storage.pl -H 127.0.0.1 -C public -m zzzz -w 80 -c 81 **-v**
snmpv3 login	./check_snmp_storage.pl -H 127.0.0.1 -l login -x passwd

Unix/Linux :

%used of /home is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public **-m /home -w 80% -c 90%**
%free of /home is above 10% and 5%	./check_snmp_storage.pl -H 127.0.0.1 -C public -m /home **-w 10% -c 5% -T pl**
Mb used of /home is less than 800 Mb and 900 Mb	./check_snmp_storage.pl -H 127.0.0.1 -C public -m /home **-w 800 -c 900 -T bu**
Mb free of /home is above 100Mb and 30Mb	./check_snmp_storage.pl -H 127.0.0.1 -C public -m /home **-w 100 -c 30 -T bl**
All mountpoints have %used less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public -m **/** -w 80% -c 90%
%used of / mountpoint only is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public -m **/ -r** -w 80% -c 90%
%used of mountpoint index 1 only is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public **-m 1 -p** -w 80% -c 90%
Swap %used is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public **-m Swap** -w 80% -c 90%
Memory %used is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public **-m "Real Memory"** -w 80% -c 90%

Windows :

%used of C is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public -m **^C:** -w 80% -c 90%
%used of C, D and E is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public -m **^[CDE]:** -w 80% -c 90%
%used of C+D+E is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public -m **^[CDE]: -s** -w 80% -c 90%

Memory %used is less than 80% and 90%	./check_snmp_storage.pl -H 127.0.0.1 -C public -m **^Virtual Memory$** -w 80% -c 90%

check_snmp_load.pl
check_manubulon_snmp_load.pl

Description :

Checks by snmp v1, v2c or v3 cpu or average load.
Works on Windows, Linux/Unix, AS400, Cisco, Cisco catalyst, HP Procurve, LinkProof, Blucoat, Nokia, Fortinet, Netscreen.

Check types are selected by the -T option :

-T value	System	-w & -c values
stand	standard MIBII (works with Windows)	1 value in %
netsl	Linux : load provided by Net SNMP	3 values : load average on 1 min, 5 min, 15 min (absolute)
netsc	Linux : CPU usage given by net-snmp	1 value in %
as400	as400 CPU usage	1 value in %
cisco	cisco CPU usage	3 values : CPU average on 5sec, 1 min & 5 min (%)
cata	cisco catalyst CPU usage	3 values : CPU average on 5sec, 1 min & 5 min (%)
nsc	Netscreen CPU usage	3 values : CPU average on 5sec, 1 min & 5 min (%)
fg	Fortigate CPU usage	1 value in %
bc	bluecoat CPU usage	1 value in %
nokia	nokia CPU usage	1 value in %
hp	HP procurve switch CPU usage	1 value in %
lp	Linkproof CPU usage	1 value in %
hpux	HP-UX load	3 values : load average on 1 min, 5 min, 15 min

Vérifie par snmp v1,v2c ou v3 la charge ou l'ocupation CPU.

Output of check_snmp_load.pl -h

```
SNMP Load & CPU Monitor for Nagios version 1.3
(c)2004-2006 Patrick Proy

Usage: ./check_snmp_load.pl [-v] -H <host> -C <snmp_community> [-2] | (-l login -x passwd [-X pass -L <authp>,<privp>]) [-p
<port>] -w <warn level> -c <crit level> -T=[stand|netsl|netsc|as400|cisco|cata|nsc|fg|bc|nokia|hp|lp] [-f] [-t <timeout>] [-V]
-v, --verbose
print extra debugging information
-h, --help
print this help message
-H, --hostname=HOST
name or IP address of host to check
-C, --community=COMMUNITY NAME
community name for the host's SNMP agent (implies v1 protocol)
-2, --v2c
Use snmp v2c
-l, --login=LOGIN ; -x, --passwd=PASSWD
Login and auth password for snmpv3 authentication
If no priv password exists, implies AuthNoPriv
-X, --privpass=PASSWD
Priv password for snmpv3 (AuthPriv protocol)
-L, --protocols=<authproto>,<privproto>
<authproto> : Authentication protocol (md5|sha : default md5)
<privproto> : Priv protocole (des|aes : default des)
-P, --port=PORT
SNMP port (Default 161)
-w, --warn=INTEGER | INT,INT,INT
1 value check : warning level for cpu in percent (on one minute)
3 value check : comma separated level for load or cpu for 1min, 5min, 15min
-c, --crit=INTEGER | INT,INT,INT
critical level for cpu in percent (on one minute)
1 value check : critical level for cpu in percent (on one minute)
3 value check : comma separated level for load or cpu for 1min, 5min, 15min
```

-T, --type=stand|netsl|netsc|as400|cisco|bc|nokia|hp|lp
CPU check :
stand : standard MIBII (works with Windows),
can handle multiple CPU.
netsl : linux load provided by Net SNMP
netsc : cpu usage given by net-snmp (100-idle)
as400 : as400 CPU usage
cisco : Cisco CPU usage
cata : Cisco catalyst CPU usage
nsc : NetScreen CPU usage
fg : Fortigate CPU usage
bc : Bluecoat CPU usage
nokia : Nokia CPU usage
hp : HP procurve switch CPU usage
lp : Linkproof CPU usage
-f, --perfparse
Perfparse compatible output
-t, --timeout=INTEGER
timeout for SNMP in seconds (Default: 5)
-V, --version
prints version number

SNMP Login

See snmp info page

Requirements :

- Perl in /usr/bin/perl - or just run 'perl script'
- Net::SNMP
- file 'utils.pm' in plugin diretory

Configurations examples

Changelog **: On CVS repository on sourceforge : http://nagios-snmp.cvs.sourceforge.net/nagios-snmp/plugins/.**

Examples :

All examples below are considering the script is local directory. Host to be checked is 127.0.0.1 with snmp community "public".

If multiple interfaces are selected, all must be up to get an OK result

Get help	./check_snmp_load.pl **-h**
Check loads on linux with Net-SNMP : checks the 1, 5 and 15 minutes load average.	./check_snmp_load.pl -H 127.0.0.1 -C public **-w 3,3,2 -c 4,4,3 -T netsl**
Check cpu load (generic) : checks the %used CPU for the last minute	./check_snmp_load.pl -H 127.0.0.1 -C public **-w 98% -c 99%**
Check cpu load on AS/400	./check_snmp_load.pl -H 127.0.0.1 -C public **-w 98% -c 99% -T as400**

check_snmp_memory.pl
check_manubulon_snmp_memory.pl

Description :

Checks by snmp v1, v2c or v3 :

- Memory and swap usage on Linux given by Net-snmp.
It checks memory and swap usage independantly : one warning and critical level for each of them.

- Memory usage on cisco routers or Pix : the plugin will add all of the memory pool and then checks the warning and critical levels.
On cisco routeurs, it will add 'IO' and 'Processor' memory
On Pix, it will check the memory used (one memory pool only on Pix).

- Memory usage on HP Procurve switch.
Memory segments will be added then the free memory will be checked.

Vérification de :
- l'utilisation mémoire et swap sur Linux/Net-snmp
- l'utilisation mémoire sur Cisco (routeurs et Pix).
Pour d'autres SE (Windows, AS400), lire la suite.

Checking memory for other os :

AS/400 : use the check_snmp_storage command to check the RAM

./check_snmp_storage.pl -H **<IP>** -C **<community>** -m RAM -s -w **<warn%>** -c **<crit%>**

Windows : use the check_snmp_storage command to check the virtual memory (RAM+Swap)

./check_snmp_storage.pl -H **<IP>** -C **<community>** -m "^Virtual Memory$" -w **<warn%>** -c **<crit%>**

Output of check_snmp_mem.pl -h <

```
SNMP Memory Monitor for Nagios version 1.1
(c)2004-2006 - Patrick Proy

Usage: ./check_snmp_mem.pl [-v] -H <host> -C <snmp_community> [-2] | (-l login -x passwd
[-X pass -L <authp>,<privp>])  [-p <port>] -w <warn level> -c <crit level> [-I|-N|-E] [-
f] [-m] [-t <timeout>] [-V]
-v, --verbose
   print extra debugging information (including interface list on the system)
-h, --help
   print this help message
-H, --hostname=HOST
   name or IP address of host to check
-C, --community=COMMUNITY NAME
   community name for the host's SNMP agent (implies SNMP v1 or v2c with option)
-2, --v2c
   Use snmp v2c
-l, --login=LOGIN ; -x, --passwd=PASSWD
   Login and auth password for snmpv3 authentication
   If no priv password exists, implies AuthNoPriv
-X, --privpass=PASSWD
   Priv password for snmpv3 (AuthPriv protocol)
-L, --protocols=<authproto>,<privproto>
   <authproto> : Authentication protocol (md5|sha : default md5)
   <privproto> : Priv protocole (des|aes : default des)
-P, --port=PORT
   SNMP port (Default 161)
-w, --warn=INTEGER | INT,INT
   warning level for memory in percent (0 for no checks)
```

```
     Default (-N switch) : comma separated level for Real Memory and Swap
     -I switch : warning level
-c, --crit=INTEGER | INT,INT
   critical level for memory in percent (0 for no checks)
     Default (-N switch) : comma separated level for Real Memory and Swap
     -I switch : critical level
-N, --netsnmp (default)
   check linux memory & swap provided by Net SNMP
-m, --memcache
   include cached memory in used memory (only with Net-SNMP)
-I, --cisco
   check cisco memory (sum of all memory pools)
-E, --hp
   check HP proccurve memory
-f, --perfdata
   Performance data output
-t, --timeout=INTEGER
   timeout for SNMP in seconds (Default: 5)
-V, --version
   prints version number
```

SNMP Login

See snmp info page

Requirements :

- Perl in /usr/bin/perl - or just run 'perl script'
- Net::SNMP
- file 'utils.pm' in plugin diretory (/usr/local/nagios/libexec)

Configurations examples

Changelog **: On CVS repository on sourceforge : http://nagios-snmp.cvs.sourceforge.net/nagios-snmp/plugins/.**

Examples :

All examples below are considering the script is local directory. Host to be checked is 127.0.0.1 with snmp community "public".

Get help	./check_snmp_mem.pl **-h**
Verbose output	./check_snmp_mem.pl -H <IP> -C <com> -w 80 -c 81 **-v**
snmpv3 login	./check_snmp_mem.pl -H 127.0.0.1 -l login -x passwd
	Unix/Linux
%used of - RAM < 99% and 100% - Swap : < 20% and 30%	./check_snmp_mem.pl -H <IP> -C <com> **-w 99,20 -c 100,30**
Same with no warning levels for memory	./check_snmp_mem.pl -H <IP> -C <com> -w **0**,20 -c 100,30
Check memory on Cisco	./check_snmp_mem.pl -H <IP> -C <com> **-I -w 90% -c 98%**

check_snmp_netint.pl / check_snmp_int.pl
check_manubulon_snmp_netint.pl

Description :

Checks by snmp (v1, v2c or v3) host interface state and usage.

Interfaces can be selected by regexp ('eth' will check eth0, eth1, eth2, ...).
If multiple interfaces are selected, all must be up to get an OK result

Vérifie par snmp v1, v2c ou v3 l'état des interfaces d'un hôte, leur utilisation (bande passante, erreurs, pertes) et sortir des informations de performances.
Ce script permet d'utiliser des expressions régulières pour sélectionner les interfaces : ex 'eth' va vérifier eth0, eth1, ...
Il permet également de lister toutes les interfaces disponibles sur la machine cible (option -v) pour voir leur nom en SNMP.
Pour les francophones, je ferais un manuel en Français quand j'aurais le temps... (et toute aide est bienvenue).

Output of check_snmp_int.pl -h

```
SNMP Network Interface Monitor for Nagios version 1.4.8
GPL licence, (c)2004-2007 Patrick Proy

Usage: ./check_snmp_int.pl [-v] -H <host> -C <snmp_community> [-2] | (-l login -x passwd [-X pass -L <authp>,<privp>) [-p
<port>] -n <name in desc_oid> [-i -a -D] [-r] [-f[eSyY]] [-k[qBMGu] -g -w<warn levels> -c<crit levels> -d<delta>] [-o
<octet_length>] [-t <timeout>] [-s] --label [-V]
-v, --verbose
print extra debugging information (including interface list on the system)
-h, --help
print this help message
-H, --hostname=HOST
name or IP address of host to check
-C, --community=COMMUNITY NAME
community name for the host's SNMP agent (implies v1 protocol)
-l, --login=LOGIN ; -x, --passwd=PASSWD, -2, --v2c
Login and auth password for snmpv3 authentication
If no priv password exists, implies AuthNoPriv
-2 : use snmp v2c
-X, --privpass=PASSWD
Priv password for snmpv3 (AuthPriv protocol)
-L, --protocols=<authproto>,<privproto>
<authproto> : Authentication protocol (md5|sha : default md5)
<privproto> : Priv protocole (des|aes : default des)
-P, --port=PORT
SNMP port (Default 161)
-n, --name=NAME
Name in description OID (eth0, ppp0 ...).
This is treated as a regexp : -n eth will match eth0,eth1,...
Test it before, because there are known bugs (ex : trailling /)
-r, --noregexp
Do not use regexp to match NAME in description OID
-i, --inverse
Make critical when up
-a, --admin
Use administrative status instead of operational
-D, --dormant
Dormant state is an OK state
-o, --octetlength=INTEGER
max-size of the SNMP message, usefull in case of Too Long responses.
Be carefull with network filters. Range 484 - 65535, default are
usually 1472,1452,1460 or 1440.
-f, --perfparse
Perfparse compatible output (no output when interface is down).
-e, --error
Add error & discard to Perfparse output
-S, --intspeed
Include speed in performance output in bits/s
-y, --perfprct ; -Y, --perfspeed
-y : output performance data in % of max speed
-Y : output performance data in bits/s or Bytes/s (depending on -B)
-k, --perfcheck ; -q, --extperfcheck
-k check the input/ouput bandwidth of the interface
-q also check the error and discard input/output
--label
```

Add label before speed in output : in=, out=, errors-out=, etc...
-g, --64bits
Use 64 bits counters instead of the standard counters when checking
bandwidth & performance data for interface >= 1Gbps.
You must use snmp v2c or v3 to get 64 bits counters.
-d, --delta=seconds
make an average of <delta> seconds (default 300=5min)
-B, --kbits
Make the warning and critical levels in K|M|G Bits/s instead of K|M|G Bytes/s
-G, --giga ; -M, --mega ; -u, --prct
-G : Make the warning and critical levels in Gbps (with -B) or GBps
-M : Make the warning and critical levels in Mbps (with -B) or MBps
-u : Make the warning and critical levels in % of reported interface speed.
-w, --warning=input,output[,error in,error out,discard in,discard out]
warning level for input / output bandwidth (0 for no warning)
unit depends on B,M,G,u options
warning for error & discard input / output in error/min (need -q)
-c, --critical=input,output[,error in,error out,discard in,discard out]
critical level for input / output bandwidth (0 for no critical)
unit depends on B,M,G,u options
critical for error & discard input / output in error/min (need -q)
-s, --short=int
Make the output shorter : only the first <n> chars of the interface(s)
If the number is negative, then get the <n> LAST caracters.
-t, --timeout=INTEGER
timeout for SNMP in seconds (Default: 5)
-V, --version
prints version number
Note : when multiple interface are selected with regexp,
all be must be up (or down with -i) to get an OK result.

Standard checks

The script will check interface operationnal status using the MIB-II table. The interface is (are) selected by the -n option.
This option will be treated as a regular expression (eth will match eth0, eth1, eth2...). You can disable this with the -r option : the interface will be selected if it's description exactly matches the name given by -n

The interfaces are selected by their description in the MIB-II table.
To see how interface looks like by snmp, you can list all of them with the '-v' switch.

The script will return OK if ALL interfaces selected are UP, or CRITICAL if at least one interface is down.

You can make the script return a OK value when all interfaces are down (and CRITICAL when at least one is up) with the -i option.

You can make the same tests on administrative status instead with the -a option.

If you have ISDN interface, and want that DORMANT state returns ok, put -D.

To make output shorter, specially when you have multiple interface, you can put the -s option.
It will get only the first <n> caracters of the interface descrition. If the number is negative then get the last <n> caracters.

Ex : EL20005 3Com Gigabit NIC (3C2000 Family)
-s 4 will output : "EL20".
-s -4 will output : "ily)".

Performance output

-f option : activate performance output (default the In/out octet as a counter).
-e option : in/out errors and discarded packets. -f must also be set.
-S option : Include speed in performance output in bits/s as '<interface_name>_speed_bps'
-y option : output performance data in % of interface speed
-Y option : output performance data in bits/s or Bytes/s (depending on -B)

Note : -y and -Y options need the usage check to be active (-k)

Warning : the counters needed by -e are not always available on all machines (ex Nokia IP)

Usage check (-k)

A temporary file will be created in "/tmp" by default : this can be changed at the beginning of the script.
The file name will be : tmp_Nagios_int.<host IP>.<Interface name>. One file will be created by interface.

The status UNKNOWN is returned when the script doesn't have enough information (see -d option).

You will have to tell the warning and critical levels, separated with "," and you can use decimal (ex : 10.3).
For standard checks (no "-q" option) :
-w *<In warn>*,*<Out warn>* -c *<In warn>*,*<Out warn>*
In warn : warning level for incomming traffic
Out warn : warning level for outgoing traffic
In crit : critical level for incomming traffic
Out crit : critical level for outgoing traffic

The unit for the check depends on the -B, -M and -G option :

	-B set	-B not set
-M & -G not set	Kbps	KBps
-M set	Mbps	MBps
-G set	Gbps	GBps

It is possible to put warning and critical levels with -b option.
0 means no warning or critical level checks

When the extended checks are activated (-q option), the warning levels are
-w *<In bytes>*,*<Out bytes>*,*<In error>*,*<Out error>*,*<In disc>*,*<Out disc>* -c *<In warn>*,*<Out warn>*, ...
In error : warn/crit level in inboud error/minute
Out error : warn/crit level in outbound error/minute
In disc : warn/crit level in inboud discarded packets/minute
Out disc : warn/crit level in outbound discarded packets/minute

-k : activates the standard usage feature
-q : activates the extended usage
-d : delta in seconds (default is 300s)
-w : warning levels
-c : critical levels

-d: delta time
You can put the delta time as an option : the "delta" is the prefered time between two values that the script will use to calculate the average Kbytes/s or error/min. The delta time should (not must) be bigger than the check interval.
Here is an example : Check interval of 2 minutes and delta of 4min

T0 : value 1 : can't calculate usage
T0+2 : value 2 : can't calculate usage
T0+4 : value 3 : usage=(value3-value1)/((T0+4)-T0)
T0+6 : value 4 : usage=(value4-value2)/((T0+6)-T0+2)
(Yes I know TO+4-T0=4, it's just to explain..)
.....

The script will allow 10% less of the delta and 300% more than delta as a correct interval.
For example, with a delta of 5 minutes, the acceptable interval will be between 4'30" and 15 minutes.

Msg size option (*-o* option)

In case you get a "ERROR: running table : Message size exceeded maxMsgSize" error, you may need to adjust the maxMsgSize, i.e. the maximum size of snmp message with the -o option. Try a value with the -o AND the -v option : the script will output the actual value so you can add some octets to it with the -o option.

--label option

This option just put label before the speed output :
Without : eth1:UP (10.3Kbps/4.4Kbps), eth0:UP (10.9Kbps/16.4Kbps):2 UP: OK
With : eth1:UP (in=14.4Kbps/out=6.2Kbps), eth0:UP (in=15.3Kbps/out=22.9Kbps):2 UP: OK

SNMP Login

See snmp info page

Requirements :

- Perl in /usr/bin/perl - or just run 'perl script'
- Net::SNMP
- file 'utils.pm' in plugin diretory (/usr/local/nagios/libexec)

Configuration examples

Examples :

All examples below are considering the script is local directory. Host to be checked is 127.0.0.1 with snmp community "public".

If multiple interfaces are selected, all must be up to get an OK result

Get help	./check_snmp_int.pl -h
List all interfaces	./check_snmp_int.pl -H 127.0.0.1 -C public -n zzzz -v
snmpv3 login	./check_snmp_int.pl -H 127.0.0.1 -l login -w passwd
Check eth0 interface is up	./check_snmp_int.pl -H 127.0.0.1 -C public -n eth0 -r
Check that all eth interface are up	./check_snmp_int.pl -H 127.0.0.1 -C public -n eth
Check that all ppp interface are down	./check_snmp_int.pl -H 127.0.0.1 -C public -n ppp -i
Check that all eth interface are administratively up	./check_snmp_int.pl -H 127.0.0.1 -C public -n eth -a
Check that FastEternet0/11 to 0/14 are up (Cisco)	./check_snmp_int.pl -H 127.0.0.1 -C public -n "Fast.*0.1[1234]"
Check the eth0 usage Note : no critical inbound (0)	./check_snmp_int.pl -H 127.0.0.1 -C public -n eth0 -k -w 200,400 -c 0,600

Changelog : On CVS repository on sourceforge : **http://nagios-snmp.cvs.sourceforge.net/nagios-snmp/plugins/**.

check_snmp_process.pl
check_manubulon_snmp_process.pl

Description :

Checks by snmp v1 or v3 if a process is running and how many instances are running (minimum & maximum).
It is also possible to check memory and cpu used by one or a group of process

Works on Windows, Linux/Unix, AS400.

Vérifie par snmp v1 ou v3 si un process tourne et combien d'instances de ce process tournent (minimum et maximum).
Il est également possible de vérifier la mémoire et le cpu utilisé.

Standard checks

The plugin checks if there is at least one process matching the filter (**-n** option) when no warning or critical levels are set.
The filter is treated as a regular expression by default, but you can deactivate this (**-r**)

With the following options, you can add to your process selection :

-f : get full path of the script instead of only it's name

-A : add parameters with the script name

Option	how the script will see the process
None	named
-f	/usr/sbin/named
-A	named -u named -t /var/named/chroot
-f -A	/usr/sbin/named -u named -t /var/named/chroot

Warning : the -f & -A option will not function properly for Windows hosts (the snmp agent don't give this information)

You can use -w and -c options to set the warning and critical levels :
-w <minW>,<maxW> : with minW and maxW the minimum and maximum number of processes.

-c <minC>,<maxC> : same thing
Of course : **minC <= minW < maxW <=maxC**

You can omit <maxW> and <maxC>

Saying N is the current number of processes
- N < minC : critical
- minC < N <=minW : warning
- minW< N <= maxW : OK
- maxW< N <= maxC : warning
- maxC < N : critical

Memory checks

The **-m** option can check the memory used by the selected processes.
By default, this will select the process wich use the maximum memory. The **-a** switch will make an average

Ex : **-m 7,20** will send a warning if a process uses more than 7 Mb, and critical for more than 20Mb.

CPU checks

When you use the **-u** option, a temporary file will be created in "/tmp" by default : this can be changed at the beginning of the script.
The file name will be : tmp_Nagios_proc.<host IP>.<process filter>.

The -u option will add all the cpu used by all selected process and the make the check

-u 91,95 : will send a warning if more than 91% of cpu is used, and critical if more than 95% is used.

On multiprocessor hosts, the % of cpu use can be > 100% : on a 4 CPU host, cpu usage can go up to 400% (the script doesn't check if a host is multiprocessor or not).

The script curently wants a minimum of 5 minutes between values taken from host (can be changed at the beginning of the scripts). You can check more than once every 5 minutes but don't put check-interval to more than 15 minutes.
When the script doesn't have enough data to compute the CPU use (for example, the first time it is run), then it will return a UNKNOWN status.

Msg size option (*-o* option)

In case you get a "ERROR: running table : Message size exceeded maxMsgSize" error, you may need to adjust the maxMsgSize, i.e. the maximum size of snmp message with the -o option. Try a value with the -o AND the -v option : the script will output the actual value so you can add some octets to it with the -o option.

SNMP Login

See snmp info page

Requirements :

- Perl in /usr/bin/perl - or just run 'perl script'
- Net::SNMP
- file 'utils.pm' in plugin diretory

Configurations examples

Changelog : On CVS repository on sourceforge : http://nagios-snmp.cvs.sourceforge.net/nagios-snmp/plugins/.

Examples :

All examples below are considering the script is local directory. Host to be checked is 127.0.0.1 with snmp community "public".

If multiple interfaces are selected, all must be up to get an OK result

Get help	./check_snmp_process.pl **-h**
snmpv3 login	./check_snmp_process.pl -H 127.0.0.1 -l login -x passwd
Check if at least one process matching http is running	./check_snmp_process.pl -H 127.0.0.1 -C public **-n http**
Result example :	3 process matching http : > 0 : OK
Check if at least 3 process matching http are running	./check_snmp_process.pl -H 127.0.0.1 -C public -n http **-w 2 -c 0**
Result example : (<=2 will return warning, 0 critical)	3 process matching httpd : > 2 : OK
Check if at least one process named "httpd" exists (no regexp)	./check_snmp_process.pl -H 127.0.0.1 -C public -n http **-r**
Result example :	3 process **named** httpd : > 0 : OK
Check process by their full path : check process of /opt/soft/bin/ (at least one)	./check_snmp_process.pl -H 127.0.0.1 -C public **-n /opt/soft/bin/ -f**
Check that at least 3 process but not more than 8 are running	./check_snmp_process.pl -H 127.0.0.1 -C public -n http **-w 3,8 -c 0,15**
Same checks + checks maximum	./check_snmp_process.pl -H 127.0.0.1 -C public -n http -w 3,8 -c 0,15 **-m 9,25**

memory used by process (in Mb) : warning and critical levels	
Same check but sum all CPU used by all selected process	./check_snmp_process.pl -H 127.0.0.1 -C public -n http -w 3,8 -c 0,15 -m 9,25 **-u 70,99**

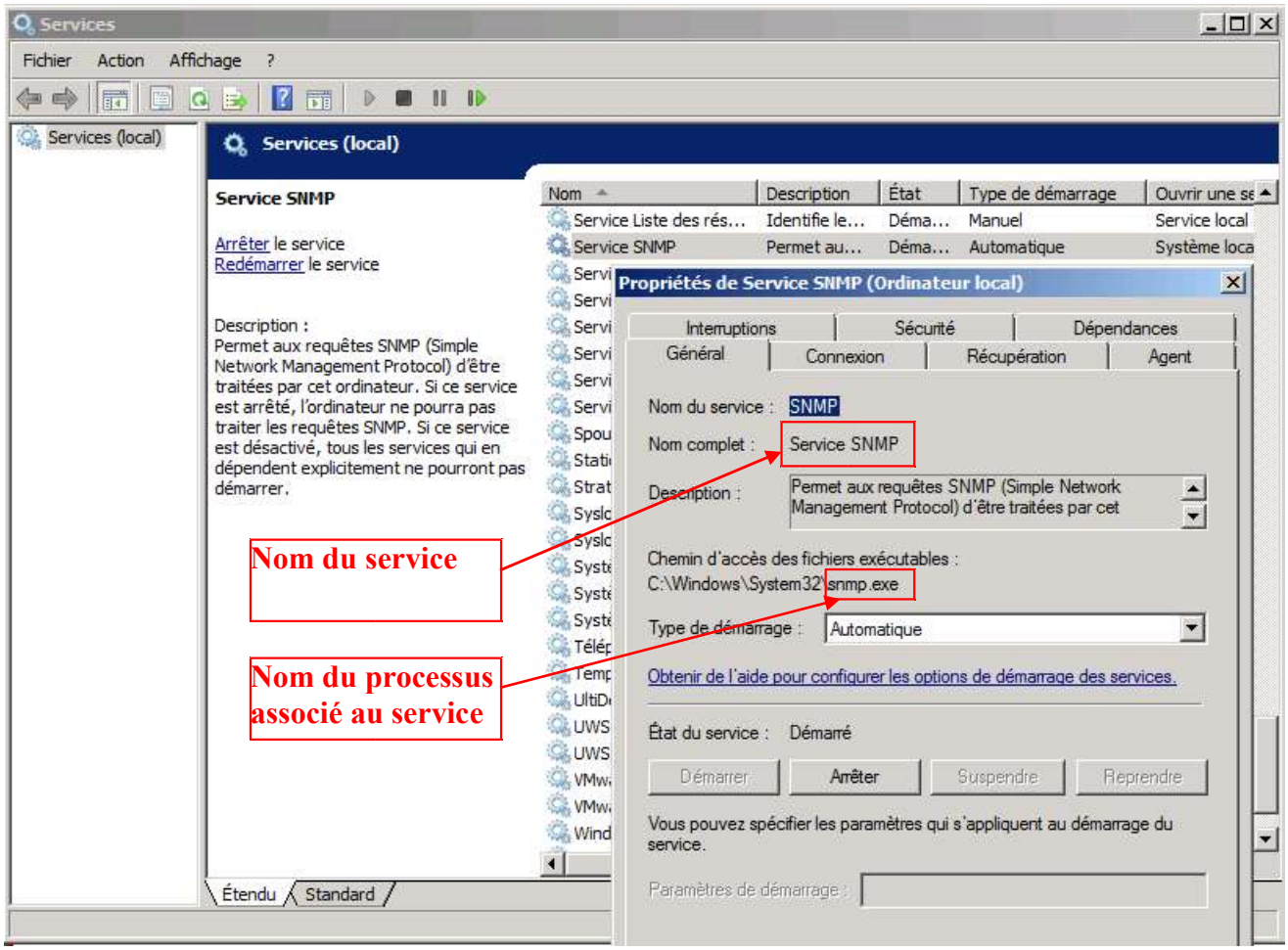
Output of check_snmp_process.pl -h

```
Usage: ./check_snmp_process.pl [-v] -H <host> -C <snmp_community> [-2] | (-l login -x passwd) [-p <port>] -n <name> [-w
<min_proc>[,<max_proc>] -c <min_proc>[,max_proc] ] [-m<warn Mb>,<crit Mb> -a -u<warn %>,<crit%> -d<delta> ] [-t
<timeout>] [-o <octet_length>] [-f -A -F ] [-r] [-V] [-g]
-v, --verbose
print extra debugging information (and lists all storages)
-h, --help
print this help message
-H, --hostname=HOST
name or IP address of host to check
-C, --community=COMMUNITY NAME
community name for the host's SNMP agent (implies SNMP v1 or v2c with option)
-l, --login=LOGIN ; -x, --passwd=PASSWD, -2, --v2c
Login and auth password for snmpv3 authentication
If no priv password exists, implies AuthNoPriv
-2 : use snmp v2c
-X, --privpass=PASSWD
Priv password for snmpv3 (AuthPriv protocol)
-L, --protocols=<authproto>,<privproto>
<authproto> : Authentication protocol (md5|sha : default md5)
<privproto> : Priv protocole (des|aes : default des)
-p, --port=PORT
SNMP port (Default 161)
-n, --name=NAME
Name of the process (regexp)
No trailing slash !
-r, --noregexp
Do not use regexp to match NAME in description OID
-f, --fullpath
Use full path name instead of process name
(Windows doesn't provide full path name)
-A, --param
Add parameters to select processes.
ex : "named.*-t /var/named/chroot" will only select named process with this parameter
-F, --perfout
Add performance output
outputs : memory_usage, num_process, cpu_usage
-w, --warn=MIN[,MAX]
Number of process that will cause a warning
-1 for no warning, MAX must be >0. Ex : -w-1,50
-c, --critical=MIN[,MAX]
number of process that will cause an error (
-1 for no critical, MAX must be >0. Ex : -c-1,50
Notes on warning and critical :
with the following options : -w m1,x1 -c m2,x2, you must have : m2 <= m1 < x1 <= x2
you can omit x1 or x2 or both
-m, --memory=WARN,CRIT
checks memory usage (default max of all process)
values are warning and critical values in Mb
-a, --average
makes an average of memory used by process instead of max
-u, --cpu=WARN,CRIT
checks cpu usage of all process
values are warning and critical values in % of CPU usage
if more than one CPU, value can be > 100% : 100%=1 CPU
-d, --delta=seconds
make an average of <delta> seconds for CPU (default 300=5min)
-g, --getall
In some cases, it is necessary to get all data at once because process die very frequently.
This option eats bandwidth an cpu (for remote host) at breakfast.
-o, --octetlength=INTEGER
max-size of the SNMP message, usefull in case of Too Long responses.
Be carefull with network filters. Range 484 - 65535, default are
usually 1472,1452,1460 or 1440.
-t, --timeout=INTEGER
timeout for SNMP in seconds (Default: 5)
-V, --version
prints version number
Note :
```

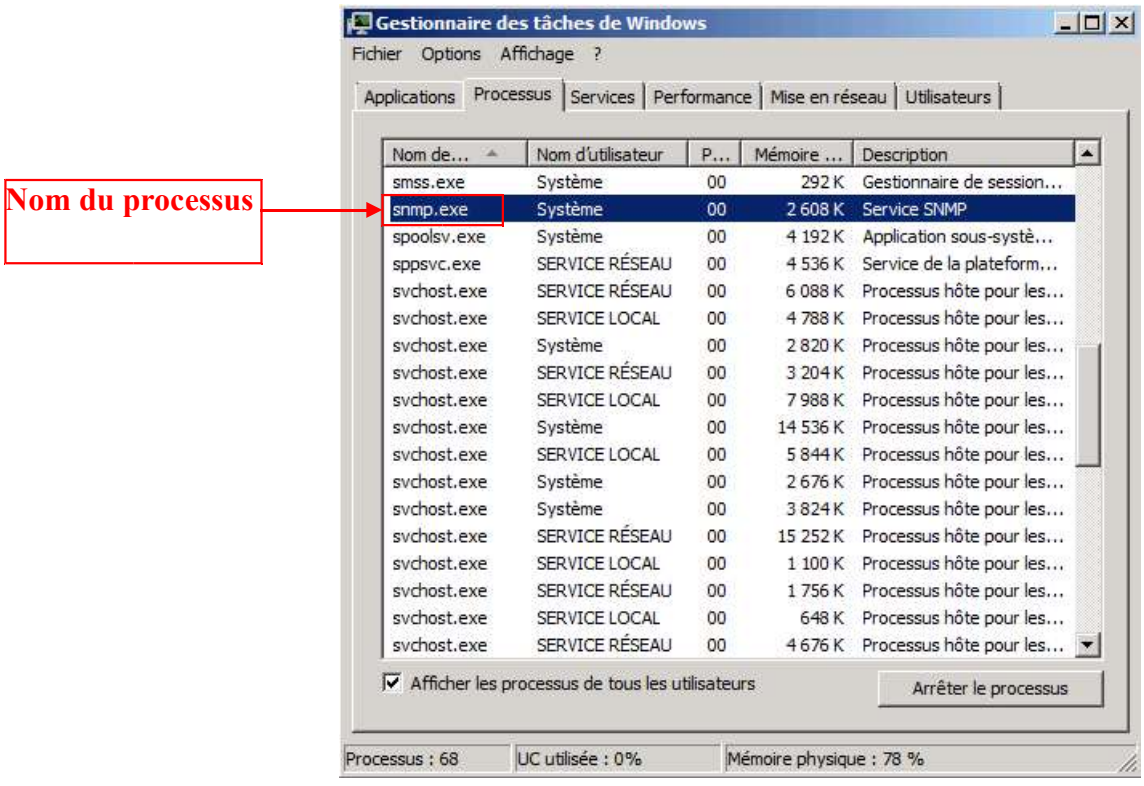
CPU usage is in % of one cpu, so maximum can be 100% * number of CPU
example :
Browse process list : <script> -C <community> -H <host> -n <anything> -v
the -n option allows regexp in perl format :
All process of /opt/soft/bin : -n /opt/soft/bin/ -f
All 'named' process : -n named

Remarques :

- Sur les machines Windows, le nom du processus correspondant à un nom de service Windows est obtenu en affichant les propriétés du service (bouton *Démarrer* / *Outils d'administration* / *Services*) :



On retrouve ce processus dans la liste des processus (*CTRL ALT SUPPR* puis *Gestionnaire des tâches*):



- Sur les machines Linux (Debian, Pfsense, ...), un processus (*process*) est parfois appelé démon (*daemon*) ou service (*service*).

Nom du processus associé au service