



Configuring Custom Script-based Monitoring

With Sentry-go Quick & Plus! monitors

© 3Ds (UK) Limited, November, 2013

<http://www.Sentry-go.com>

Be Proactive, Not Reactive!

The script monitoring component allows you to fully customise the checks made by Sentry-go and is aimed primarily at ...

- Advanced users who have programming or scripting knowledge
- Those that have a specific, or more unusual monitoring need, not covered by other monitoring options
- Those who have little or no programming or scripting knowledge but wish to perform more specific or unusual monitoring using the Sentry-go Scripting Wizard.

In this guide

System requirements	2
How it works	2
Configuring script monitoring	4
Defining a script, file or job to run	5
Setting script options	7
Running the script as another user	10
Determining success or failure	11
Scheduling a check	13
Temporarily ignoring a configured check	13
Configuring an automatic response	13
Configuring an alert	13
Web reporting with this monitoring component	13
The script status report	14
More Information	15

System requirements

This component is fully compatible with both Sentry-go Quick Monitors v6 and above, and Sentry-go Plus! v6 monitors and above. Access to VBScript and the Sentry-go Scripting Wizard is required.

How it works

This monitoring component works by managing one or more monitoring scripts, or jobs. These are periodically run automatically, based on the configured schedule and the output analysed by Sentry-go. The success or failure of the job is then determined, based on this output.

Although called the Script monitoring component, a job can in fact be defined as ...

- A command line program
- A batch file
- A script, such as VBScript or Jscript
- An executable program, or .NET command-based application.

During execution, Sentry-go monitors the job, waiting for it to finish. A job finishes either when ...

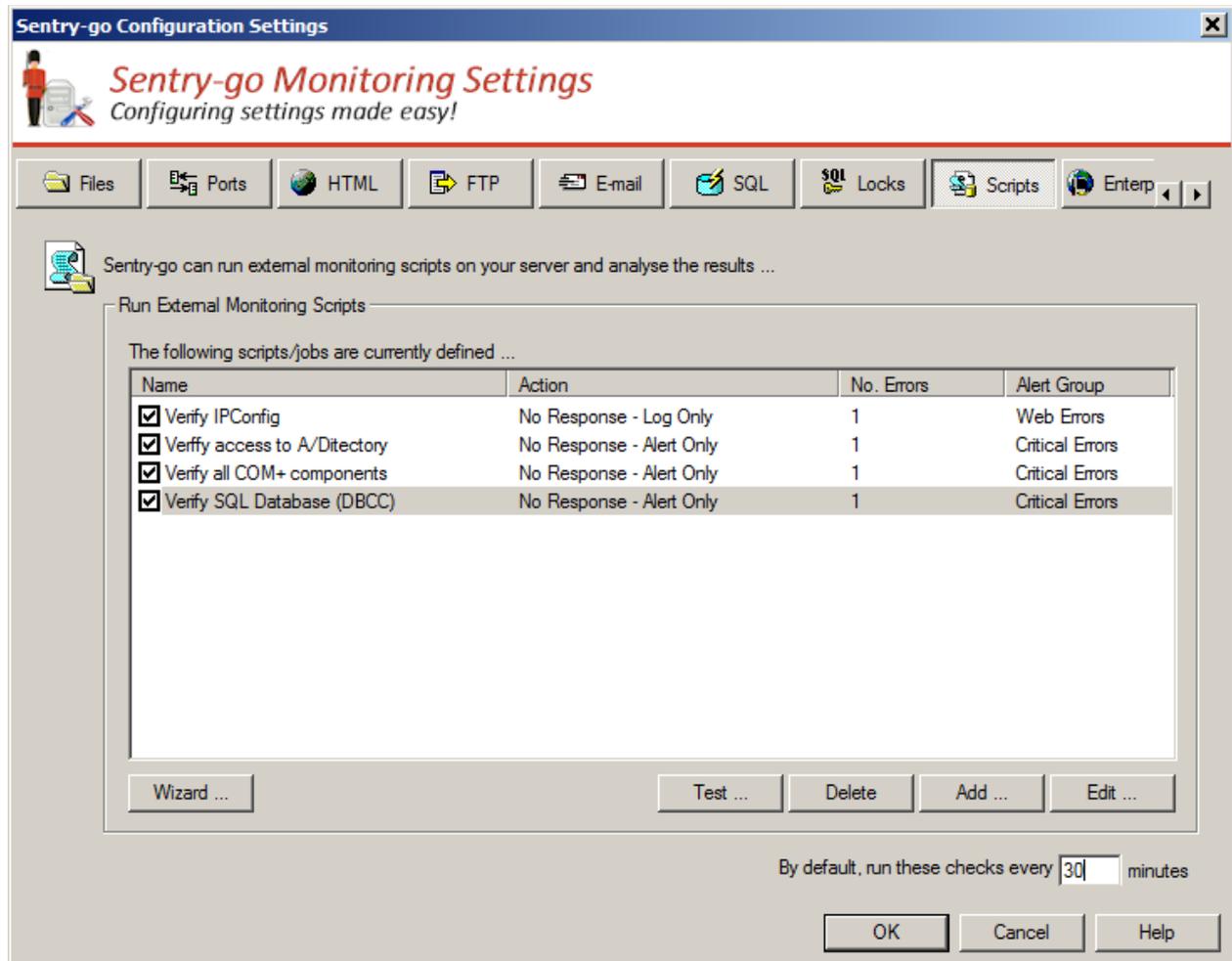
- It completes its last line of code
- It exits because of a defined condition
- It fails to start or because of an error (e.g. a compilation or runtime error)
- It times out (doesn't complete before the expected, configured period of time).

Once complete, the results are analysed, based on ...

- Either the numeric return code
- By analysing the output of the job
- By analysing the contents of a given file defined by the monitor and written to by the script.

Once the outcome is determined, the component can take action itself, repeat the check, or trigger an alert to one or more administrators.

To set up monitoring, configure the appropriate monitor and select the “Scripts” tab.



Wizard ...

Click this button to launch the Sentry-go Scripting Wizard. This in turn allows you to generate scripts using template files & without the need for you to write a line of code.

 For more information, please see the “Sentry-go Scripting Wizard” guide.

Test ...

Click this button to run the selected script & display the results in a web page.

 The script will be run by the monitor, which must itself be running and accept web connections from the client.

The script will be run but no alerts will be triggered even if the script fails or the result indicates an error.

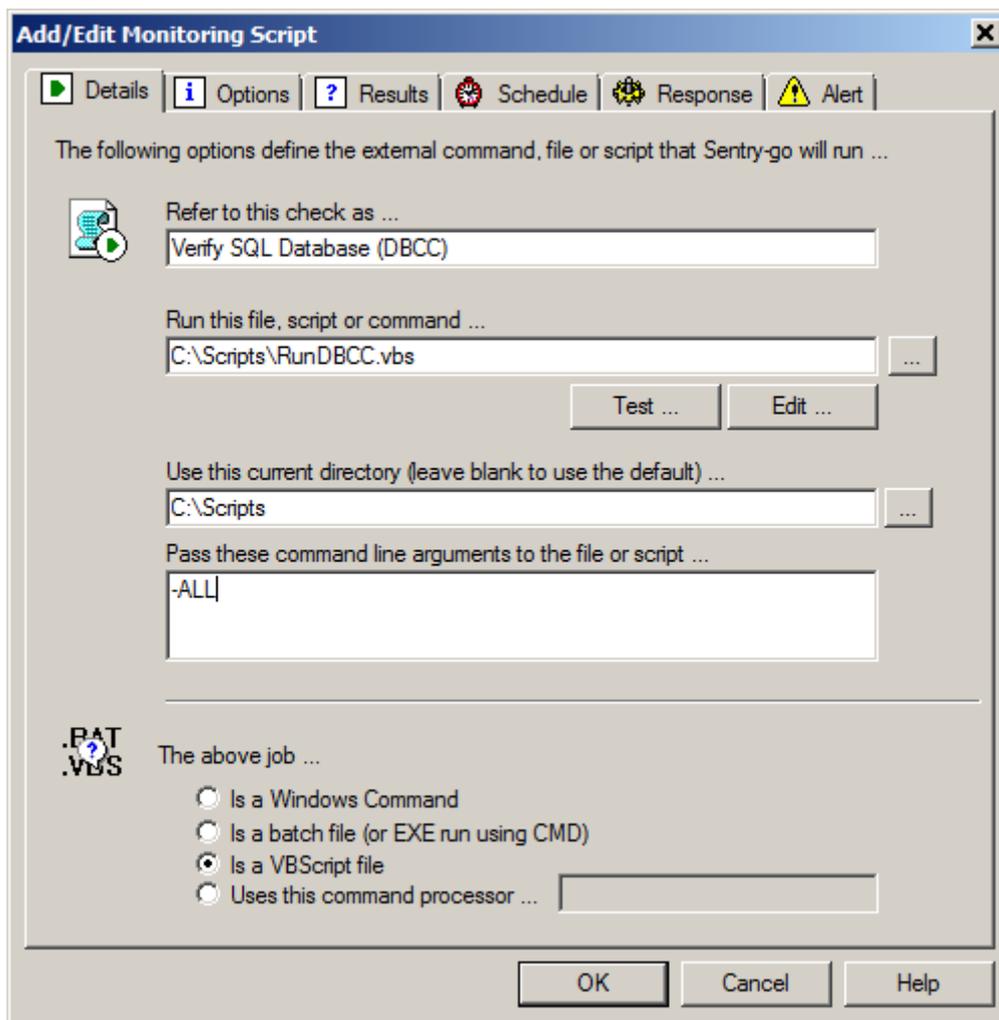
By default, run these checks every (minutes)

This value specifies how often, in minutes, Sentry-go, by default will run each script. To disable all default checks, set this value to 0.

This interval is only used when the interval is set to "Default". You can also configure the monitor to run a particular script at a given time or interval – e.g. hourly, daily etc. To do this, edit the particular script job from the list or configure the interval when the script is defined/added to the list.

Configuring script monitoring

To define a new monitoring script or edit an existing one, select the Add or Edit option from the main window.



A series of tabs will be presented, similar to the window shown above, allowing you to define all aspects of the job, how it should be run and how Sentry-go will determine its success or failure.

Defining a script, file or job to run

The first tab allows you to define the script itself & controls how it should be run by the monitor.

Refer to this check as

This field allows you to define a user friendly name for the check. This name is used within alerts as well as on web-based reporting.

Run this file, script or command

Enter the full path and name of the file or script you wish to run. Bear in mind paths should be relative to the local server running Sentry-go.

For remote paths (not recommended), use a full UNC name (e.g. \\ServerName\ShareName\ScriptName) etc.

Click the “...” to select your script from the file system

Generate ...

Click this button to launch the Sentry-go Scripting Wizard, which allows a number of predefined scripts to be selected & customised to your individual needs without the need for coding.



For more information, please see the “Sentry-go Scripting Wizard” guide.

Edit ...

Once a filename has been entered above, click “Edit” to launch Notepad in order to view the script’s logic or edit it further. Only text-based files can be loaded into Notepad.

Use this current directory ...

When the external command, file or script runs, it may make assumptions about the current directory. To force the current directory to be a specific location, enter the path here or click the “...” to chose a folder from Windows.

If no value is entered, the monitor’s own current directory will be used when launching & executing the script.

Pass these command line arguments to the file or script

If your file or script accepts or requires one or more command-line parameters, enter them here. These will be passed, as command-line arguments to the script at runtime.



Typically each parameter will be separated with a comma & strings or long filenames should be enclosed in double quotes.

The above job (script type)

These options allows you to select the type of file or script you will be running which in turn determines how it will be launched by the monitor. You can select one of the following ...

- **Batch file (.bat).**

Batch files are run through the Windows command-line interpreter (CMD), as though they were run on the command line. No additional command interpreter is required.

- **VBScript (.vbs).**

These scripts are run using the Windows Scripting Host (CScript).

- **Executable or Command (.exe).**

Like batch files, these programs are run directly from the command line (CMD) as an executable. Standard command-line commands such as "PING" are also, in effect, .EXEs. No additional command interpreter is required.

- **Other.**

For other types of file, or where you wish to use an alternative command processor, select this option & enter the appropriate command in the field provided. The command entered will be run, and the script passed to it, along with any command-line arguments..



Regardless of the type selected, the script monitor will generated & run the command line using a Windows Command Shell (CMD). If a command processor (or CScript) is used, this will be called from the command shell itself, thus allowing the monitor to access output information more readily.

The above selection will, however, be used to determine how the command line is generated and in particular, which command is initiated first.

Command Processor

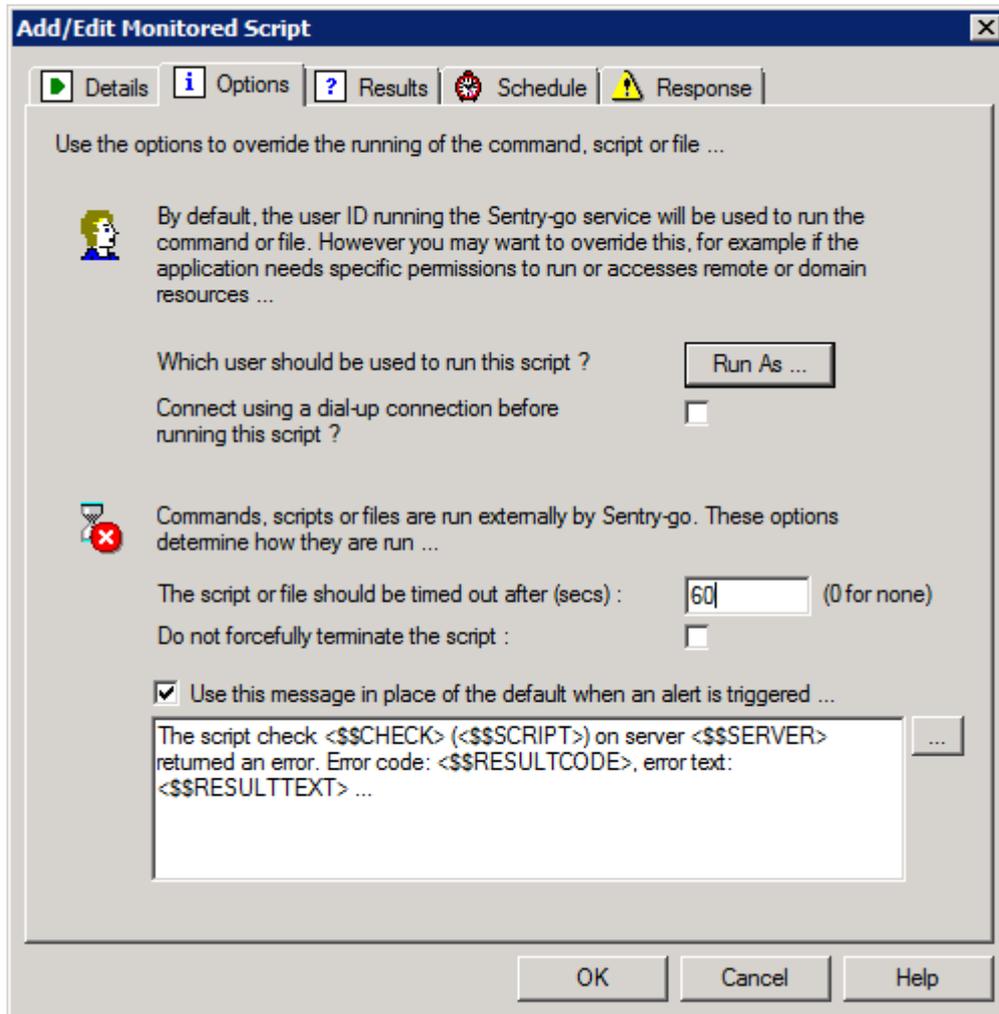
If you selected "other" above, this is the name of the command processor (if any) you wish to use.



The command processor entered must be available on the server being configured. If it isn't, the job may fail with unpredictable results.

Setting script options

The second tab allows you to control how the script will be run and how Sentry-go will monitor it whilst it is running.



Run As ...

By default the file or script will be run using the same user account that is running the monitor service. However, if the job requires specific permissions, or access to resources only available to a given domain user for example, you can specify the user here. For more information, see "Running the script as another user" below.

Connect using dial-up ...

Tick this option if your server isn't permanently connected to the network & you want Sentry-go to use dial-up to connect prior to running the script. You may wish to do this if the script accesses remote resources for example, that are only available once a dial-up connection has been established.

For more information, please see the "Sentry-go Dial-up Networking" guide.

The script or job should time out after ...

This option allows you to control how long the monitor should wait before a script that is yet to complete is forcefully timed out. If set to 0, no timeout will occur and the monitor will wait indefinitely. If set to a value, the value here determines the maximum wait time. If the job is timed out, it is considered to have failed (and will fail with a timeout error).

Do not forcefully terminate the script (not recommended) ...

Tick this option to instruct the monitor not to forcefully attempt to terminate the script even if it is timed out. In this case, the script will simply fail with a timeout error and be forgotten, even though it remains running.



Use this option with care. This is useful if you want to ensure the script doesn't leave any data or resources in an indeterminate state, however it should be used with caution as a timeout may also indicate a failure within the called script or logic. The script will continue to be called and so repeated failures could cause server resources to be depleted.

Even if the called job is terminated, the underlying command processor (e.g. the CScript job) may not. Timeout errors should always be investigated at the earliest opportunity & rectified as soon as possible.

Use this error message ...

By default, when un-ticked, a default error message will be used when a script either fails, or returns a failure. By ticking this option, you can override this default text for the file or script being defined and use your own message instead. The message text is entered in the field below.

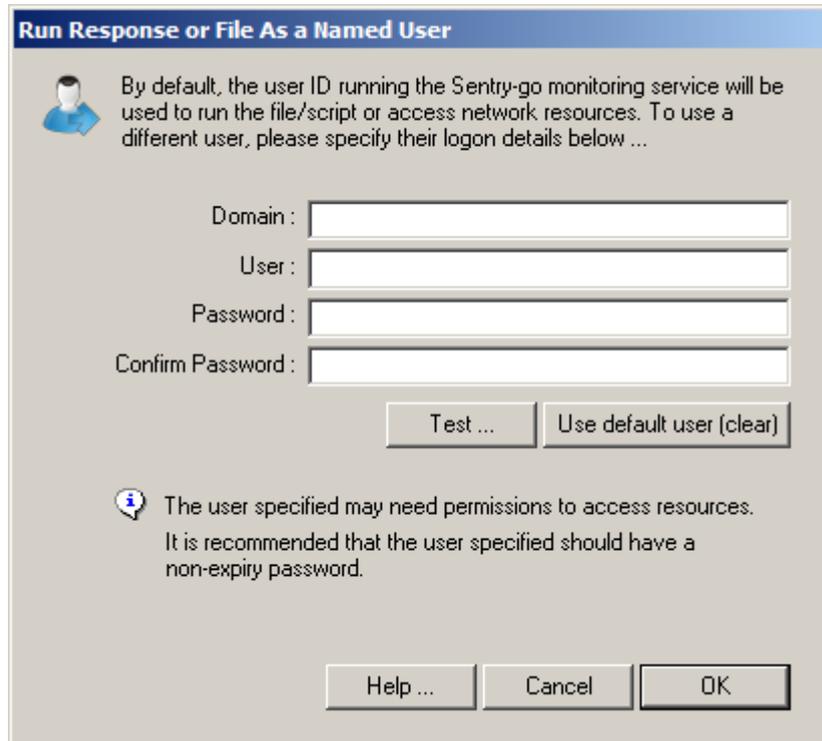
For help in defining this message text, click the "Build ..." button to launch the message builder. See the "Sentry-go Place-markers" guide for more information.

In addition to normal text, the following place-markers can also be included within the message. These are resolved at runtime and allow you to include your own error-specific information ...

This place-marker ...	Resolves to ...
<code><\$\$SERVER></code>	The server on which the script is running and the alert was triggered from.
<code><\$\$SCRIPT></code>	The full path & name of the script being run.
<code><\$\$RESULTCODE></code>	The numeric result returned from the script, or 0 if no result was returned.
<code><\$\$RESULTTEXT></code>	All result text output by the script (merged into a single string), or blank if no text or output was returned.
<code><\$\$RESULTERROR></code>	Where possible, the line that contained the error text, or if not, the complete text output, or blank if no text or output was returned ... <ul style="list-style-type: none">• If only one keyword indicates an error, the line containing the first keyword found will be returned.• If all keywords on the same line indicate an error, the first line containing all keywords will be returned.• Otherwise, the complete text output (merged into a single string) will be returned.
<code><\$\$CHECK></code> or <code><\$\$TEST></code>	Both these markers resolve to the name of the test being performed – the name as defined to Sentry-go.

Running the script as another user

By default, the file or script will be run as the user running the Sentry-go monitoring service. Typically this will be the local system account which has administrative privileges to the local machine only. If you wish to run the file under a specific account – e.g. a domain account with specific permissions or an account which has access to network resources etc, you can specify a named user using the “Run As ...” button ...



Run Response or File As a Named User

By default, the user ID running the Sentry-go monitoring service will be used to run the file/script or access network resources. To use a different user, please specify their logon details below ...

Domain :

User :

Password :

Confirm Password :

 The user specified may need permissions to access resources. It is recommended that the user specified should have a non-expiry password.

 To run the script under the same account as the user running the Sentry-go monitoring service (the default behaviour), simply ensure no entries are made – click the “Use default user (clear)” button.

 In order to ensure the script is run correctly, the entered user must ...

- Have “logon as a service” permission.
- Permissions to access any required resources.
- Have a non-expiry password (recommended).

To check that the user/password combination is valid from the monitoring server click the “Test” button. When selected, the Client Console connects to the target monitoring server (the server being configured) in order to verify the logon details & attempts to load their profile. The results are also displayed in the resulting web page.

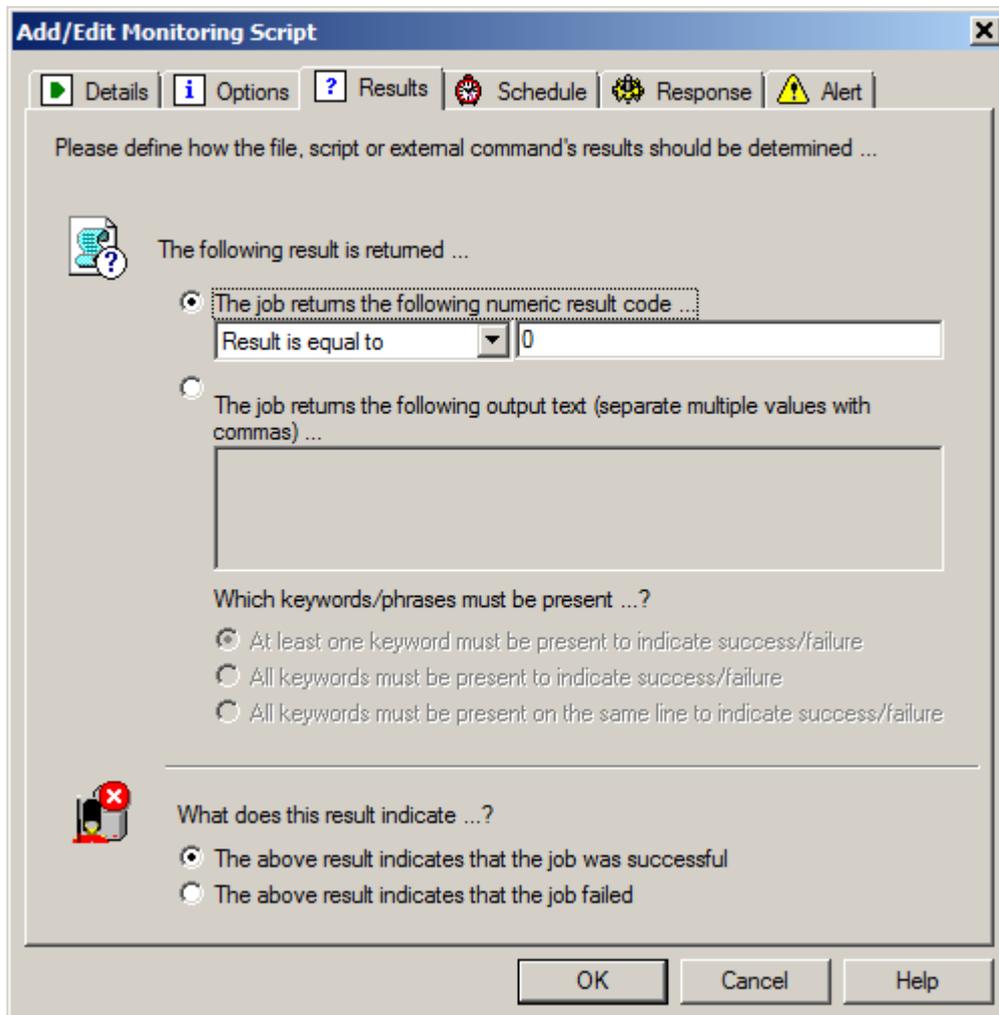
 In order to check the configuration, the target Sentry-go monitor must be running with web reports enabled.

The script itself is not run, only logon details are verified at this stage.

For more information on the Sentry-go log file, please see the “Configuring Logging Options” guide.

Determining success or failure

Lastly, Sentry-go needs to know how to interpret the results returned from the script. In particular, you define how the script will return a result and the meaning of these results.



The screenshot shows the 'Add/Edit Monitoring Script' dialog box with the 'Results' tab selected. The dialog has a title bar with a close button and a menu bar with 'Details', 'Options', 'Results', 'Schedule', 'Response', and 'Alert'. The main content area is titled 'Please define how the file, script or external command's results should be determined ...'. It contains three sections: 1. 'The following result is returned ...' with a radio button selected for 'The job returns the following numeric result code ...'. Below this is a dropdown menu set to 'Result is equal to' and a text input field containing '0'. 2. 'Which keywords/phrases must be present ...?' with three radio buttons: 'At least one keyword must be present to indicate success/failure' (selected), 'All keywords must be present to indicate success/failure', and 'All keywords must be present on the same line to indicate success/failure'. 3. 'What does this result indicate ...?' with two radio buttons: 'The above result indicates that the job was successful' (selected) and 'The above result indicates that the job failed'. At the bottom are 'OK', 'Cancel', and 'Help' buttons.

The following result is returned

The job or script can return a result either as a numeric return code, or via its output ...

- **The job returns the following return code**

If the script returns a numeric value, select this option. You can then use the associated fields to indicate how the results should be interpreted. In most cases, a return code of 0 indicates success whereas any other code indicates failure.

- **The job returns the following output text**

In some cases, you may wish to use the output from the script instead of the numeric return code. This is also true for command-line tools which cannot or do not pass back a return code. In this case, you can automatically analyse the results that ordinarily would have been displayed as output.

During execution, output is directed to a temporary file. If this option has been specified, the monitor will compare the keywords/phrases entered here against those found in this output file. This will then be used to determine whether the job was successful or not. Enter one or more text strings, separated by commas.

With this option, the following options are used to indicate how the keywords must be present to indicate the given result (success or failure – see below) ...

- **At least one keyword must be present**

Select this option if finding any of the keywords or phrases present is enough to indicate success or failure of the script.

- **All keywords must be present**

Select this option if all keywords must have been output to indicate success or failure of the script. If multiple output lines are returned, the keywords can appear on any of them, but they must all be present.

- **All keywords must be present on the same line**

Select this option if all keywords must have been output on the same line to indicate success or failure of the script. If multiple output lines are returned, the keywords must all appear together on the same line.

These results indicate success/failure

The above options allow the monitor to search for results after the script or job has completed. Use this option to determine whether the values entered above indicate that the script worked successfully, or whether it indicates a failure.

- If set to success, the above value(s) indicate that script worked and no alert will be triggered. Any values not listed will indicate failure and cause an alert to be triggered.
- If set to failure, the above value(s) indicate that script failed and an alert will be triggered. Any values not listed will indicate that the script was successful.

Scheduling a check

By default, each check is performed periodically at regular intervals throughout the day. The frequency of these checks is determined by the value specified at the bottom of the main list.

However, there may be times when you wish to perform the check at a different time, maybe at a set time each day, or on certain days etc. To do this, select the “Schedule” tab.

For more information, please see the “Sentry-go Monitoring Schedule” guide.

Temporarily ignoring a configured check

In some cases, you may wish to exclude a check from monitoring without removing it permanently. To do this, simply remove the “tick” or check against the entry you wish to ignore in the main list.

Configuring an automatic response

In the event an error is detected, Sentry-go can be configured to optionally respond automatically - i.e. to take action itself. This might be, for example, to terminate an unauthorized application.

To configure this, select entry from the list and click Edit. On the resulting window, select the “Response” tab.



For more information on the options available as well as details on how to configure automatic responses, please see the “Configuring Automatic Responses” guide.

Configuring an alert

In the event an error is detected and either no automatic response is defined or the response doesn't resolve the fault, an alert will be triggered. Depending on the monitor's general settings, you can either notify one or more contacts individually, or specify the alert group you wish to inform.

To configure these options, select the entry from the list and click Edit. On the resulting window, select the “Alert” tab.



For more information, please see the “Configuring Sentry-go Alerts” guide.

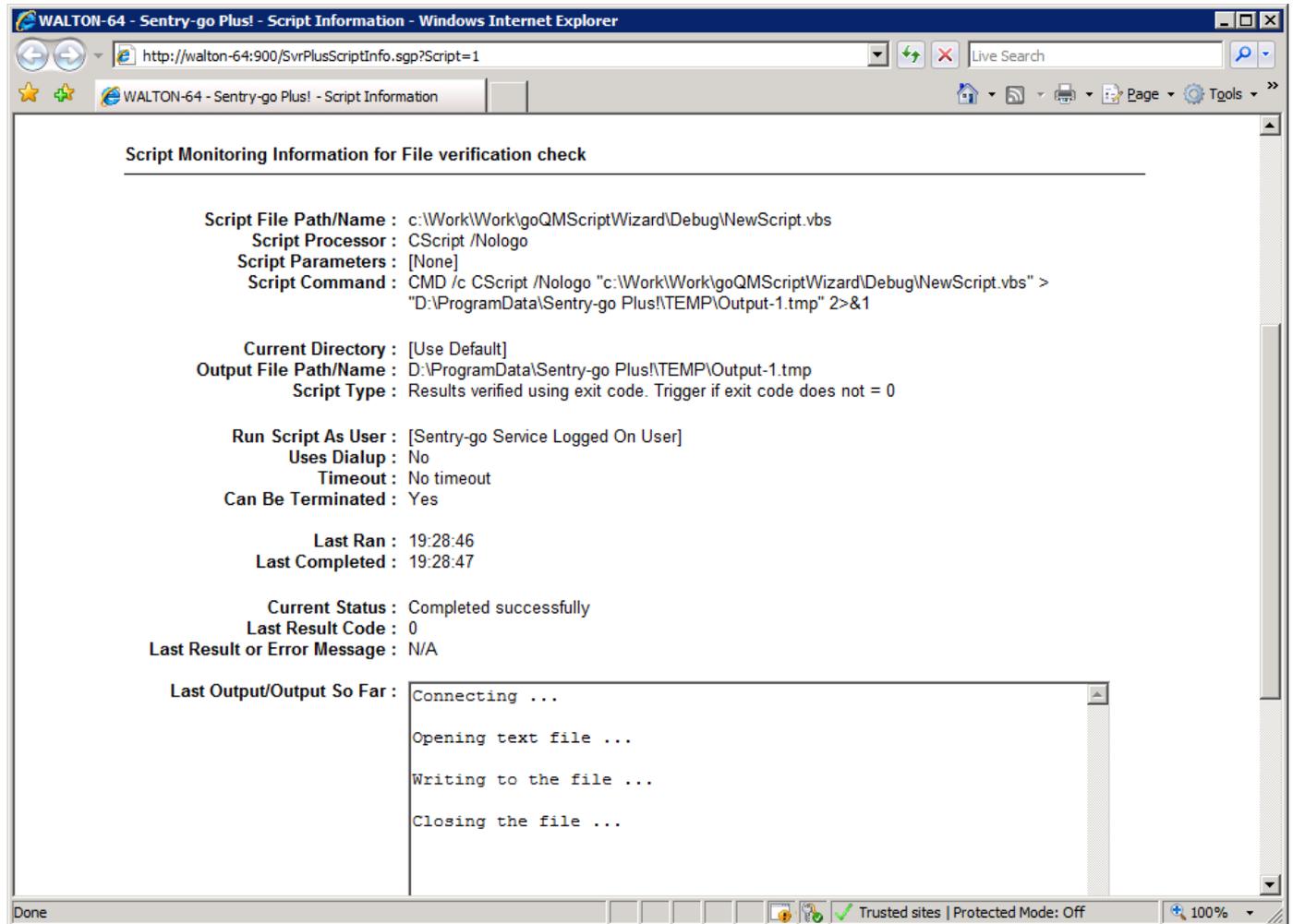
Web reporting with this monitoring component

In addition to the standard Sentry-go web reports, this component provides the following additional reports. These can be accessed directly from the URL, or from the monitor's home page.

The script status report

URL, accessed from : `http://<Server Name>:<Port>/SgoMntrStatus.sgp`

This report gives full details of the select script, the last time it ran and the results it generated. It is accessed by clicking the appropriate "Full Details" link within the Script-monitoring section of the Current Status report.



The screenshot shows a web browser window titled "WALTON-64 - Sentry-go Plus! - Script Information - Windows Internet Explorer". The address bar contains the URL `http://walton-64:900/SvrPlusScriptInfo.sgp?Script=1`. The page content is titled "Script Monitoring Information for File verification check" and displays the following details:

- Script File Path/Name :** `c:\Work\Work\goQMScriptWizard\Debug\NewScript.vbs`
- Script Processor :** `CScript /Nologo`
- Script Parameters :** `[None]`
- Script Command :** `CMD /c CScript /Nologo "c:\Work\Work\goQMScriptWizard\Debug\NewScript.vbs" > "D:\ProgramData\Sentry-go Plus\TEMP\Output-1.tmp" 2>&1`
- Current Directory :** `[Use Default]`
- Output File Path/Name :** `D:\ProgramData\Sentry-go Plus\TEMP\Output-1.tmp`
- Script Type :** `Results verified using exit code. Trigger if exit code does not = 0`
- Run Script As User :** `[Sentry-go Service Logged On User]`
- Uses Dialup :** `No`
- Timeout :** `No timeout`
- Can Be Terminated :** `Yes`
- Last Ran :** `19:28:46`
- Last Completed :** `19:28:47`
- Current Status :** `Completed successfully`
- Last Result Code :** `0`
- Last Result or Error Message :** `N/A`
- Last Output/Output So Far :**

```
Connecting ...
Opening text file ...
Writing to the file ...
Closing the file ...
```

The browser status bar at the bottom shows "Done", "Trusted sites | Protected Mode: Off", and "100%" zoom.

More Information

If you need more help or information on this topic ...

- Read all [papers/documents on-line](#).
- Watch [demonstrations & walkthrough videos on-line](#).
- Visit <http://www.Sentry-go.com>.
- Contact our [Support Team](#).



*Sentry-go, © 3Ds (UK) Limited, 2000-2013
East Molesey, Surrey. United Kingdom
T. 0208 144 4141
W. <http://www.Sentry-go.com>*