NAME

thresh – a data-directed SNMP threshold poller

SYNOPSIS

thresh [*varname=value ...*]

DESCRIPTION

thresh is a data-directed SNMP threshold poller, and uses the file system for configuration. status, and logging. Each host or device to be monitored is configured in a separate directory, using files listing SNMP variables, values, and a comparison indicator.

In normal operation, **thresh** starts scanning a data hierarchy (as described in **theshdata**(5)) at a particular directory (set by the *topdir* variable), reading *DEFAULTS* files, variable files, querying hosts and devices, and recording and reporting the results.

thresh variables, as described in **threshvars**(5), and set in *DEFAULTS* files or on the command line, change **thresh**'s default behaviour and notification mechanisms. Any *varname=value* settings on the command line override both the built-in defaults and the settings in any *DEFAULTS* files encountered during processing.

thresh would typically be called periodically by **cron**(8).

EXAMPLES

In normal use:

    % thresh

To use a non-default start directory:

    % thresh topdir=/some/other/place

To traverse the data hierarchy and provide information on what would normally be queried:

    % thresh walkonly=true

To do almost nothing:

    % thresh 'ignore=*'

NOTE

thresh is written in **Tcl**(n), using **scotty**(1) and the **Tnm**(n) network management extensions.

FILES

thresh uses just about any file and directory names. The name *.thresh* is reserved for naming the subdirectories used by **thresh** to store status and logging information.

Any files matching *.thresh/log_** are log files, which will grow without bound, and which you should arrange to rotate, archive, or truncate periodically.

BUGS

thresh currently only works with SNMP V1. The logging to **syslog**(3) should be internalized in some way, and not depend on **logger**(1). There should be some mechanism for "including" one file from another, to reduce the dependance on symbolic links for sharing files. **thresh** is unlikely to scale to handle arbitrarily large networks.

SEE ALSO

threshdata(5)
threshvars(5)
scotty(1)
Tcl(n)
Tnm(n)

AUTHOR

John Sellens

NAME

threshdata – thresh data hierarchy description

DESCRIPTION

The **thresh**(1) SNMP poller uses a configuration hierarchy to direct its actions, maintain its status information, and store its logs.

Each directory in the configuration hierarchy (under the *topdir* directory) is assumed to relate to a network host or device, or to an intermediate name in a DNS naming hierarchy.

NAMING

By default, the *topdir* directory is assumed to refer to a device named “” – the empty string. Each directory below *topdir* normally adds one more element on the right hand end of a DNS name. For example, below *topdir*, the directory

    org/usenix/conference

is related to the DNS sub-domain ‘‘conference.usenix.org’’, and the directory

    org/usenix/conference/ts1

is related to the device ‘‘ts1.conference.usenix.org’’. This naming relation can be overridden by the use of the *name* variable.

FILES

Each directory may contain a *DEFAULTS* file, which contains variable settings (see **threshvars**(5)) that apply to that directory, and to all directories below that point, unless overridden on the command line or by other, lower, *DEFAULTS* files.

Any other files found in a directory (other than those ignored by the *baseignore* and *ignore* variables) are assumed to contain a list of SNMP variables to monitor. **thresh** uses sub-directories named *.thresh* to store status and log information.

thresh data files consist of zero or more lines in the following format:

    <ws>type<ws>variable-or-OID<ws>value<ws>
    <ws># comment ...
    <ws>

where ‘‘<ws>’’ indicates white space.

The data file elements are defined as follows:

type A single capital letter indicating the comparison to be made in determining ‘‘normal’’.

- C Changeable – the variable’s value may change, but should be reported each time it changes. This is useful for semi-static data, or for monitoring things such as device interface status changes.
- G Greater than – the variable is reported if its current value is less than or equal to the specified value.
- I Increasing – the variable is reported if its current value is less than its previous value. This is handy for watching for reset times, such as the ‘‘system.sysUpTime.0’’ variable resetting when a device reboots.
- L Less than – the variable is reported if its current value is greater than or equal to the specified value.
- S Static – the variable is reported if its current value is not equal to the specified value. If no value is specified, then it is compared against the first-retrieved value of the variable. This is useful for monitoring things that should never change, such as ‘‘system.sysName.0’’.
- V Variable – the value can be anything, but it is queried and tracked to allow for later investigation or review.

variable-or-OID
: An SNMP variable name (or name fragment that **scotty**(1) can interpret) or SNMP OID (e.g. 1.3.6.1.2.1.1.3.0) to be monitored.

value
: A value for the comparison. Required for G and L, optional for S, and not allowed for C, I, and V.

SEE ALSO

thresh(1)
threshvars(5)

AUTHOR

John Sellens

NAME

threshmail − mail notifier for thresh messages

SYNOPSIS

threshmail *recipient ...*

DESCRIPTION

threshmail expects notification messages from **thresh**(1) on its standard input, which it appropriately reformats into a mail message, and sends to every recipient given on the command line.

threshmail would typically be set as the *notifier* in a **thresh** *DEFAULTS* file.

AUTHOR

John Sellens

NAME

threshvars – configuration variables understood by thresh

DESCRIPTION

thresh(1) understands and observes certain configuration variables. Those variables can be provided on the command line or in files named *DEFAULTS* within the **thresh** data hierarchy.

Variable names are case sensitive and are expected to be in lower case letters.

DEFAULTS FILES

DEFAULTS files consist of zero or more lines in the following format:

<ws>VARNAME<ws>=<ws>VALUE<ws>
<ws># comment ...
<ws>

where ‘‘<ws>’’ indicates white space. Values can contain embedded blanks.

Variables set by a *DEFAULTS* file apply at that level of the data hierarchy and below, unless overridden on the command line or in a *DEFAULTS* file further down the tree.

GENERAL VARIABLES

debug Generate debugging output.
Boolean. Default: true

verbose Generate informational messages.
Boolean. Default: true

walkonly
Walk the data tree, describing the hierarchy, but not querying, reporting, or logging.
Boolean. Default: false

topdir The top of the data hierarchy.
Default: /usr/local/thresh

name The DNS name or partial name of the device or hierarchy represented by the current directory in the data hierarchy. Gets extended by the name of each directory as **thresh** descends down the hierarchy, but can be overridden in a *DEFAULTS* file.
Default:

baseignore
The base list of ‘‘glob’’ patterns of file and directory names to ignore in the data hierarchy. If you override this variable, make sure that you end up ignoring the *.thresh* status directories.
Default: . .. .* CVS RCS README README.* DEFAULTS core *.core

ignore The extended list of ‘‘glob’’ patterns of file and directory names to ignore in the data hierarchy. Having two variables makes it easy to augment the default list of names to ignore without overriding the base list. Note that setting
ignore = *
will cause the hierarchy rooted at that location to be excluded from all processing.
Default:

prune If true, do not process further down this hierarchy if the current node is unreachable. This is essentially the equivalent of setting
ignore = *
if the current node is unreachable. This is useful, for example, for limiting the error messages that are generated if a gateway router is unreachable.
Boolean. Default: false

SNMP VARIABLES

community
: The SNMP V1 read community string to use to query hosts and devices.
Default: public

mib
: Specifies a file name that contains an SNMP MIB that will immediately be read and compiled into the running program.
Default:

timeout
: How long to wait for a response to an SNMP get request, in seconds.
Default: 10

retries
: Number of times to retransmit an SNMP get request during the timeout interval.
Default: 5

NOTIFICATION VARIABLES

notifier
: Pipe notification messages to this program, often a mailer or mail interface.
Default: /bin/cat

frequency
: Minimum time before sending another identical notification message, in minutes.
Default: 30

describe
: Whether or not to include a variable's MIB description field in notification messages.
Boolean. Default: true

msgformat
: A printf-style format string used to print notification messages, with the (cryptically named) variables smnpvar, message, complabel, compval, newlabel, newval, desc. This could use a little more sophistication.
Default: %s: %s\n %s %s\n %s %s%s\n

LOGGING VARIABLES

log
: Write out of spec entries to a log file.
Boolean. Default: true

logger
: A command like **logger**(1) that writes to **syslog**(3).
Default: /usr/bin/logger

syslog
: A syslog facility.level pair, as accepted by the **logger**(1) command, such as ''local1.info''. If set, out of spec entries are piped to the *logger* command.
Default:

syslogtag
: Tag to use on syslog'd entries.
Default: thresh

SEE ALSO

thresh(1)
threshdata(5)

AUTHOR

John Sellens