NAME

nsca–ng.cfg – NSCA–ng server configuration file

SYNOPSIS

/etc/nsca–ng.cfg

DESCRIPTION

The **nsca–ng**(8) process reads configuration data from the file specified with **–c** on the command line or from */etc/nsca–ng.cfg*.

File Format

Zero or more global settings and one or more authorizations must be defined in the configuration file (see the **Global Settings** subsection and the **Authorizations** subsection, respectively). They may appear in arbitrary order. An authorization is specified using the **authorize** keyword followed by a (possibly quoted) client identity string and a brace-enclosed block of corresponding authorization settings. However, an authorization setting may also be specified as a global setting outside of these **authorize** sections. In this case, it serves as a global fallback for authorization sections that don't define the setting in question.

Global settings and authorization settings are defined by specifying a variable name followed by an equals sign ("=") and a value (or possibly a list of values). Values can be strings, integers, or floating-point numbers. Strings have to be enclosed in single or double quotes if they contain whitespace characters, hash mark characters, or literal quotation marks. Otherwise, quoting is optional. To specify a literal single or double quote in a string, either escape it by preceding it with a backslash ("\") or quote the string using the other quote character. A literal backslash must be preceded with a second backslash if the string is enclosed in double quotes.

A variable can be set to the value of an environment variable by specifying **${***FOO***},** where *FOO* is the name of the environment variable. The same can be done by specifying **${***FOO***:–***bar***},** except that in this case, the value *bar* will be assigned when the environment variable *FOO* is not set.

Any whitespace surrounding tokens is ignored. Empty lines and comments are also ignored. Comments are introduced with a hash mark character ("#") and span to the end of the line. If the last character of a line is a backslash ("\"), the subsequent line is treated as a continuation of the current line (and the backslash is otherwise ignored).

The special directive **include("***file***")** tells **nsca–ng**(8) to treat the contents of the specified *file* as if those contents had appeared at the point where this directive appears. If a directory is specified instead of a *file*, all files with a *.cfg* or *.conf* extension in this directory and all subdirectories will be included. Symbolic links are followed.

In the following subsections, the type of each value is denoted after an equals sign in angle brackets.

Global Settings

The **nsca–ng**(8) server recognizes the following global variables.

chroot = <*string*>
: On startup, perform a **chroot**(2) operation to the specified directory. By default, **nsca–ng**(8) does not call **chroot**(2). If this directive is used, the **command_file**, **pid_file**, and **temp_directory** must be specified relative to this directory.

command_file = <*string*>
: Submit monitoring commands to the specified path name. This should be the named pipe (FIFO) that Nagios (or a compatible monitoring solution) checks for external commands to process. The default is */var/nagios/rw/nagios.cmd*. The specified value will be overridden if **nsca–ng**(8) is called with the **–C** option.

listen = <*string*>
: Bind to the specified IP address or host name. The default setting is "*", which tells **nsca–ng**(8) to listen on all available interfaces. A colon (":") followed by a service name or port number may be appended to override the default port (5668) used by the **nsca–ng**(8)

server. The specified value will be ignored if **nsca–ng**(8) is called with the **–b** option, of if it is socket activated by **systemd**(1).

log_level = <*integer*>
Use the specified log level, which must be an integer value between 0 and 5 inclusive. A value of 0 tells **nsca–ng**(8) to generate only fatal error messages, 1 adds non-fatal error messages, 2 adds warnings, 3 additionally spits out every submitted command (plus startup and shutdown notices), 4 also logs each message sent or received at the protocol level, and 5 generates additional debug output. The default log level is 3. The specified value will be overridden if **nsca–ng**(8) is called with the **–l** option.

max_command_size = <*integer*>
Refuse monitoring commands (including check result submissions) which are longer than the specified number of bytes. Setting this variable to 0 tells **nsca–ng**(8) to accept commands of arbitrary length. The default value is 16384.

max_queue_size = <*integer*>
Don't queue more than the specified number of megabytes worth of monitoring commands while Nagios isn't running (or not reading the command file). When the amount of available data exceeds this threshold, the queued data is thrown away. If this variable is set to 0, **nsca–ng**(8) queues an unlimited amount of data (until it exits due to running out of memory). The default value is 1024 (i.e., 1 gigabyte).

pid_file = <*string*>
During startup, try to create and lock the specified file and write the process ID of the **nsca–ng**(8) daemon into it. Bail out if another process holds a lock on that file. By default, no such PID file is written. The specified value will be overridden if **nsca–ng**(8) is called with the **–p** option.

temp_directory = <*string*>
Write temporary files to the specified directory. Temporary files are only written if clients submit very large commands (which cannot be written to the named pipe atomically). It is recommended to specify a directory which resides on a memory file system. By default, */tmp* is used.

timeout = <*floating–point*>
Close the connection if a client didn't show any activity for the specified number of seconds. If this value is set to 0.0, **nsca–ng**(8) won't enforce connection timeouts. The default setting is 60.0 seconds.

tls_ciphers = <*string*>
Limit the acceptable TLS-PSK cipher suites to the specified list of ciphers. The format of the string is described in the **ciphers**(1) manual. By default, the ciphers in the list `PSK-AES256-CBC-SHA:PSK-AES128-CBC-SHA:PSK-3DES-EDE-CBC-SHA:PSK-RC4-SHA` will be accepted.

user = <*string*>
Switch to the specified user, and to the groups the user belongs to. This is done early on startup: after the configuration file has been read, but before the listening socket and (possibly) the PID file are created. By default, **nsca–ng**(8) runs with the privileges of the invoking user.

Authorizations

As mentioned above, an authorization section is introduced with the **authorize** keyword and a client identity field followed by a brace-delimited block of one or more authorization settings. A client provides its identity during the connection handshake. The server uses the provided identity string for looking up the **authorize** section applicable to the client. The corresponding section, if any, defines the authentication and authorization settings for the client in question. If no section explicitly defined for this client identity is found, but a section for the special client identity "*" (including the quotes) is defined, this section is used as a fallback. Note

that no other wildcard characters are available, and that the "*" character has no special meaning in the client identity field except when specified exactly as described.

Within the brace-delimited block of an authorization section, values may be assigned to the variables listed below. The pattern strings assigned to the **commands**, **hosts**, and **services** variables are POSIX "extended" regular expressions, but with an implicit "^" at the beginning and "$" at the end of the patterns. Multiple patterns can be specified as a brace-enclosed, comma-separated list; check results and commands will then be accepted if they match any of the specified patterns. Commands and check results will be rejected unless these settings authorize the client to submit them.

commands = *<(list of) string(s)>*
: Match the specified regular expression(s) against submitted monitoring commands and accept commands that match any of these expressions. The patterns are matched against the full command string supplied by the client, *except* for the leading bracketed timestamp and any whitespace following that timestamp.

hosts = *<(list of) string(s)>*
: Match the specified regular expression(s) against the "host name" field of client-supplied PROCESS_HOST_CHECK_RESULT commands and accept such commands if they match any of these expressions.

password = *<string>*
: Reject connections from clients that don't use the specified password. This setting is mandatory.

services = *<(list of) string(s)>*
: Match the specified regular expression(s) against the "service description" field of client-supplied PROCESS_SERVICE_CHECK_RESULT commands and accept such commands if they match any of these expressions. If a specified string includes one or more at signs ("@"), only the part preceding the last of these at signs is matched against the "service description" field. The part following this at sign is used as a separate pattern which is matched against the "host name" field of the same command. A service check result is then accepted only if both matches succeed for a given command.

EXAMPLES

The **/etc/nsca–ng.cfg** file might look similar to the following example.

```
user = "nagios"
chroot = "/var/nagios" # Other paths are relative to this one!
command_file = "/rw/nagios.cmd"
pid_file = "/run/nsca-ng.pid"
temp_directory = "/dev/shm"
listen = "monitoring.example.com:5668"
tls_ciphers = "PSK-AES256-CBC-SHA"
log_level = 3
max_command_size = 65536
max_queue_size = 128
timeout = 15.0

#
# Authenticated "root" clients may submit arbitrary check
# results and any other monitoring commands (see:
# <http://nagios.org/developerinfo/externalcommands/>).
#
authorize "root" {
    password = "g3m25sMCUAO4NecZGld1H4xcJ9uDWvhH"
    commands = ".*"
```

```
}

#
# Authenticated "checker" clients may submit arbitrary check
# results, but no other commands.
#
authorize "checker" {
    password = "ilzNanlE9XjMLdjrMkXnk09XBCTFQrj5"
    hosts = ".*"
    services = ".*"
}

#
# Authenticated "web-checker" clients may submit check results
# for arbitrary services on hosts whose names begin with "www".
#
authorize "web-checker" {
    password = "m2uaIWwiq3AIqN55m3QdjwptkU1Q4Oov"
    services = ".+@www.*"
}

#
# Authenticated "nsca-checker" clients may talk to the NSCA-ng
# server, but may not submit anything to Nagios.
#
authorize "nsca-checker" {
    password = "ceOKwxpz14lKXroC4yUjJZbov6VAyKuT"
}

#
# Other authenticated clients may submit check results for the
# "disk", "swap", and "load" services on arbitrary hosts.
#
authorize "*" {
    password = "awHW5vxr3DcA9EvcUC9T3a90QfEexsWd"
    services = {
        "disk",
        "swap",
        "load"
    }
}
```

CAVEATS

Please set the permissions appropriately to make sure that only authorized users can access the **/etc/nsca–ng.cfg** file.

SEE ALSO

nsca–ng(8), **send_nsca**(8), **send_nsca.cfg**(5), **regex**(7)

http://www.nagios.org/developerinfo/externalcommands/

AUTHOR

Holger Weiss <holger@weiss.in-berlin.de>