

# Alerts and Actions

## Understanding Alerts

When a problem occurs at a Datacenter, Application, or SLA, the Monitoring Station can send *alerts* to users. Alerts are notifications that inform users who are configured to receive alerts of the problem. The notification message contains the following information:

- the type of notification - either *Problem* or *Recovery*
- the date and time when the problem occurred
- the name of the host on which the problem occurred
- the status of the host (see [Understanding the Status of Services](#) for more information)
- the name of the service that is experiencing the problem
- the current state of the service
- any output from the monitor

Whenever the status of an Element changes - for example from Critical to Warning - *up.time* sends an alert.

You can also configure *alert escalations* that occur if a warning is sent and is not acted upon. For example, if an alert is sent to a system administrator and the administrator does not attend to the problem within a specified amount of time, then the alert will be sent to the administrator's manager.

*up.time* can send alerts via:

- email messages to a cell phone or a pager, or to one or more email addresses
- a Windows popup

The following is a sample email alert:

*Notification type: Problem*

*1/12/2008 10:52*

*Host: filter*

*Host State: N/A*

*Service: FS Capacity - Filter*

*Service State: WARN/*

*Output: /var is 92% full*

The following is a sample pager alert:

*subject:*

*CRIT Alert*

*content:*

*5/7/2005 13:22*

*Type: Problem*

*Service: FTP (CRIT)*

*Host: filter (CRIT)*

For more information on alerts, see [Monitor Alert Settings](#).

## Understanding the Alert Flow

Alerts in *up.time* follow a specific flow. When *up.time* detects a problem with a host, it issues an alert. *up.time* then continues to check the host at specific intervals and reports on the status of the host.

Considering the following example:

- *up.time* checks the host system every 15 minutes
- alerts are sent continually every check interval until *up.time* detects a change in the state of the host system
- whenever an error is encountered, *up.time* rechecks the system every minute
- if all rechecks up to the maximum number of rechecks fails, *up.time* issues an alert

*up.time* encounters a critical error on a host. *up.time* performs three rechecks at one minute intervals - all of which return a critical error - and then sends an alert after the third recheck.

*up.time* then checks the host every two hours. While *up.time* encounters two critical errors, it does not send an alert. Then, the status of the host changes from critical to warning. When this change is detected, *up.time* sends an alert informing recipients of the change in status. When the status of the host changes to OK, *up.time* issues an alert informing recipients that the host has recovered.

This alert flow is illustrated in the following diagram:

## Alert Profiles

Alert Profiles are templates that tell *up.time* how to react to various alerts that are generated by service checks. Alert Profiles enable *up.time* to execute a series of actions in response to the failure of a service check or when a threshold is exceeded. The following diagram illustrates how an Alert Profile works:

An Alert Profile can send an alert via email, or to a pager or a cell phone, or a Windows popup alert. You can configure any or all of these actions to occur simultaneously. For example, if a Web server process stops responding, the system administrator can be notified.

## Enabling the Windows Messaging Service

In order to receive popup alerts from *up.time*, the Windows messaging service must be enabled on the recipient's computer.

To enable the Windows messaging service, do the following:

1. In Windows, select *Start > Control Panel*.
2. In the *Control Panel*, double click *Administrative Tools*, and then double click *Services*. The *Services* window appears.
3. Find and then double click *Messenger* in the list of services. The *Messenger Properties* dialog box appears.
4. In the *Messenger Properties* dialog box, select *Automatic* from the *Startup type* dropdown list.
5. Click *Apply*.

## Creating Alert Profiles

To create Alert Profiles, do the following:

1. On the *up.time* tool bar, click *Services*.
2. In the *Tree* panel, click *Add Alert Profile*. The *Add Alert Profile* window appears.
3. Type a descriptive name for the profile in the *Name of Alert Profile* field.
4. In the *Start alerting on notification number* field, enter the number of times an error must occur before *up.time* sends an alert notification.
5. Enter the number of times to re-send the notification in the *End alerting on notification number* field. Optionally, click the *Never Stop Notifying* option to have *up.time* continually send notifications.
6. Select one of the following notification options:
  - Email Alert
    - Sends the alert to the email addresses of the members of a Notification Group.
  - Pager Alert
    - Sends the alert to the pagers of the members of a Notification Group.
  - Script Alert
    - Uses a script to send the alert via SMS to the mobile phones of the members of a Notification Group.
    - Since this alert option relies on a script or batch file, you must enter its name and path in the *Script Path* field (for example, `/usr/local/uptime/scripts/scriptAlert.sh`).
    - When the alert is triggered, *up.time* runs the script and passes the script or batch file a set of parameters. The script is run for each *up.time* user who will receive the SMS message.
    - For details on how to create the script, see the Client Care Web site Knowledge Base article "Creating Custom Alert Scripts in *up.time* Alert Profiles".
  - Windows Popup Alert
    - Sends the alert via the Windows messaging service to the desktops of the members of a Notification Group.
7. Select one or more groups that will receive the notifications from the Available Notification Groups list, and then click *Add*.
8. Click *Save*.

## Viewing Alert Profiles

To view Alert Profiles, do the following:

1. On the *up.time* tool bar, click *Services*.
2. In the *Tree* panel, click *View Alert Profiles*. The *Alert Profiles* subpanel appears. The subpanel displays the settings that you configured when you created the profile, as well as a list of the services that are attached to the profile.
3. To test whether or not the profile will send alerts, click the *Test Alert Profile* button. A popup window appears, and the alert is sent using the notification method - email, pager, script, or Windows popup - that is specified in the profile. The following is an example of an email alert:

```
Notification type: Problem 27/4/2006 09:19
Host: Test Host (OK)
Service: Test Monitor
Service State: OK
Output: This is a test notification; please ignore.
```

When the alert is sent, the message *Alert Profile Tested* appears in the popup window. If an error message appears in the popup window, edit the profile and test it again.

## Editing Alert Profiles

To edit Alert Profiles, do the following:

1. On the *up.time* tool bar, click Services.
2. In the *Tree* panel, click *View Alert Profiles*.
3. Click the *Edit Alert Profile* icon beside the name of the profile that you want to edit.  
The *Edit Alert Profile* window appears.
4. Edit the *Alert Profile* fields, as described in the section [Creating Alert Profiles](#).

## Associating Alert Profiles to Elements

You can associate an Alert Profile to any Service Monitor, Application, or SLA if their state changes from OK to Warning or Critical. Alert Profiles are normally associated with any of these monitored items at the time of their configuration; Alert Profile associations can also be modified with existing service monitor definitions.

See [Using Service Monitors](#), [Working with Applications](#), and [Adding and Editing SLA Definitions](#) for more information about configuring Service Monitors, Applications, and SLAs, respectively.

## Working with Custom Alert Formats

*up.time*'s standard alert format is well suited for most alerting needs. However, you can modify the content of the alert. *up.time* comes with three custom alert templates. You can change the content of the alert by adding or removing variables from the template.

To define a custom alert format, do the following:

1. Define an Alert Profile, as described in the section [Creating Alert Profiles](#).
2. In the *Custom Format Options* section, click *Custom Formats*.
3. From the dropdown list, select one of the following options:
  - **Small Template**  
Contains the date and time of the alert, as well as the names and status of the service and host for which the alert was generated. This corresponds to the template used for pager alerts.
  - **Medium Template**  
Contains the information in the small template, as well as an expanded subject line, the type of notification, and output from the service monitor. This corresponds to the template used for email alerts.
  - **Long Template**  
Contains the information in the medium template, as well as the status of the host.
4. Click *Fill*.  
The variables associated with the template appear in the subject and body fields.
5. Add or remove variables (see [Custom Alert Format Variables](#)) as needed. You can also add other information to the body of the alert, such as paths to custom scripts or the names of alternative contacts.
6. Click *Save*.

## Custom Alert Format Variables

The variables are the building blocks of a custom alert format. You can add or remove variables to suit your needs.

These alert variables are also available as input parameter values when configuring an Action Profile to initiate a VMware vCenter Orchestrator workflow.

The table below explains the variables available in custom alerts, as well as Orchestrator input parameters :

Variable	Definition
<b>\$DISPLAYNAME\$</b>	The name of the Element as it appears in the <i>up.time</i> Web interface.  A system can have a different display name than the hostname. For example, you can assign the display name <i>Toronto Mail Server</i> to a system with the host name <i>10.1.1.6</i> .
<b>\$DATETIME\$</b>	The date and time at which the alert was generated. This appears in the subject line of the message.
<b>\$SERVICENAME\$</b>	The name of the service, along with the name of the host for which the alert was generated.  For example, if the alert was generated by the ping check for the server MailHub, then <i>PING-MailHub</i> appears in the alert.  This appears in the subject line of the message.
<b>\$SERVICESTATE\$</b>	One of the following: <ul style="list-style-type: none"> <li>• <i>OK</i></li> <li>• <i>WARN</i></li> <li>• <i>CRIT</i></li> <li>• <i>MAINT</i></li> <li>• <i>UNKNOWN</i></li> </ul> This appears in the subject line of the message.
<b>\$DATE\$</b>	The date on which the alert was generated.
<b>\$TIME\$</b>	The time at which the alert was generated.
<b>\$HOSTNAME\$</b>	The name of the host (as saved in <i>up.time</i> ) for which this alert was generated.

<b>\$HOSTSTATES\$</b>	The status of the host, which can be one of the following: <ul style="list-style-type: none"> <li>• <i>OK</i></li> <li>• <i>WARN</i></li> <li>• <i>CRIT</i></li> <li>• <i>MAINT</i></li> <li>• <i>UNKNOWN</i></li> </ul>
<b>\$TYPES\$</b>	The type of notification, which can be one of the following: <ul style="list-style-type: none"> <li>• <i>Problem</i></li> <li>• <i>Recovery</i></li> </ul>
<b>\$OUTPUT\$</b>	The output of the monitor that generated the alert. For example, <i>Ping completed: 1 sent, 100.0% loss, 0.0ms average round trip time</i>

## Action Profiles

Action Profiles are templates that direct *up.time* when it encounters a problem on a monitored system. You can associate an Action Profile to any Service Monitor, Application, or SLA if their state changes from OK to Warning or Critical. Action Profiles are normally associated with any of these monitored Elements at the time of their configuration; Action Profile associations can also be changed when you are modifying existing service monitor definitions.

See [Using Service Monitors](#), [Working with Applications](#), and [Adding and Editing SLA Definitions](#) for more information about configuring Service Monitors, Applications, and SLAs, respectively.

Actions include one of the following tasks:

- write an entry to a log file
- run a recovery script that can reboot a non-responsive server; or restart an application, process, or service
- stop, start, or restart a Windows server
- initiate a VMware vCenter Orchestrator workflow
- send an SNMP trap to a specific trap host and trap community

As templates, Action Profiles can be reused for any number of Service Monitor configurations. This means you can create a series of them as standard actions used to respond to typical types of problems you may encounter, depending on what role a Service Monitor is playing (e.g., availability or performance).

## VMware vCenter Orchestrator Workflow Actions

If an administrator has integrated *up.time* with VMware vCenter Orchestrator (see [VMware vCenter Orchestrator Integration](#)), you can configure Action Profiles to initiate Orchestrator workflows.

Orchestrator is a VMware vCenter Server add-on that allows its administrators to create workflows that automate vCenter management tasks. These Orchestrator workflows are open ended: all vCenter actions are available for automation through the processing of parameters and runtime arguments. *up.time* Action Profiles can be configured to provide input parameters to specific workflows, thus integrating vCenter management with *up.time*'s monitoring and alerting capabilities.

For example, if *up.time* is monitoring memory, CPU, and hard disk use for a virtualized server, the passing of performance thresholds can trigger an Action Profile that, in turn, triggers an Orchestrator workflow that creates a new virtual machine to alleviate resource strain. In a converse example, if *up.time* is monitoring a virtualized server for long periods of inactivity, a triggered Action Profile can initiate an Orchestrator workflow that shuts down the instance to free up resources.

By tightly integrating *up.time*'s monitoring and alerting with VMware vCenter Orchestrator's automated virtual environment administration, you can accelerate your organization's reaction time with virtual systems management, and map established policies to automated actions.

When configuring Action Profiles, *up.time* communicates with Orchestrator and dynamically produces a list of all available workflows. (This includes any third-party workflow packages that have been installed on the Orchestrator server, including the *up.time* Orchestrator package.)

When a workflow is selected, and the *Get Parameters* button is clicked, the corresponding input parameter fields are dynamically displayed, allowing you to specify parameter values required to completely configure the workflow for execution should an *up.time* alert initiate it.

### Orchestrator Input Parameter Variables

When configuring a VMware vCenter Orchestrator workflow, you have at your disposal a set of *up.time*-specific variables that can be entered as parameter variables, and whose ensuing runtime values will be passed to the Orchestrator workflow during execution. The variables available to you are those that are used when creating a custom alert format. See [Custom Alert Format Variables](#) for information.

## SNMP Trap Actions

You can also configure an Action Profile to send an SNMP trap to a particular host. An SNMP trap is notification that is issued by a system that is running SNMP when a problem occurs. The host to which the SNMP trap is sent must be running an SNMP trap listener.

If you use SNMP traps, the trap message will be sent in the format specified by the *up.time* MIB. This MIB is found in the *scripts* directory. The uptime software enterprise OID is *.1.3.6.1.4.1.24216*.

## Creating Action Profiles

To create Action Profiles, do the following:

1. On the *up.time* tool bar, click Services.

2. In the Tree panel, click Add Action Profile.  
The Add Action Profile window appears.
  3. Enter a name for this profile in the Name of Action Profile field.
  4. Specify the number of times an error must occur before up.time sends a notification in the Start action on notification number field.
  5. Specify the number of times action will be carried out in the End action on notification number field.  
Optionally, select the Never Stop Notifying option to continually carry out the action in this profile until the problem is resolved.
  6. If VMware vCenter Orchestrator integration has been enabled, and you would like the Action Profile to drive an Orchestrator workflow, do the following:
    - a. In the Select Workflow field, input a workflow to configure.  
You can either scroll through and select the workflow from the drop-down list, or begin typing the workflow's name.
    - b. Click Get Parameters .  
up.time will retrieve information from the Orchestrator server and dynamically display configuration fields for the chosen workflow's input parameters.
    - c. Configure the input parameter fields for the workflow.  
For information on the specific configuration parameters available for the chosen workflow, consult the appropriate developer's documentation.
  7. If you would like the Action Profile to write to a log, in the Log File field, enter the name and path to a log file on the Monitoring Station to which error information will be written.
  8. If you would like the Action Profile to run a recovery script, in the Recovery Script field, enter the name and path to a script that will reboot a server, or restart an application, process, or service.  
The recovery script will also have the following information appended to it:
    - the date and time on which the error occurred
    - the type of error notification that was sent
    - the name of the host on which the error occurred
    - the state of the host
    - the name of the service that threw the error
    - the state of the service
    - the output that was generated by the error
 for example:  

```
"/usr/local/uptime/recover.sh" "24/12/2007 5:01:05" "Problem" "printserver" "null" "WinSrv-Print Spooler" "CRIT/threshold error" "servicestatus: Not Running does not match Running (Service 'Print Spooler' found, status: Not Running, took 12ms)"
```
- Note - You can also use the recovery script to file trouble tickets with a system like Remedy, or to interact with third party software packages.
9. If you are setting up an Action Profile for a Windows server, you can also leave the Windows Service as Agent , and complete the following fields:
    - Windows Host  
The name of the host on which the service is running.
    - Agent Port  
The port on which the up.time agent that is installed on the system is listening. The default is 9998 .
    - Use SSL  
Select this option if up.time will securely communicate with the host using SSL (Secure Sockets Layer).
    - Agent Password  
Enter the password that is required to access the agent that is running on the system that is being monitored. For information on setting the agent password, see the uptime software Knowledge Base article entitled, "*What is the password for the Windows agent?*"
    - Windows Service  
The display name of the specific Windows service to which the Action Profile will apply. The display name of a service appears in the **Na me** column of the **Services** Control Panel, or in the **Description** column of the Windows Task Manager **Services** tab.



The service display name must be entered verbatim, including spaces, otherwise it will not be correctly processed. Double-clicking a service name in the **Services** Control Panel opens a properties window where you can highlight and copy the service **Display name**.

- Action  
Select one of the following actions:
    - None
    - Start
    - Stop
    - Restart
10. If you are setting up an Action Profile for a Windows server that is using a WMI implementation, you can also select the Windows Service as WMI, and complete the following fields:
    - WMI Host:  
The name of the host on which the service is running.
    - Windows Domain:  
The Windows domain in which WMI has been implemented.
    - Username:  
The name of the account with access to WMI on the Windows domain.
    - Password:  
The password for the account with access to WMI on the windows domain.
    - Windows Service  
The display name of the specific Windows service to which the Action Profile will apply. The display name of a service appears in the **Na me** column of the **Services** Control Panel, or in the **Description** column of the Windows Task Manager **Services** tab.



The service display name must be entered verbatim, including spaces, otherwise it will not be correctly processed. Double-clicking a service name in the **Services** Control Panel opens a properties window where you can highlight and copy the service **Display name**.

- Action  
Select one of the following actions:
  - None
  - Start
  - Stop
  - Restart

- If you want to send SNMP traps to a particular host, complete the following fields:
  - SNMP Trap Host  
The name of the host that monitors SNMP traps.
  - SNMP Trap Port  
The port number on the trap host to which the SNMP trap is sent.
  - SNMP Trap Community  
The name which acts as a password for sending trap notifications to the trap host.
  - SNMP Trap OID (optional)  
The object identifier (OID) that identifies the SNMP trap - for example, .1.3.6.1.2.1.34.4.1.7.
- If Splunk integration has been enabled, and you would like the Action Profile to write to the Splunk log, complete the following fields:
  - Splunk Hostname  
The host name of the server on which Splunk is running.
  - Logging Port  
The port on which the Splunk server is listening for logging requests. This port is configured in Splunk, and you will need to contact the Splunk administrator for this information.
  - Click the Use SSL option to securely access the Splunk server using SSL.

For more information on Splunk integration, see [See Splunk Settings](#).
- Click Save.

## Viewing Action Profiles

To view Action Profiles, do the following:

- On the *up.time* tool bar, click *Services*.
- In the *Tree* panel, click *View Action Profiles*.  
The *Action Profiles* subpanel appears, displaying the settings that you configured when you created the profile, as well as a list of the services that are attached to the profile.
- To test whether or not the profile works, click the *Test Action Profile* button.  
A popup window appears, and the Monitoring Station tries to carry out the action defined in the profile. When the action is completed, the message *Action Profile tested* appears in the popup window.  
If an error message appears in the popup window, edit the profile and test it again.

## Editing Action Profiles

To edit Action Profiles, do the following:

- On the *up.time* tool bar, click *Services*.
- In the *Tree* panel, click *View Action Profiles*.
- Click the *Edit Action Profile* icon beside the name of the profile that you want to edit.  
The *Edit Action Profile* window appears.
- Edit the Action Profile fields as described in the section [Creating Action Profiles](#).

## Monitoring Periods

Monitoring Periods are the times over which a service monitor will be actively monitoring a host. The Monitoring Periods also apply to the times when *up.time* sends alerts

*up.time* comes with the following Monitoring Periods:

- 24x7

Monitoring is performed 24 hours a day, seven days a week.

- 9am to 5pm weekdays

Monitoring is performed from 9 a.m. to 5 p.m., Monday to Friday.

- Never

No monitoring is carried out.

You can add Monitoring Periods that suit your needs. For example, you can create a Monitoring Period called *Weekends* that only monitors a host from 12:00 a.m. on Saturday to 11:59 p.m. on Sunday.

## Adding Monitoring Periods

To add Monitoring Periods, do the following:

On the *up.time* tool bar, click *Services*.

- In the *Tree* panel, click *Add Monitoring Period*.  
The *Add Monitoring Periods* window appears.
- Type a name in the *Monitoring Period Name* field.
- In the *Definition* section, enter one or more time period expressions that combine to create a full Monitoring Period definition.  
See [Time Period Definitions](#) for information on the types of time period expressions that are valid in *up.time*.
- Click *Save*.

