

Advanced Monitors

Overview

In some cases, the standard Uptime Infrastructure Monitor service monitors may not fully enable you to monitor all of the systems, applications, and proprietary devices in your environment; in some cases, you may need to capture unique metrics. To do this, you can configure advanced service monitors, or download and install plugin monitors.

These advanced monitors can be simple scripts that run service checks on a host. You can write a shell script, or use a higher-level scripting language like Perl, Python, or Ruby. Or, the advanced monitors can be binary programs that interact with more sophisticated applications. On top of that, advanced monitors do not require an agent to be installed on the system that you are monitoring.

Regardless of how you develop your advanced monitor scripts or programs, those scripts or programs should return the following codes:

- *0* - OK
The services are functioning properly.
- *1* - Warning
There is a potential problem with one or more of the monitored services.
- *2* - Critical
There is a critical problem with one or more of the monitored services.
- *3* - Unknown
There is an error in the configuration of the monitor itself, or Uptime Infrastructure Monitor cannot execute the service check.

Uptime Infrastructure Monitor captures the output from the script or program, usually from standard output (*stdout*). The output appears in the service status section of the *Global Scan* dashboard (see [Understanding the Status of Services](#)). The Uptime Infrastructure Monitor monitoring framework picks up any error codes and triggers the appropriate monitoring action.

If you have already written scripts or programs for other monitoring tools, you can re-use those scripts or programs with Uptime Infrastructure Monitor. You simply point your advanced monitor to where your scripts or programs are located and Uptime Infrastructure Monitor runs them.

The *uptime* user account on the Uptime Infrastructure Monitor Monitoring Station must be able to execute the script or program that you use.

Contact IDERA customer support Services for help with creating custom monitor scripts.

Before You Begin

When creating a script or an executable for an advanced monitor, you should ensure that:

- the necessary interpreter for the scripting language that you are using is installed on the Monitoring Station
- you have determined the arguments that the script or program requires, and the parameters that you want your script or program to return
- you use forward slashes when specifying directory paths in your scripts, regardless of the operating system (e.g., *C:/* on Windows, or */opt* on Solaris or Linux)

Many of the fields that you use to define an advanced monitor are the same as those used with agent and agentless monitors. You can find more information about those fields in the following sections.

- To learn how to access the custom monitor definition window, see [Using Agentless Monitors](#).
- For a description of monitor identification information fields, see [Monitor Identification](#).
- For a description of monitor timing settings, see [Monitor Timing Settings](#).
- For a description of alert settings, see [Monitor Alert Settings](#).
- For a description of Alert Profiles, see [Alert Profiles](#).
- For a description of Action Profile, see [Action Profiles](#).

Custom Monitors

A Custom monitor runs a script that captures information which is related to a situation that may be unique to your environment. When the script is run, the monitored system returns a single line of information to standard output (*stdout*). The script reads *stdout*, which may contain an error or return value. This error or return value is then displayed in the Uptime Infrastructure Monitor Monitoring Station.

As well, you can specify that the monitor writes the data that the script returns to the Uptime Infrastructure Monitor DataStore. You can use the retained data to later generate a Service Metrics report (see [Service Monitor Metrics Report](#)) or a Service Metrics graph.

Configuring Custom Monitors

To configure Custom monitors, do the following:

1. In the Custom monitor template, complete the monitor information fields.
To learn about monitor information fields, see [Monitor Identification](#).
2. Complete following fields:
 - Script Name
The name of, and path to, the script or program on the Monitoring Station that collects metrics.

The *uptime* user account on the Uptime Infrastructure Monitor Monitoring Station must be able to execute the script or program that you use. Ensure that the permissions for the *uptime* user account are set correctly.

- Arguments (Optional)
Specify any arguments that are required by the script or program.
 - Output (Optional)
Specify a comparison method to override the settings of an Alert Profile, or to return only the most severe errors. Do this by selecting an option from the *Comparison Method* dropdown lists beside the *Warning* and *Critical* fields. Then, enter a value in the field. For example, to return only unknown errors you can select *Exactly Matches* from the dropdown list, and type *UNKNOWN* in the field.
For more information on comparison methods, see [Comparison Methods](#).
 - Response Time
Optionally, enter the Warning and Critical Response Time thresholds. For more information, see [Configuring Warning and Critical Thresholds](#).
3. Click the *Save for Graphing* option to save the output in the DataStore. You can later use the retained data to generate a report or a graph.
 4. Complete the following settings:
 - Timing Settings (see [Adding Monitor Timing Settings Information](#) for more information)
 - Alert Settings (see [Monitor Alert Settings](#) for more information)
 - Monitoring Period settings (see [Monitor Timing Settings](#) for more information)
 - Alert Profile settings (see [Alert Profiles](#) for more information)
 - Action Profile settings (see [Action Profiles](#) for more information)
 - Click *Finish*.

Custom with Retained Data

Custom monitors with Retained Data return the following information:

- up to 10 values that you can save and evaluate
- a return status of 0 to 3 (see [Overview](#) for more information)

You can specify whether the monitor writes any returned data to the Uptime Infrastructure Monitor DataStore. You can use the retained data to later generate a Service Metrics report (see [Service Monitor Metrics Report](#)) or a Service Metrics graph (see [Viewing System and Service Information](#)).

You can also specify whether the monitor uses retained values as arguments in the custom monitor script. The values used are from the last instance the monitor was run. This most recent data sample can be used in cases where you would like to set alerts based on value comparisons.

When you wish to refer to retained values in a script, use the following environment variable as an argument:

```
%UPTIME_PREV_CUSTOM#%
```

Replace "*#*" with the ordered value the custom monitor is configured to save (i.e., 1 - 10).

For example, running a sample script called *application.exe* with the following arguments would make use of retained values for *Variable 1* and *Variable 5* as retained by the monitor:

```
%UPTIME_PREV_CUSTOM1% %UPTIME_PREV_CUSTOM5%
```

Configuring Custom Monitors with Retained Data

To configure Custom monitors with Retained Data, do the following:

1. In the Custom with Retained Data monitor template, complete the monitor information fields.
To learn how to configure monitor information fields, see [Monitor Identification](#).
2. Complete the following fields:
 - Script Name
The name of, and path to, the script or program on the Monitoring Station that collects metrics on the system.

The script or program that you specify must be executable by the *uptime* user account on the Uptime Infrastructure Monitor Monitoring Station. Ensure that the permissions are set correctly.
 - Arguments (Optional)
Specify any arguments required by the script or program.
 - Pass retained variables from previous collection as environment variables
Select this checkbox if you would like the monitor to be able to use the most recently collected variables as arguments in the script.
 - Variable 1 to Variable 10 (Optional)
Specify up to 10 variables that your custom script returns to the Uptime Infrastructure Monitor Monitoring Station. If you click the *Save for Graphing* checkbox, these variables are saved to the DataStore.
 - Response Time
Enter the Warning and Critical Response Time thresholds. For more information, see [Configuring Warning and Critical Thresholds](#).
3. Complete the following settings:
 - Timing Settings (see [Adding Monitor Timing Settings Information](#) for more information)
 - Alert Settings (see [Monitor Alert Settings](#) for more information)
 - Monitoring Period settings (see [Monitor Timing Settings](#) for more information)
 - Alert Profile settings (see [Alert Profiles](#) for more information)
 - Action Profile settings (see [Action Profiles](#) for more information)
4. Click *Finish*.

External Check

The External Check monitor captures asynchronous events. Uptime Infrastructure Monitor does not actively monitor these events by polling or initiating service checks; instead, External Check monitors rely on an external event to generate the information that the monitors capture. External Check monitors enable you to determine when to collect service data for the event that you specify.

After you define an External Check monitor, it has a status of UNKNOWN until it is updated with a URI similar to the following template:

```
http://$UPTIME_HOST:9996/command?
command=externalcheck&name=$ELEMENT_NAME&status=$STATUS&message=$MESSAGE&hostname=$MONITOR_NAME
```

`$UPTIME_HOST`: the hostname of the Uptime Infrastructure Monitor Monitoring Station

`$ELEMENT_NAME`: the hostname (not display name) of the system the External Check is assigned to

`$STATUS`: an integer indicating the status of the monitor: 0 = OK, 1 = WARN, 2 = CRIT

`$MESSAGE`: the message explaining the status

`$MONITOR_NAME`: the name of the External Check monitor

A sample script, `extevent.pl`, is included with Uptime Infrastructure Monitor in the `/scripts` subdirectory, and serves as an example of how to automate the receiving of asynchronous events, and update the External Check monitor accordingly. When the script is called with the appropriate arguments, it connects to Uptime Infrastructure Monitor command port (9996), and updates the status, triggering the appropriate Alert Profiles and Action Profiles.

Before using an External Check monitor, contact IDERA Customer Support for assistance. You need specific detailed instructions for configuring this monitor depending on the nature of the applications that generate asynchronous events for Uptime Infrastructure Monitor.

Configuring External Check Monitors

To configure External Check monitors, do the following:

1. In the External Check monitor template, complete the monitor information fields.
To learn how to configure monitor information fields, see [Monitor Identification](#).
2. Complete the following settings:
 - Timing Settings (see [Adding Monitor Timing Settings Information](#) for more information)
 - Alert Settings (see [Monitor Alert Settings](#) for more information)
 - Monitoring Period settings (see [Monitor Timing Settings](#) for more information)
 - Alert Profile settings (see [Alert Profiles](#) for more information)
 - Action Profile settings (see [Action Profiles](#) for more information)
3. Click *Finish*.

Plugin Monitors

Plugins are custom service monitors that are not part of the standard Uptime Infrastructure Monitor distribution, but can be integrated with it to augment the type of metrics collected. They are typically created by IDERA Client Support Engineers or Solutions Architects to address specific customer needs. Whether created by IDERA or other customers, plugins are hosted on [the Grid](#), the community repository for custom monitors, gadgets, dashboards, and other enhancements. Each plugin on the Grid includes the main distributable archive, any applicable auxiliary files, and essential installation steps.

Installing and maintaining plugins can also be managed from within Uptime Infrastructure Monitor. Users can browse, install, and update plugins using the Extension Manager if their User Role permits plugin management (see [Working with User Roles](#) for more information).

Users who are allowed to manage plugins can access this option by clicking the **Search for monitors** link at the top of the **Add Service Monitor** pop-up window (accessed by clicking **Services** > **Add Service Monitor**). Clicking this link opens the **Extension Manager**, and displays the following:

- **INSTALL**: the plugin is available on the Grid, but is not installed on your Uptime Infrastructure Monitor deployment
- **INSTALLED**: the plugin is installed, and is available to be configured and linked to monitored Elements
- **UPGRADE**: the plugin is installed, but there is an update available on the Grid

Installing and Upgrading Plugins

When you install a plugin through the Extension Manager, the plugin is downloaded from the Grid and installed automatically. A confirmation dialog indicates that the plugin is installed and is now available to be selected in the **Add Service Monitor** pop-up window. (The categories the plugin appears in depend on its service monitor definition; see [Plugin Guide](#) for more information.) The confirmation dialog also instructs you on follow-up steps, if applicable:

- requires additional steps: some manual installation steps need to be performed
- requires agent-side scripts: the plugin interfaces with an agent-side script, which needs to be installed on monitored Elements
- restart the Data Collector service: the plugin introduces changes that require the Uptime Infrastructure Monitor Core (or Data Collector service) to be restarted

When upgrading a plugin, typically, all the necessary follow-up steps are already performed.