



WhatsConfigured for WhatsUpGold

v15.0

User Guide



IPSWITCH

CHAPTER 1 Welcome to WhatsConfigured

Finding more information and updates	2
Sending feedback	2

CHAPTER 2 Installing and Licensing WhatsConfigured

WhatsConfigured System Requirements	3
Installing and Licensing WhatsConfigured	3
Activating WhatsConfigured licenses	3

CHAPTER 3 Accessing WhatsConfigured Features in WhatsUp Gold

Finding WhatsConfigured on the Configure menu	5
Finding WhatsConfigured on Device Properties	6
Finding the WhatsConfigured Task Log	7
Finding WhatsConfigured thresholds in Alert Center	8

CHAPTER 4 Using Configuration Tasks

About Tasks	9
About the WhatsConfigured Task Library	10
Configuring tasks	10
Configuring schedulable tasks	11
Configuring password tasks	16
Assigning a task to a device	21
Viewing Task results	22

CHAPTER 5 Using Task Scripts

About Task Scripts	25
Using the WhatsConfigured Task Script Library	25
About the backup startup config task script	26
About the backup running config task script	27
Configuring custom task scripts	27

CHAPTER 6 The WhatsConfigured Custom Script Language

About the WhatsConfigured Custom Script Language	28
--	----

CHAPTER 7 Using WhatsConfigured Comments

About WhatsConfigured comments.....	30
-------------------------------------	----

CHAPTER 8 Using WhatsConfigured Variables

About variables	31
Variable definitions.....	31
Accessing protocol settings.....	32
Using reserved WhatsConfigured variable names	33

CHAPTER 9 Using WhatsConfigured Commands

About commands.....	34
About basic WhatsConfigured command syntax	35
About strings and regular expressions in WhatsConfigured	35
Storing WhatsConfigured command output in the WhatsUp Gold database	37
Editing WhatsConfigured command output.....	37
Using WhatsConfigured commands with queries.....	38
About WhatsConfigured command layout.....	39
WhatsConfigured script variables affecting command execution	40
About WhatsConfigured command types.....	41
@login.....	41
@enable	42
Device commands	43
Low-level commands.....	44

CHAPTER 10 Script Examples

Example Scripts	47
-----------------------	----

CHAPTER 11 WCD3.0 Using Policies

About policies	49
About the Policy Library	50
Configuring a policy	51
Auditing a policy	52

CHAPTER 12 About Device Properties - Tasks

Using Device Properties - Tasks	54
Viewing a configuration file	55
Comparing two configuration files	55
Restoring a device to an archived configuration	55
Deleting a configuration file from a device.....	55
Running a scheduled task immediately	56
Removing tasks from a device	56

CHAPTER 13 WCDv3 Using Alert Center with WhatsConfigured

Assigning an Alert Center threshold to a task.....	57
--	----

CHAPTER 14 Using Archive Search

About Archive Search.....	59
Performing an archive search.....	59

CHAPTER 15 Using WhatsConfigured reports

About the Configuration Task Log	61
--	----

CHAPTER 16 Using the WhatsUp Services Controller

About the WhatsUp Services Controller.....	63
Stopping the WhatsConfigured Service	64
Starting the WhatsConfigured Service.....	64
Restarting the WhatsConfigured Service	65
Stopping the TFTP server.....	65
Starting the TFTP server	65
Restarting the TFTP server.....	66

Copyright notice

Welcome to WhatsConfigured

In This Chapter

Finding more information and updates.....	1
Sending feedback.....	2

WhatsConfigured enables effective management of one of the most critical assets on your network—device configurations. As a fully integrated plug-in for WhatsUp Gold, WhatsConfigured automates the key configuration and change management tasks required to maintain and control configuration files for networking devices, reducing the risk of network outages caused by misconfigured devices. Network managers can leverage this automated configuration to reduce the amount of time spent ensuring their network devices are configured correctly, freeing valuable time.

WhatsConfigured is built around an automated task execution engine that allows network managers to dynamically gather configuration data about their network devices through configuration tasks. These configuration tasks can be scheduled to run on a regular basis or can be manually ran as needed to upload, download, and backup configuration files, manage device credentials, and much more. WhatsConfigured comes with several pre-defined configuration tasks with the option to create custom tasks. Additionally, WhatsConfigured works with the WhatsUp Gold Alert Center and can alert you on the success or failure of a task, or when changes are detected on a device.

With support for heterogeneous networks, WhatsConfigured provides secure SNMP, SSH, Telnet or TFTP access, to download and store device configuration files in a secure repository, keeping them readily available for file compares and restoration on a device.

WhatsConfigured not only reduces the time and effort required to maintain device configurations and changes while providing increased security, compliance, and visibility, it also reduces the risk of costly network downtime.

Finding more information and updates

The following are information resources for WhatsConfigured. This information may be periodically updated and available on the *WhatsUp Gold web site* (<http://www.whatsupgold.com/wugtechsupport>).

- **Release Notes.** The release notes provide an overview of changes, known issues, and bug fixes for the current release. The release notes are available on the *WhatsUp Gold web site* (<http://www.whatsupgold.com/wugtechsupport>).
- **Application Help for the console and web interface.** The console and web help contain dialog assistance, general configuration information, and how-to's that explain how to use the features. The Table of Contents is organized by functional area, and can be accessed from the main menu or by clicking **Help** in the console, or the **?** icon in the web interface.
- **Additional WhatsUp Gold resources.** For a listing of current and previous guides and help available for WhatsUp Gold products, see the *WhatsUp Gold web site* (<http://www.whatsupgold.com/support/guides.aspx>).
- **WhatsUp Gold optional plug-ins.** You can extend the core features of WhatsUp Gold by installing plug-ins. For information on available plug-ins and to see release notes for each plug-in, see *WhatsUp Gold plug-ins documentation* (<http://www.whatsupgold.com/support/guides.aspx>).
- **Licensing Information.** Licensing and support information is available on the *MyIpswitch licensing portal* (<http://www.myipswitch.com/>). The web portal provides enhanced web-based capabilities to view and manage Ipswitch product licenses.
- **Technical Support.** Use the WhatsUp Gold Support Site for a variety of WhatsUp Gold product help resources. From here you can view product documentation, search Knowledge Base articles, access the community site for help from other users, and get other Technical Support information. The Support Site is available on the *WhatsUp Gold web site* (<http://www.whatsupgold.com/wugtechsupport>).

Sending feedback

We value your opinions on our products and welcome your feedback.

To provide feedback on existing features, suggest new features or enhancements, or suggest ways to make our products easier to use, please fill out our *product feedback form* (<http://www.whatsupgold.com/wugfeedback>).

CHAPTER 2

Installing and Licensing WhatsConfigured

In This Chapter

WhatsConfigured System Requirements.....	3
Installing and Licensing WhatsConfigured	3
Activating WhatsConfigured licenses	3

WhatsConfigured System Requirements

The minimum software and hardware requirements are based on the Ipswitch WhatsUp Gold requirements. For more information, see the *WhatsUp Gold web site* (<http://www.whatsupgold.com/WUG15relnotes>).

Installing and Licensing WhatsConfigured

The files for WhatsConfigured plug-in are installed automatically with Ipswitch WhatsUp Gold. Your license file determines whether or not you can access the WhatsConfigured plug-in, and how many devices you can assign to a WhatsConfigured task.

To update your license with a purchased WhatsConfigured plug-in, or to increase your device-per-task count, visit the *MyIpswitch portal* (<http://www.myipswitch.com>).

Activating WhatsConfigured licenses

If WhatsConfigured is installed using the installation application downloaded from the web link provided in the purchase confirmation email, the program is fully functional immediately after installation.

If the WhatsConfigured license is not automatically activated during installation, you can manually activate WhatsConfigured using the activation program in the WhatsUp Gold group on the Windows Start menu.

To activate WhatsConfigured manually:



Note: Before you begin the manual activation process, make sure that you have your product serial number available to use in the activation program.

- 1 Click **Start > Programs > Ipswitch WhatsUp Gold > Manage WhatsUp Gold License**. The activation program appears.
- 2 Follow the onscreen instructions to complete the product activation.



Note: When activation completes, a confirmation page indicates that the license has been activated. If activation does not complete successfully, you may be behind a proxy or firewall that is blocking the activation request. In this case, click **Offline** and follow the onscreen instructions.

For additional help and information about managing your product license, go to the *MyIpswitch licensing portal* (<http://www.myipswitch.com/>).

CHAPTER 3

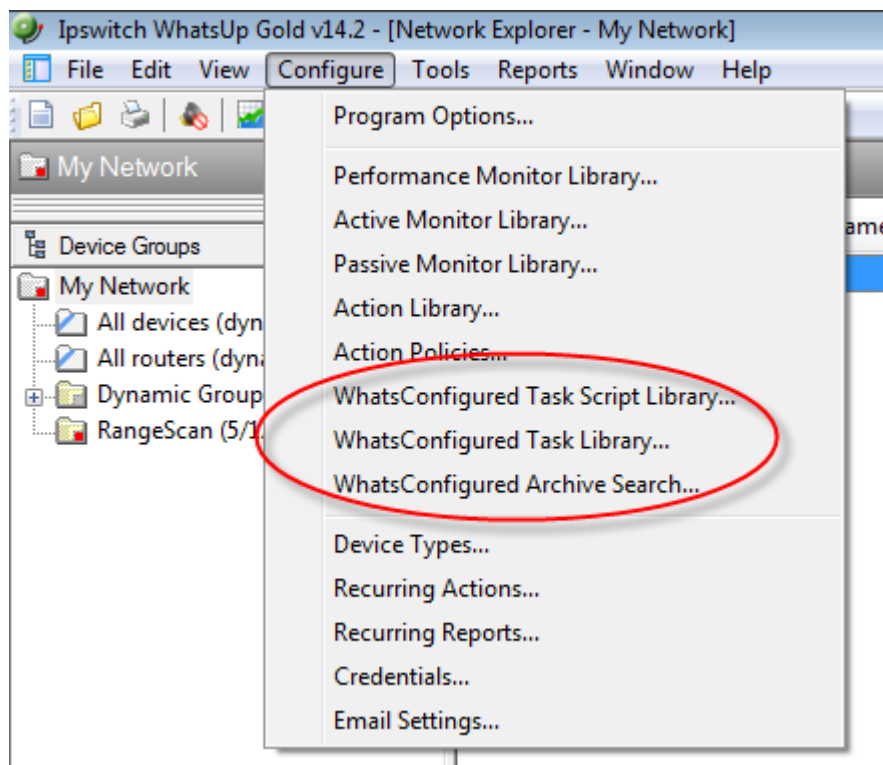
Accessing WhatsConfigured Features in WhatsUp Gold

In This Chapter

Finding WhatsConfigured on the Configure menu	5
Finding WhatsConfigured on Device Properties	6
Finding the WhatsConfigured Task Log.....	7
Finding WhatsConfigured thresholds in Alert Center.....	8

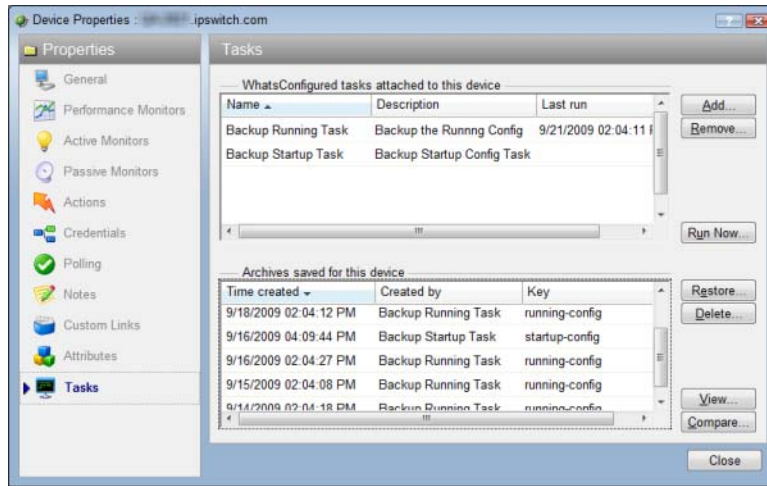
Finding WhatsConfigured on the Configure menu

You can access WhatsConfigured from the WhatsUp Gold console's main menu at **Configure > WhatsConfigured Task Library** or **Configure > WhatsConfigured Task Script Library**.



Finding WhatsConfigured on Device Properties

WhatsConfigured can be found on the Tasks section of the WhatsUp Gold Device Properties dialog.



To access WhatsConfigured through Device Properties:

- 1 In the Device List or Map View, right-click a device, then click **Properties**. The Device Properties dialog appears.
- 2 Click **Tasks**. The Tasks section of Device Properties appears.

Finding the WhatsConfigured Task Log

You can access the WhatsConfigured Task Log from the WhatsUp Gold web interface's Reports tab. On the Overview page, the WhatsConfigured Task Log is listed as a **Quick Link**.



From the All Reports page, you can find the WhatsConfigured Task Log under the **General** category.

Report name	Description
System	View logs and diagnostic data for all devices
Group	Compare availability and performance data for devices within a selected group
Device	Focus on availability and performance data for a selected device
Performance	Focus on performance data for a selected device or group
Problem Areas	View alerts reported across the entire network and troubleshoot problems
General	View workspaces and WhatsUp Gold application logs
Activity Log	A history of system-wide configuration and application initialization messages generated by WhatsUp.
Home Workspace	Your home workspace.
Recurring Action Log	Results of Recurring Actions.
Recurring Report Log	Results of Recurring Report executions.
Web User Activity Log	Shows the history of user activity on the system.
Actions Applied	Shows how actions are applied to devices and monitors in a group.
Quarterly Availability Summary	Shows the availability summary for a group.
State Summary	A summary of device states organized by device group.
Top 10	A collection of Top 10 reports.
WhatsConfigured Task Log	A record of all log messages generated by WhatsConfigured filterable by device and task.
Device Status	A detailed look at a specific device.
Favorites	

Finding WhatsConfigured thresholds in Alert Center

If you assign an Alert Center threshold to a WhatsConfigured task, a custom threshold workspace report for the WhatsConfigured task threshold is displayed on the Alert Center tab.

For more information, see *Using Alert Center with WhatsConfigured* (on page 57).

CHAPTER 4

Using Configuration Tasks

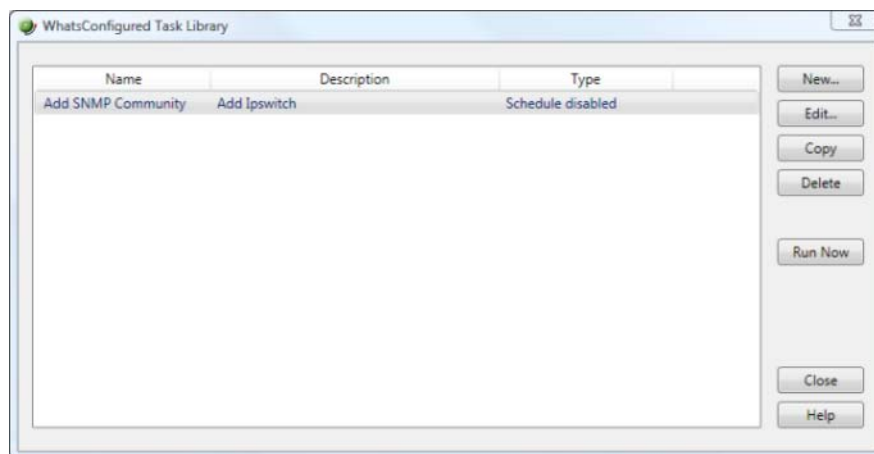
In This Chapter

About Tasks	9
About the WhatsConfigured Task Library	10
Configuring tasks.....	10
Assigning a task to a device.....	21
Viewing Task results	22

About Tasks

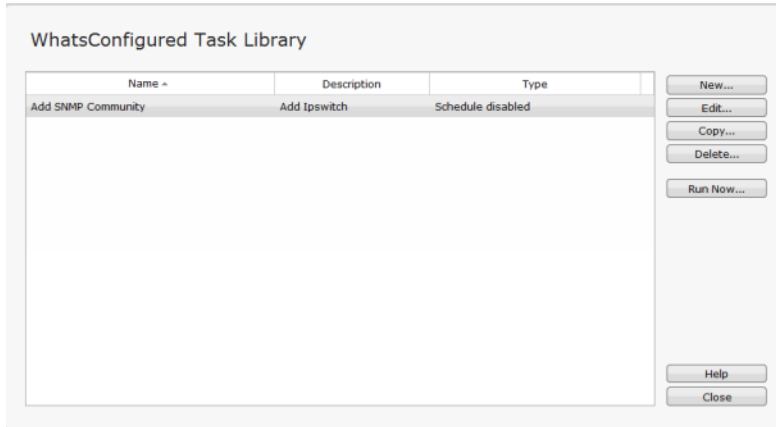
Task scripts are powered by user-configured *tasks*. When you configure a WhatsConfigured task, you select the specific task script that you want the task to execute at the time it is run.

Tasks are configured from and stored in the WhatsConfigured Task Library and are associated with devices in the WhatsConfigured Task dialog. Additionally, if you use WhatsConfigured in WhatsUp Gold, you can view tasks associated with a specific device from the Device Properties - Tasks dialog in WhatsUp Gold.



About the WhatsConfigured Task Library

The WhatsConfigured Task Library displays all tasks configured for use in WhatsConfigured.



To access the WhatsConfigured Task Library:

- Access the WhatsConfigured Task Library from the WhatsUp Gold console at **Configure > WhatsConfigured Task Library**.

Use the WhatsConfigured Task Library to configure new or existing tasks.

- Click **Add** to configure a new task.
- Select an existing task, then click **Edit** to modify its configuration.
- Select an existing task, then click **Copy** to create a new task based on the selected task.
- Select an existing task, then click **Delete** to remove it from the list.
- Select a task, then click **Run Now** to perform the task immediately. The task will be run for all devices to which it is assigned. Additionally, to run a task only for a single device, use the Run Now option on the WhatsUp Gold Device Properties - Tasks dialog for a specific device.

Access the WhatsConfigured Task Library from the WhatsUp Gold console at **Configure > WhatsConfigured Task Library**.

Configuring tasks

There are two types of tasks that you can configure:

- Schedulable
- Password

Schedulable Tasks run associated task scripts on a regularly scheduled basis. Password Tasks modify credentials on the devices to which they are assigned.

Configuring schedulable tasks

Schedulable tasks are configured to run on the regularly scheduled basis that you choose. You can configure a task to run on a daily, weekly, monthly, yearly, or custom schedule.

The screenshot shows the 'WhatsConfigured Task' dialog box with the 'Schedule' tab selected. The 'Name' field contains 'Daily Running Config Backup'. The 'Description' field is empty. The 'Script' dropdown is set to 'Backup Startup Config'. The 'Save copy of configurations to' checkbox is checked, and the path is 'C:\Documents and Settings\Administrator\My Documents\WhatsConn'. The 'Schedule' tab is active, showing 'Enable this schedule' checked. Under 'Run this task', the 'Repeat interval' is 'Daily' and the 'Start Time' is '02:00:00'. Two radio buttons are present: 'every 1 day(s) at 02:00:00' (unselected) and 'every weekday at 02:00:00' (selected). The 'Next Scheduled Task' is '5/14/2010 2:00 AM'. At the bottom, there are two radio buttons: 'Keep up to 5 configuration backups' (selected) and 'Do not limit the number of configuration backups' (unselected). The 'OK', 'Cancel', and 'Help' buttons are on the right.

Configuring a daily task schedule

To configure a daily task schedule:

- 1 Open the Task Library from the WhatsUp Gold console at **Configure > WhatsConfigured Task Library**. The Task Library appears.
- 2 Click **New**. The WhatsConfigured Task dialog appears.
- 3 Select the **Schedule** tab.
- 4 Select **Enable this schedule**.

- 5 Under the **Interval** list, select **Daily**.

The screenshot shows the 'WhatsConfigured Task' dialog box with the 'Schedule' tab selected. The 'Name' field contains 'Daily Running Config Backup'. The 'Description' field is empty. The 'Script' dropdown is set to 'Backup Startup Config'. The 'Save copy of configurations to' checkbox is checked, and the path is 'C:\Documents and Settings\Administrator\My Documents\WhatsConn'. The 'Enable this schedule' checkbox is checked. Under 'Run this task', the 'Repeat interval' is set to 'Daily' and the 'Start Time' is '02:00:00'. Two radio buttons are present: 'every 1 day(s) at 02:00:00' (unselected) and 'every weekday at 02:00:00' (selected). The 'Next Scheduled Task' is '5/14/2010 2:00 AM'. At the bottom, there are two radio buttons: 'Keep up to 5 configuration backups' (selected) and 'Do not limit the number of configuration backups' (unselected). The 'OK', 'Cancel', and 'Help' buttons are at the bottom right.

- 6 Specify the task's **Start Time**.
- 7 Specify how often the task should be performed. For example, if you want the task to run every other day, specify that the task should repeat every 2 days. You can select to have the task **run every ___ day**, or **every weekday** at the specified time.

Configuring a weekly task schedule

To configure a weekly task schedule:

- 1 Open the Task Library from the WhatsUp Gold console at **Configure > WhatsConfigured Task Library**. The Task Library appears.
- 2 Click **New**. The WhatsConfigured Task dialog appears.
- 3 Select the **Schedule** tab.
- 4 Select **Enable this schedule**.

- 5 Under the **Interval** list, select **Weekly**.

The screenshot shows the 'WhatsConfigured Task' dialog box with the 'Schedule' tab selected. The 'Name' field contains 'Weekly Running Backup'. The 'Description' field is empty. The 'Script' dropdown is set to 'Backup Startup Config'. The 'Save copy of configurations to:' checkbox is checked, and the path is 'C:\Documents and Settings\Administrator\My Documents\WhatsConn'. The 'Enable this schedule' checkbox is checked. The 'Run this task' section shows a 'Repeat interval' of 'Weekly' and a 'Start Time' of '02 : 00 : 00'. Below this, it says 'every 2 week(s) at 02:00:00 on the selected day(s)'. The days 'Monday', 'Tuesday', 'Wednesday', 'Thursday', and 'Friday' are selected with checkboxes. 'Sunday' and 'Saturday' are not selected. The 'Next Scheduled Task' is '5/14/2010 2:00 AM'. At the bottom, the 'Keep up to 5 configuration backups' radio button is selected. The 'OK', 'Cancel', and 'Help' buttons are on the right.

- 6 Specify the task's **Start Time**.
- 7 Specify how often the task should be performed. For example, if you want the task to run to run every other week during the work week, specify that the task run every 2 weeks and select Monday through Friday.

Configuring a monthly task schedule

To configure a monthly task schedule:

- 1 Open the Task Library from the WhatsUp Gold console at **Configure > WhatsConfigured Task Library**. The Task Library appears.
- 2 Click **New**. The WhatsConfigured Task dialog appears.
- 3 Select the **Schedule** tab.
- 4 Select **Enable this schedule**.

- 5 Under the **Interval** list, select **Monthly**.

The screenshot shows the 'WhatsConfigured Task' dialog box with the 'Schedule' tab selected. The 'Name' field is 'Monthly Running Backup'. The 'Script' dropdown is 'Backup Startup Config'. The 'Save copy of configurations to' checkbox is checked, with the path 'C:\Documents and Settings\Administrator\My Documents\WhatsConn'. The 'Enable this schedule' checkbox is checked. Under 'Run this task', the 'Repeat interval' is 'Monthly' and the 'Start Time' is '02 : 00 : 00'. Two scheduling options are shown: 'on day 15 every 1 month at 02:00:00' (selected) and 'on the first Friday every 1 month at 02:00:00'. The 'Next Scheduled Task' is '5/15/2010 2:00 AM'. At the bottom, 'Keep up to 5 configuration backups' is selected. 'OK', 'Cancel', and 'Help' buttons are on the right.

- 6 Specify the task's **Start Time**.
- 7 Specify the day of the month the task should run. You can select a numerical date, such as the 15th, or a generic date, such as the third Wednesday.
- 8 Specify how often the task should be performed. For example, if you want the task to run every other month, specify that the task repeat every 2 months.

Configuring a yearly task schedule

To configure a yearly task schedule:

- 1 Open the Task Library from the WhatsUp Gold console at **Configure > WhatsConfigured Task Library**. The Task Library appears.
- 2 Click **New**. The WhatsConfigured Task dialog appears.
- 3 Select the **Schedule** tab.
- 4 Select **Enable this schedule**.

- 5 Under the **Interval** list, select **Yearly**.

The screenshot shows the 'WhatsConfigured Task' dialog box with the 'Schedule' tab selected. The 'Name' field contains 'Yearly Running Backup'. The 'Script' dropdown is set to 'Backup Startup Config'. The 'Save copy of configurations to' checkbox is checked, with the path 'C:\Documents and Settings\Administrator\My Documents\WhatsConn' entered. The 'Enable this schedule' checkbox is checked. Under 'Run this task', the 'Repeat interval' is set to 'Yearly' and the 'Start Time' is '02 : 00 : 00'. Below this, there are two radio button options: 'every June 1' (selected) and 'every first Friday of June'. The 'Next Scheduled Task' is listed as '6/13/2010 2:00 AM'. At the bottom, there are two radio button options for backups: 'Keep up to 5 configuration backups' (selected) and 'Do not limit the number of configuration backups'. The 'OK', 'Cancel', and 'Help' buttons are on the right.

- 6 Specify the task's **Start Time**.
- 7 Specify the day and month the task should run. You can select a month with a numerical date, such as the June 1st, or a generic date with a month, such as the first Friday of June.

Configuring a custom task schedule

To configure a custom task schedule:

- 1 Open the Task Library from the WhatsUp Gold console at **Configure > WhatsConfigured Task Library**. The Task Library appears.
- 2 Click **New**. The WhatsConfigured Task dialog appears.
- 3 Select the **Schedule** tab.
- 4 Select **Enable this schedule**.

- 5 Under the **Interval** list, select **Custom**.

The screenshot shows the 'WhatsConfigured Task' dialog box with the 'Schedule' tab selected. The 'Name' field contains 'Every Other Day Running Backup'. The 'Script' dropdown is set to 'Backup Startup Config'. The 'Save copy of configurations to' checkbox is checked, with the path 'C:\Documents and Settings\Administrator\My Documents\WhatsConn' entered. The 'Enable this schedule' checkbox is checked. Under 'Run this task', the 'Repeat interval' dropdown is set to 'Custom', and the 'Start Time' is set to '02 : 00 : 00'. Below this, the text 'every 2 days' is displayed. At the bottom, the 'Keep up to 5 configuration backups' radio button is selected. The 'OK', 'Cancel', and 'Help' buttons are on the right.

- 6 Specify the task's **Start time**.
- 7 Specify how often the task should be performed. You can select minutes, hours, or days. For example, you can specify that the task run at 2:00:00 AM every 2 days.

Configuring password tasks

Password tasks allow you to add, change, or remove device SNMP, SSH, or Telnet credentials as needed.

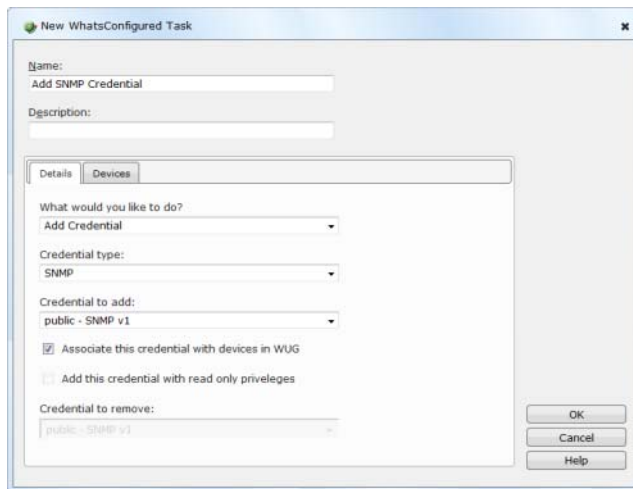


Note: Password tasks only modify credentials by device. Changes made using a WhatsConfigured password task do not effect the WhatsUp Gold Credential Library.

Adding a credential type

To add SNMP, SSH, or Telnet credentials to a device:

- 1 On the WhatsConfigured console, go to **Libraries > Task Library**. The WCD> Task Library appears.
- 2 Do one of the following:
 - Click **New** to configure a new task. The Select Task type dialog appears.
 - Select *Password Task*, then click **OK**. The New Task dialog appears.
 - or -
 - Select an existing Password task, then click **Edit**. The Edit Task dialog appears.
- 3 Enter or select the appropriate information in the dialog fields.
 - Enter a **Name** for the task. This name is listed in the WhatsConfigured Task Library.
 - Enter a brief **Description** for the task. This description is listed in the WhatsConfigured Task Library to help you differentiate it from other tasks.
- 4 Under **What you would like to do** using the Password Task, select *Add Credential*.



- 5 Select the **Credential Type** that you want to add, either *SNMP*, *SSH*, or *Telnet*.
- 6 Select the specific **Credential to add**.



Tip: On the WhatsUp Gold console, you can browse (...) to the Credentials Library.

- 7 Choose whether to **Associate this credential with devices in WUG**. Selecting this option will add the set of credentials to the selected device in WhatsUp Gold.
- 8 Choose whether to **Add this credential with read only privileges**. Selecting this option will disable the ability for other users to edit the credential.



Note: If you are using HP ProCurve series devices, you must select to **Add Credential** or **Change Credential** first, then select the **Add this credential with read only privileges** box to remove the *Operator* credential password or you must clear the **Add this credential with read only privileges** box to remove the *Manager* credential password. The SNMP credential type only allows *Manager* credentials and the SSH and Telnet credential types allow both *Manager* and *Operator* credential types.

- 9 Under **Devices to update**, click **Add** to select the device or device group to which you want to add the credentials.
- 10 Click **OK** to save changes.

Changing a device's credentials

To change a device's SNMP, SSH, or Telnet credentials:

- 1 On the WhatsConfigured console, go to **Libraries > Task Library**. The WCD> Task Library appears.
- 2 Do one of the following:
 - Click **New** to configure a new task. The Select Task type dialog appears.
 - Select *Password Task*, then click **OK**. The New Task dialog appears.

- or -
- 3 Select an existing Password task, then click **Edit**. The Edit Task dialog appears.
- 4 Enter or select the appropriate information in the dialog fields.
 - Enter a **Name** for the task. This name is listed in the WhatsConfigured Task Library.
 - Enter a brief **Description** for the task. This description is listed in the WhatsConfigured Task Library to help you differentiate it from other tasks.
- 5 Under **What you would like to do** using the Password Task, select *Change Credential*.

- 6 Select the **Credential Type** that you want to modify, either *SNMP*, *SSH*, or *Telnet*.
- 7 Select the specific **Credential to add**.

- 8 Additionally,
 - Choose whether to **Associate this credential with devices in WUG**. Selecting this option will add the set of credentials to the selected device in WhatsUp Gold.
 - Choose whether to **Add this credential with read only privileges**.



Note: If you are using HP ProCurve series devices, you must select to **Add Credential** or **Change Credential** first, then select the **Add this credential with read only privileges** box to remove the *Operator* credential password or you must clear the **Add this credential with read only privileges** box to remove the *Manager* credential password. The SNMP credential type only allows *Manager* credentials and the SSH and Telnet credential types allow both *Manager* and *Operator* credential types.

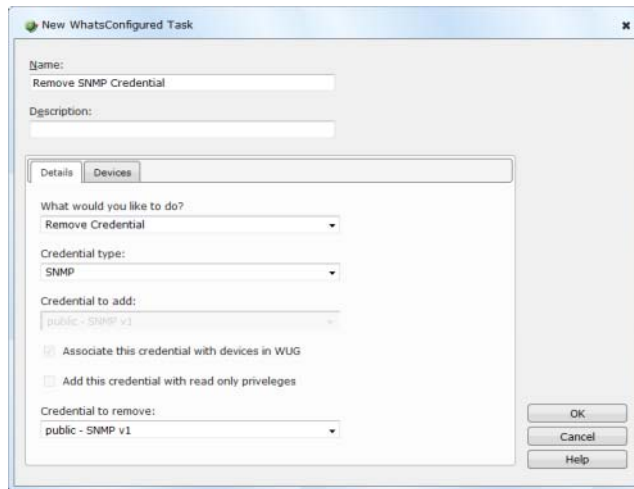
- 9 Select the specific **Credential to remove**.
- 10 Under **Devices to update**, click **Add** to select the device or device group to which you want to modify credentials.
- 11 Click **OK** to save changes.

Removing credentials from a device

To remove a device's SNMP, SSH, or Telnet credentials:

- 1 On the WhatsConfigured console, go to **Libraries > Task Library**. The WCD> Task Library appears.
- 2 Do one of the following:
 - Click **New** to configure a new task. The Select Task type dialog appears.
 - Select *Password Task*, then click **OK**. The New Task dialog appears.
 - or -
- 3 Select an existing Password task, then click **Edit**. The Edit Task dialog appears.
- 4 Enter or select the appropriate information in the dialog fields.
 - Enter a **Name** for the task. This name is listed in the WhatsConfigured Task Library.
 - Enter a brief **Description** for the task. This description is listed in the WhatsConfigured Task Library to help you differentiate it from other tasks.

- 5 Under **What you would like to do** using the Password Task, select *Remove Credential*.

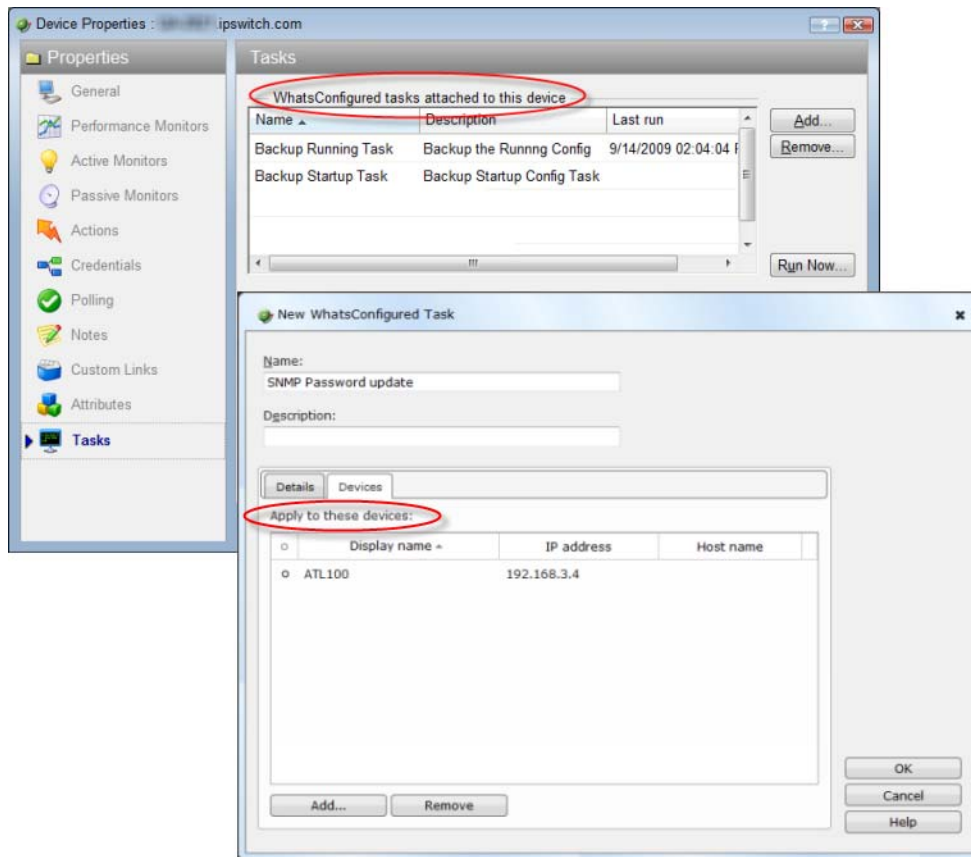


The screenshot shows a 'New WhatsConfigured Task' dialog box. The 'Name' field contains 'Remove SNMP Credential'. The 'Description' field is empty. The 'Details' tab is active, displaying a 'What would you like to do?' dropdown set to 'Remove Credential'. Below this, the 'Credential type' is set to 'SNMP' and the 'Credential to add' is 'public - SNMP v1'. There are two checkboxes: 'Associate this credential with devices in WUG' (checked) and 'Add this credential with read only privileges' (unchecked). At the bottom, the 'Credential to remove' dropdown is also set to 'public - SNMP v1'. The 'OK', 'Cancel', and 'Help' buttons are located in the bottom right corner.

- 6 Select the **Credential Type** that you want to remove, either *SNMP*, *SSH*, or *Telnet*.
- 7 Select the specific **Credential to remove**. This list is populated with credentials currently assigned to the device.
- 8 Under **Devices to update**, click **Add** to select the device or device group from which you want to remove credentials.
- 9 Click **OK** to save changes.

Assigning a task to a device

Tasks are assigned to individual devices from either the Device Properties - Tasks dialog or when configuring the task in the Task Library.



To assign a WhatsConfigured task to a device from Device Properties:

- 1 In the Device List or Map View, right-click a device, then click **Properties**. The Device Properties dialog appears.
- 2 Click **Tasks**. The Tasks section of Device Properties appears.
- 3 Under **WhatsConfigured tasks attached to this device**, click **Add**. The Add Task to Device dialog appears.
- 4 Select the task that you want to assign to the device, then click **OK**. If the list is empty, or you do not see the task you want to assign, browse (...) to the WhatsConfigured Task Library to configure a new task.

For information on assigning a device during task configuration, see Configuring Tasks.

Viewing Task results

The Task Results dialog displays results for tasks that have been run using the Scheduled Task Library's **Run Now** option.

To view Task Results for a task:

- 1 On the WhatsConfigured console, go to **Libraries > Task Library**. The Task Library appears.
- 2 Select a task, then click **Run Now**. A dialog displaying the task's progress appears.
- 3 When the task completes, click **View Results**. The Task Results dialog appears.

The dialog displays the following result for a task that was ran using the **Run Now** option:

- **Task status.** The result of the entire task. A task is considered to be successful only if the task completes successfully for all devices for which it runs. In the event that the task fails, the task message displays information regarding the failure.
- **Task Message.** A message that explains why the task failed. If the task runs successfully for all devices, this field is empty.
- **Task Devices.** The devices for which the task ran.



Tip: Select a device to view its result information in the following section of the dialog.

Below, the dialog displays device-specific results in six tabs.



Tip: Select a dialog tab to view information for its specific dialog fields.

The **Output** tab displays the task's result, relevant messages, and a trace of all communication between the device and the WhatsConfigured service.

For each task it displays:

- **Result.** The result of the task for the selected device.
- **Message.** Any message pertaining to the task for the selected device. In some instances, this field may be empty.
- **Trace.** A history of all communication that takes place between the device and the WhatsConfigured service during the task's attempted completion. If the task collects a configuration file as part of the task, it is included in the trace. If the task was successful for this device, the trace displays what the command prompt would have looked like if the user consoled into the device and run the commands manually using a command prompt.

- If the task failed and no communication took place between the device and the WhatsConfigured service due to communication or configuration errors, the box displays *"No communication with the device was recorded."* Finding the cause of this failure may be accomplished by reviewing the credentials listed on the Settings tab, reviewing device configurations, attempting to communicate with the device manually, or by checking the log.

The **Script** tab displays the task script assigned to this task as it is saved in the Task Script Library, and how the task looks after it is processed through the WhatsConfigured task runner.

For each task it displays:

- **Script Text.** The script assigned to be run by the task. If this script is a custom script, it appears exactly as it did when it was configured in the New/Edit WhatsConfigured Task Script dialog. If this is a predefined password or backup task, the script displayed is the script chosen for this device based on the WhatsConfigured script registry.



Note: Scripts for predefined WhatsConfigured tasks are looked up based on the OID associated with the device. If there is no OID assigned to the device, the lookup fails and no script is listed. OID's can be assigned to a device from the Device Properties - Tasks dialog, or collected by discovering the device. Due to the large number of devices and their varying commands this script to device mapping may fail.

- **Processed Text.** The WhatsConfigured scripting language allows for variable replacement within scripts. WhatsConfigured pre-defined scripts utilize this ability when running password tasks. Before the script is run the script is processed and all variable references are replaced with the variables corresponding value. The processed text displays the resulting script after processing. This field allows the user to ensure variable declarations are being assigned and interpreted properly.



Tip: If you are experiencing a problem with a script, **Save** the results listed in the script tab to a text (.txt) file. If you contact Technical Support, this file will aid in troubleshooting your script problem.

The **Variables** tab displays the name and value of all variables associated with the task script.

For each task it displays:

The **Commands** tab displays a list of the commands as they were interpreted by the WhatsConfigured script runner. It also displays the results of those commands if they were run against the device when the task was run.

For each task it displays:

- **Command.** The specific command; for example, *login* or *show configuration*.
- **Result.** The success or failure of the command when it was ran by the task.
- **Output.** The results of the responses declared by the WhatsConfigured script language.

The **Log** tab displays any error messages that were logged as the task ran.

The **Settings** tab displays the protocol credentials used to complete the task.

For each task it displays:

- **Type.** The type of protocol credentials; for example, SSH or Telnet.



Note: WhatsConfigured defaults to SSH credentials when available. If SSH credentials are not assigned to a device, WhatsConfigured will look for/use Telnet credentials.

- **Name.** The name of the credentials as assigned in the Credentials Library.
- **Description.** The description of the credentials as assigned in the Credentials Library.

CHAPTER 5

Using Task Scripts

In This Chapter

About Task Scripts.....	25
Using the WhatsConfigured Task Script Library.....	25
Configuring custom task scripts	27

About Task Scripts

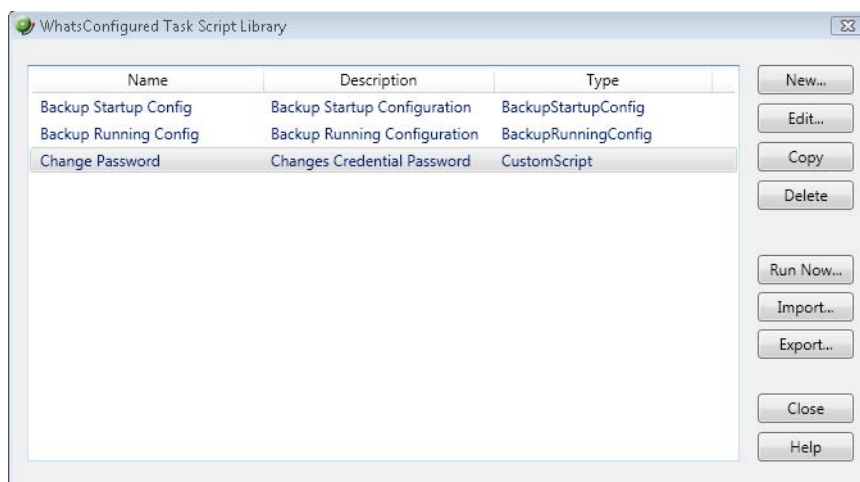
Task scripts login to devices through SSH or Telnet and run command-line interface (CLI) commands on devices. These tasks can perform a number of operations, such as restoring or backing up a running or startup configuration, or changing an application password.

WhatsConfigured comes with several pre-configured task scripts; you can also configure your own custom task scripts using the WhatsConfigured Custom Script Language.

Task scripts are configured from and stored in the Task Script Library and associated to WhatsConfigured tasks in the WhatsConfigured Task dialog.

Using the WhatsConfigured Task Script Library

The WhatsConfigured Task Script Library displays all task scripts currently configured for use in WhatsConfigured tasks.



There are two pre-configured task scripts available for use in WhatsConfigured.

- Backup Running Config
- Backup Startup Config

Use the WhatsConfigured Task Script Library to configure new or existing task scripts:



Note: The **Edit**, **Copy**, **Delete**, and **Export** buttons are disabled for the default, pre-configured task scripts, as you cannot modify or remove default scripts.

- Click **New** to configure a new task script.
- Select a custom task script, then click **Edit** to change its configuration.
- Select a custom task script, then click **Copy** to make a duplicate of the selected task script.
- Select a custom task script, then click **Delete** to remove it from the library.



Caution: When you delete a non-default task script from the WhatsConfigured Task Script Library, it is removed from all tasks that are using that task script.

- On the WhatsUp Gold console, select a task script, then click **Run Now** to run the task script immediately.



Note: The option to run a script on demand via the Task Script Library is not available on the WhatsUp Gold web interface.

- Select a custom task script, then click **Export** to export it as an XML file.
- Click **Import** to import an XML file into the library.



Note: Modifying XML files or attempting to create an XML file from scratch can invalidate a script file.



Note: You can only Export custom task scripts.

About the backup startup config task script

The backup running config task script makes a backup copy of a device's startup config and stores it in the WhatsUp Gold database. After you have made a backup copy of a startup config, you can restore it on the device at any time for as long as the copy is stored in the database.



Tip: You can set the number of maximum configuration files to store in WhatsUp Gold database on the New WhatsConfigured Task dialog's Schedule tab.

You can view a device's current and archived config files on the Device Properties - Tasks dialog.

About the backup running config task script

The backup running config task script makes a backup copy of a device's running config and stores it in the WhatsUp Gold database. After you have made a backup copy of a running config, you can restore it on the device at any time for as long as the copy is stored in the database.



Tip: You can set the number of maximum configuration files to store in WhatsUp Gold database on the New WhatsConfigured Task dialog's Schedule tab.

You can view a device's current and archived config files on the Device Properties - Tasks dialog.

Configuring custom task scripts

In addition to the pre-configured task scripts included in WhatsConfigured, you can configure custom task scripts that either configure devices or gather device data and store it in the WhatsUp Gold database. These tasks are configured using the WhatsConfigured Custom Script Language, a combination of WhatsConfigured and device commands.

CHAPTER 6

The WhatsConfigured Custom Script Language

In This Chapter

About the WhatsConfigured Custom Script Language..... 28

About the WhatsConfigured Custom Script Language

WhatsConfigured users can write custom scripts that log in to devices through Telnet or SSH and run CLI commands on their devices. Scripts can be used to configure devices or to capture information about them in the WhatsUp Gold database. For example, the following script uses Cisco IOS commands to capture a Cisco device's running configuration in the WhatsUp Gold database under the "running-config" key.

```
#  
# Cisco IOS Backup Running Configuration  
#  
# login to the device  
@login  
#enter privileged mode  
@enable  
  
# display the running configuration of the device and capture it in the  
WUG database [running-config] show run  
  
# logout from the device  
[-] exit
```


The WhatsConfigured custom script language is relatively simple and consists primarily of command-line interface (CLI) commands. The language is not meant to be a full-featured scripting language, such as JavaScript or VBScript, but rather is kept simple so that is accessible to all levels of WhatsConfigured users, including those with minimal programming skills. In order to meet the standards of this target audience, the language contains no constructs for looping, branching, or creating subroutines; it only supports simple sequences of commands.

The custom script language has three possible elements:

- Comments
- Variables
- Commands

Each of these elements will be explained in detail in the following sections.

CHAPTER 7

Using WhatsConfigured Comments

In This Chapter

About WhatsConfigured comments 30

About WhatsConfigured comments

In a script, you have the option to insert details or notes about the script. These notes and details are entered as comments, or lines having # as their first non-whitespace character. Comments are ignored by the script interpreter.



Note: A # character is interpreted as the beginning of a comment only if it is the first non-whitespace character on a line. If the # appears later in the line, it has no special significance.

Examples

```
# This is a comment
```

```
#       This is also a comment
```

```
123 # This is not a comment because '#' is not the first non-whitespace  
character in the line
```

CHAPTER 8

Using WhatsConfigured Variables

In This Chapter

About variables	31
Accessing protocol settings.....	32
Using reserved WhatsConfigured variable names	32

About variables

Variables are useful for giving names to values referenced in a script, especially values that are referenced multiple times. For example,

```
CommandTerminator = "\r\n"
TFTPServerAddress = 192.168.10.50
TransferFileName= startup-config.txt
@login
@write "copy tftp start"
@write $(CommandTerminator)
@write "$(TFTPServerAddress)"
@write "(TransferFileName)"
@write $(CommandTerminator)
```

Variable definitions

A variable definition must appear on a line by itself, in the following form:

Name = Value

In the example above, Name is the variable's title, and Value is the variable's value.

Variable names must begin with an alphabetic character or an underscore (a-z, A-Z, _), and subsequent characters can be any alphanumeric character or an underscore (a-z, A-Z, 0-9, _).



Note: Spaces are not allowed in variable names.

The variable's value consists of all text on the right side of = with leading and trailing whitespace removed. For example,

```
FirstUSPresident = The Honorable George Washington
```

The example above defines a variable named "FirstUSPresident" with the value "The Honorable George Washington".

A variable's value can be referenced anywhere in the script after the variable is defined. A variable reference consists of '\$' immediately followed by the variable's name in parentheses, as shown below.

```
$(FirstUSPresident)
```

A variable reference is replaced by the variable's value. If the variable is defined multiple times in the script, the most recent definition is used. In the example above, the variable reference "`$(FirstUSPresident)`" would be replaced by "The Honorable George Washington".

Accessing protocol settings

When a WhatsConfigured script runs, it executes against a particular device. The script uses the device's SSH or Telnet credentials to login to the device. Sometimes it is necessary for a script to directly access the protocol settings being used in a set of credentials. The protocol settings can be accessed through the variables listed in the following table.



Note: The values of these variables are read-only and cannot be modified by scripts, though scripts are free to reference their values.

Name	Description	Example
Settings.UserName	The SSH or Telnet username.	admin
Settings.Password	The SSH or Telnet password.	secret
Settings.PrivilegedPassword	The enable or privileged mode password.	supersecret

Using reserved WhatsConfigured variable names

Script authors can use any names they want for their variables. However, the variables listed below are used internally by WhatsConfigured. As a general rule, script authors should avoid using these variable names in their scripts.

- AccessPrivilege
- CommandPrompt
- CommandTerminator
- LoginTerminator
- MorePrompt
- MoreResponse
- NewPassword
- NewPrivilegedPassword
- NewUserName
- Password
- PasswordPrompt
- PrivilegedPassword
- TFTPServerAddress
- TransferFileName
- UserName
- UserNamePrompt

Occasionally, a script may need to re-define one or more of these variables to affect the internal operation of WhatsConfigured commands. The section on WhatsConfigured commands describes the meanings and uses of these variables, and how scripts can re-define them to modify the behavior of WhatsConfigured commands.

CHAPTER 9

Using WhatsConfigured Commands

In This Chapter

About commands.....	34
About basic WhatsConfigured command syntax	35
About strings and regular expressions in WhatsConfigured.....	35
Storing WhatsConfigured command output in the WhatsUp Gold database	37
Editing WhatsConfigured command output.....	37
Using WhatsConfigured commands with queries.....	38
About WhatsConfigured command layout.....	39
WhatsConfigured script variables affecting command execution...	40
About WhatsConfigured command types	41

About commands

Beyond commands and variables definitions, the other lines in a script contain the commands to be executed by the script.

Examples

```
@login
```

```
@enable
```

```
config t
```

```
line vty 0 4
```

```
login local
```

```
exit
```

```
username $(NewUserName) password $(NewPassword)
```

```
exit
```

```
[-] exit
```

About basic WhatsConfigured command syntax

There are two types of commands that can be included in a WhatsConfigured custom script:

- WhatsConfigured commands
- Device commands

WhatsConfigured commands are executed by WhatsConfigured itself. Device commands are executed by the device. WhatsConfigured commands begin with @ to distinguish them from device commands. Any command whose text begins with @ is a WhatsConfigured command, while any other command is a device command. In the previous example script, the @login and @write commands are WhatsConfigured commands, while all other commands are device commands.

WhatsConfigured defines the following commands:

- @login
- @connect
- @write
- @read
- @read-more

The syntax for each of these commands is defined by WhatsConfigured. In contrast, Device commands are written using the native CLI commands supported by the device (IOS or CasOS commands for Cisco devices, Linux commands for Linux devices, etc.) These commands can use whatever syntax is required by the device's CLI command set.

In its simplest form, a command is just a string specifying the name of a command along with any parameters it requires. For example, the following script contains two simple commands:

```
@ login
```

```
username $(NewUserName) password $(NewPassword)
```

About strings and regular expressions in WhatsConfigured

WhatsConfigured commands make use of two specific types of values, strings and expressions.

Strings are used to represent literal text values; string values are sequences of characters delimited by double quotes, such as:

```
"Four score and seven years ago"
```

Escape sequences (used to define special characters within strings) may be any of the following:

Escape sequence	Represents
\0	Null character
\'	Single quote
\"	Double quote
\?	Literal question mark
\\	Backslash
\a	Bell alert (audible bell)
\b	Backspace
\f	Formfeed
\n	New line
\r	Carriage return
\t	Horizontal tab
\v	Vertical tab

Additionally, the \x escape sequence can be used to include arbitrary characters in strings, including unprintable and control characters. \x should be followed by one to four hexadecimal digits which specify the value of the desired character. For example,

```
"This is Control-S: \x13"
```

Regular expressions are used for matching patterns in the output of script commands. Regular expression values are .NET regular expression strings delimited by forward slashes. For example, the following regular expression might be used to match the command prompt on a particular device (i.e., one or more characters followed by > or #):

```
/ .+(>|#)/
```

Because forward slashes are used to delimit regular expression values, including a forward slash as part of the regular expression itself, requires the use of the // escape sequence. For example, the following regular expression matches one or more characters followed by a forward slash followed by one or more characters followed by #:

```
/ .+// .+#/
```



Note: All regular expression matching is case-insensitive.

Storing WhatsConfigured command output in the WhatsUp Gold database

Most WhatsConfigured script commands return the output received from the device when the command was executed. For example, the `show run` command on Cisco devices displays the running configuration of the device. Or, on a Linux device, the `ls-al` command displays the contents of the current working directory. It is sometimes desirable to capture the output of the WhatsConfigured command and store it in the WhatsUp Gold database. To facilitate the storage of command output in the WhatsUp Gold database, a command can be preceded by a KEY which specifies the key under which the command's output should be restored in the WhatsUp Gold database. For example, the following command stores its output under the `running-config` key in the WhatsUp Gold database.

```
[running-config] show run
```

This means, execute the `show run` command and store its output in the WhatsUp Gold database under the `running-config` key.



Note: Key names can include dashes, underscores, and alphanumeric characters (-,_,a-z, A-Z). However, spaces are not allowed in key names.

Editing WhatsConfigured command output

Before storing a command's output in the WhatsUp Gold database, it is sometimes desirable to edit the output. For example, a command might place empty lines at the beginning or end of its output, and you may want to remove these empty lines before putting the output into the database. For situations like this, several operators are provided for editing command output. These operators are specified as part of the command's KEY. For example, the following command specifies that 4 lines should be trimmed from the output of the `ls -al` command before the output is stored in the WhatsUp Gold database under the key, `file-list`.

```
[file-list, trim-start-lines = "4"] ls -al
```

The following output editing operators are provided:

Name	Value	Meaning	Example
trim-start-lines	Integer	Trim the first N lines from the commands output	trim-start-lines="1"
trim-end-lines	Integer	Trim the last N lines from the commands output	trim-end-lines="1"
trim-start	String or regular expression	Trim all output before and including the first match of the specified string or regular expression	trim-start="#\n#\n#\n"
trim-end	String or regular expression	Trim all output including and after the last match of the specified string or regular expression	trim-end="#\n#\n#\n"
trim-before	String or regular expression	Trim all output before the first match of the specified string or regular expression	trim-before="!"
trim-after	String or regular expression	Trim all output after the last match of the specified string or regular expression	trim-after"!"
remove-lines	String or regular expression	Remove all lines that match the specified string or regular expression	remove-lines=/system time.+/

If multiple editing operators are used in the same command, they are applied in the order shown in the previous table.

```
[file-list, trim-start-lines = "4", trim-end="\n\n\n"]  ls -al
```

Using WhatsConfigured commands with queries

Some Device commands require users to answer a question before the command is executed. For example, the `enable` command on Cisco devices queries the user for a password before executing the command. For this reason, a command can optionally specify a QUERY which specifies the question asked by the device and the answer that should be given to the question. The QUERY is specified after the command within curly braces. For example,

```
shutdown { "Are you sure? ", "Y" }
```

The first value inside the curly braces is a String or Regular Expression describing the query prompt displayed by the device. The second value inside the curly braces is a String specifying the query response that should be entered in response to the query prompt. When the script interpreter executes this command, it will first send `shutdown` to the device. Next, it will wait until it receives the "Are you sure?" query prompt. Then, it will send `Y` to the device as the query response. Finally, the device will execute the command.



Note: Only Device commands can have a QUERY. WhatsConfigured commands do not need a QUERY, and, in fact, may not have one.

For example,

```
enable { $(PasswordPrompt), "$(Settings.PrivilegedPassword)" }
```

About WhatsConfigured command layout

The general format of a script command is:

KEY COMMAND QUERY

For example,

```
[last-words] shutdown { "Are you sure? ", "Y" }
```

As previously explained, KEY specifies the key to use when storing the command's output in the WhatsUp Gold database, and possibly operations for trimming the command output. COMMAND is the text for the command itself. QUERY specifies the query prompt and query response for commands that ask a question. COMMAND is required, while KEY and QUERY are optional.

Since commands can become long, it is legal to put the KEY, COMMAND, and QUERY parts of a command on different lines. For example, the following commands are equivalent:

```
[last-words] shutdown { "Are you sure? ", "Y" }
```

```
[last-words]
```

```
shutdown
```

```
{ "Are you sure? ", "Y" }
```

```
[last-words]
```

```
shutdown { "Are you sure? ", "Y" }
```

```
[last-words] shutdown
```

```
{ "Are you sure? ", "Y" }
```

While the KEY, COMMAND, and QUERY can be on different lines from each other, each of these individual elements must start and end on the same line (i.e., they cannot span multiple lines). For example, the following commands are not valid:

```
[  
  
last-words  
  
] shutdown { "Are you sure? ", "Y" }  
  
[last-words] shut  
  
down { "Are you sure? ", "Y" }  
  
[last-words]  
  
shutdown  
  
{  
  
"Are you sure? ",  
  
"Y"
```

WhatsConfigured script variables affecting command execution

When running a script, WhatsConfigured defines several script variables that contain information necessary to execute the script's commands. For example, the `CommandPrompt` variable contains a pattern (i.e., string or regular expression) that describes the command prompt string used by the device. This pattern is used to detect when the device is prompting for a command. Several other variables are also defined. A complete list of all script variables affecting command execution are listed in the following table.

WhatsConfigured's assigns default values to each of these variables. If a script author wants to override WhatsConfigured's default behavior, he or she may do so by re-defining one or more of these variables. For example, if a script wants to override the command prompt pattern used to run the script, it can re-define the `CommandPrompt` variable to contain the pattern of choice.

Name	Value	Meaning	Example
UserNamePrompt	String or regular expression	Pattern describing the username prompt displayed by the device when a user logs in	"login as:"
PasswordPrompt	String or regular expression	Pattern describing the password prompt displayed by the device when a user logs in	"password:"
CommandPrompt	String or regular expression	Pattern describing the command prompt displayed by the device when prompting the user for a command	/.+(>)/
MorePrompt	String or regular expression	Pattern describing the "more" prompt displayed by the device when displaying paged output	/--More-- --More--/
MoreResponse	String	String to be entered in response to a "more" prompt	""
LoginTerminator	String	Line termination sequence to be used with logging in	"\r\n"
CommandTerminator	String	Line termination sequence to be used when executing commands	"\n"

About WhatsConfigured command types

There are two types of commands in a WhatsConfigured script: device commands and WhatsConfigured commands. Device commands can be any CLI command supported by a device, and are executed by the device. WhatsConfigured commands start with '@' and are executed by WhatsConfigured rather than by the device. The following sections describe the available commands and explain when and how to use them. Most scripts will use a combination of Device commands and WhatsConfigured commands, although it is possible to write scripts using only WhatsConfigured commands.

@login

Typically, the first step in any WhatsConfigured script is to login to the device. This is typically done with the WhatsConfigured @login command. The @login command can be used to login to devices that use a traditional user-name/password login procedure that works as follows:

- 1 The device prompts the user for their user name
- 2 The user enters their user name
- 3 The device prompts the user for their password
- 4 The user enters their password

- 5 If login is successful, the device displays a command prompt and waits for the user to run commands

The `@login` command has no parameters, and is invoked as follows:

```
@login
```

When the `@login` command is executed, it does the following:

- 1 When it detects `UserNamePrompt`, it sends `Settings.UserName` to the device followed by `LoginTerminator`.
- 2 When it detects `PasswordPrompt`, it sends `Settings.Password` to the device followed by `LoginTerminator`.
- 3 When it detects `MorePrompt`, it enters `MoreResponse`.
- 4 After entering the user name and password, if `@login` detects `CommandPrompt`, it assumes that login was successful. Otherwise, it assumes that login failed.

If at any time the device's output stalls for more than `Settings.ReadTimeout` seconds, it is assumed that something is wrong, and the script returns failure.

@enable

Many device configuration tasks require a script to enter a privileged mode in order to execute the necessary device commands. On many devices, privileged mode is entered using the `enable` command. Typically, running the `enable` command on a device requires the user to enter a user name and/or password. For devices that implement this style of `enable` command, scripts can use the WhatsConfigured `@enable` command to easily enter privileged mode. The `@enable` command has no parameters, and is invoked as follows:

```
@enable
```

When the `@enable` command is executed, it does the following:

- 1 It sends `enable` to the device followed by `CommandTerminator`.
- 2 If it detects `UserNamePrompt`, it sends `Settings.UserName` to the device followed by `CommandTerminator`.
- 3 If it detects `PasswordPrompt`, it sends `Settings.PrivilegedPassword` to the device followed by `CommandTerminator`. If `Settings.PrivilegedPassword` is empty, it uses `Settings.Password` instead.
- 4 After entering the user name and password (if necessary), if `@enable` detects `CommandPrompt`, it assumes that `enable` was successful. Otherwise, it assumes that `enable` failed.

If at any time the device's output stalls for more than `Settings.ReadTimeout` seconds, it is assumed that something is wrong, and the script returns failure.

Device commands

After invoking the `@login` command (and possibly `@enable` as well), most scripts contain a sequence of Device commands that are sent to the device for execution. A typical Device command is shown below:

Device commands are executed as follows:

```
[last-words] shutdown { "Are you sure? ", "Y" }
```

The script sends the command text to the device. It terminates the command with `CommandTerminator`.

If the command has a query, the device returns the query to the script. When it detects the query prompt, the script sends the query response to the device.

Next, the device executes the commands, and sends its output back to the script.

If the command's output is long enough to result in more prompts, when the script detects a `MorePrompt`, it sends `MoreResponse` to the device.

The script consumes the command's output until it detects `CommandPrompt`, at which point it assumes that the command's output is complete.

If at any time the device's output stalls for more than `Settings.ReadTimeout`, it is assumed that something is wrong, and the script returns failure.

If the command succeeds, and it has a KEY, its output is saved in the WhatsUp Gold database.

The following script is typical:

```
@login

enable { "password: ", "${Settings.PrivilegedPassword}" }

[running-config] show run

[-] logout
```

This script first logs in with the `@login` command, then enters privileged mode with the `@enable` command.

Next, the script sends the `show run` command to the device. The output of this command is saved in the WUG database under the `running-config` key.

Finally, the script sends the `logout` command to the device, at which point the device closes the network connection.

The `[-]` key on the `logout` command tells WhatsConfigured not to expect any output from the command, because the command causes the device to close the network connection.

Typically, receiving no output from a command indicates failure, but in the case of `exit` or `logout` commands (or any other command that closes the connection), a lack of output does not indicate failure. Script authors can use the `[-]` key to indicate such commands and prevent WhatsConfigured from returning failure when the device closes the connection.

Low-level commands

Many scripts will use only `@login`, `@enable`, and `Device` commands to implement their functionality. However, some devices have non-standard Telnet or SSH interfaces that won't work with `@login` and `Device` commands. For example, some devices have non-standard login procedures for which the `@login` command will not work. Other devices have menu-driven interfaces rather than a standard command-prompt-style interface. For whatever reason, if `@login`, `@enable`, or `Device` commands do not work for a particular device, WhatsConfigured provides a set of low-level commands that can be used to interact with virtually any device, no matter how non-standard its interface might be.

The `@connect` command allows a script to precisely control the process of logging in to a device. The `@write` command allows a script to control exactly what input is sent to a device. The `@read` command allows a script to read output from a device and optionally store it in the WhatsUp Gold database.

@connect

The `@connect` command is an alternative to `@login` in cases where a script needs to precisely control the login process (e.g., in cases where `@login` doesn't work for a particular device). The `@connect` command connects to a device without trying to log in. After connecting to a device with `@connect`, if the device requires users to login, the script can control the login process precisely using the `@write` and `@read` commands, which are described later.

When calling `@connect`, scripts can specify one or more patterns (i.e., strings or regular expressions) that specify the output the script expects to receive from the device when it connects. These patterns are used by `@connect` to detect the end of the device output. `@connect` will assume that device output is complete when either the output matches one of the specified patterns, or no new output has been received from the device for `Settings.ReadTimeout` seconds. For example,

```
@connect "login as: ", "user name: "
```

When executed, `@connect` connects to the device, and reads whatever output comes back from the device. If the output matches one of the specified patterns, the command succeeds. If the connection attempt fails entirely, or the output received from the device does not match any of the specified patterns, the command fails (as well as the entire script). As with any other command, a KEY can be specified to capture the command's output in the WhatsUp Gold database, although one would rarely want to store the output of an `@connect` command.

If no patterns are specified (as shown below), `@connect` connects to the device, and returns whatever output comes back from the device. In this case, the command succeeds as long as a connection is successfully established with the device.

@connect-more

Some devices return paged output that requires `more` prompts when you initially connect to them. If a script needs to handle `more` prompts during the connection process, it can use the `@connect-more` command instead of `@connect`. `@connect-more` works just like `@connect`, except that it handles `more` prompts during the connection process, while `@connect` does not. Specifically, if `MorePrompt` is detected during the connection process, `@connect-more` sends `MoreResponse` to the device.

```
@connect-more "login as: ", "user name: "
```

@write

The `@write` command can be used to send a string of characters to the device. This command allows a script to precisely control what input is being sent to the device. For example, the following script sends the `show run` command to the device, followed by the `CommandTerminator` (typically `\n` or `\r\n`).

```
@write "show run"
```

```
@write $(CommandTerminator)
```

@read

The `@read` command can be used by scripts to read the output coming back from the device. Typically, a call to `@read` will immediately follow a call to `@write`. When calling `@read`, scripts can specify one or more patterns (i.e., strings or regular expressions) to help `@read` detect the end of the device output. `@read` will assume that device output is complete when either: the output matches one of the specified patterns, or no new output has been received from the device for `Settings.ReadTimeout` seconds. Often, the output will end with `CommandPrompt`, so `"@read $(CommandPrompt)"` is a common way to call `@read`. If desired, the output received from the device can be stored in the WhatsUp Gold database, as shown below:

```
@write "show run"
```

```
@write $(CommandTerminator)
```

```
[running-config, trim-end-lines = 1] @read $(CommandPrompt)
```

When executed, `@read` will read whatever output comes back from the device. If the output matches one of the specified patterns, the command succeeds. If the output received from the device does not match one of the specified patterns, the command fails (as well as the entire script).

If no patterns are specified (as shown below), `@read` will return whatever output comes back from the device. In this case, the command will succeed as long as the connection to the device is still open.

@read-more

Some devices return paged output that requires `more` prompts. If a script needs to handle `more` prompts during a read operation, it can use the `@read-more` command instead of `@read`. `@read-more` works just like `@read`, except that it handles `more` prompts during the reading process, while `@read` does not. Specifically, if `MorePrompt` is detected during the reading process, `@read-more` will send `MoreResponse` to the device.

CHAPTER 10

Script Examples

In This Chapter

Example Scripts 47

Example Scripts

This example shows a typical script that uses @login to login to the device, uses @enable to enter privileged mode, and then executes several Device commands.

```
@login

@enable

[running-config] show run

[-] exit
```

This example shows how to login to a device and run a command using only low-level WhatsConfigured commands:

```
@connect "login as: "

@write "${Settings.UserName}"

@write $(LoginTerminator)

@read "password: "

@write "${Settings.Password}"

@write $(LoginTerminator)

@read $(CommandPrompt)

@write "exit"

@write $(CommandTerminator)
```

This example shows how to combine high-level commands and low-level commands in the same script as above:

```
@login

@enable

@write "copy tftp start"

@write $(CommandTerminator)

@write "$(TFTPServerAddress)"

@write $(CommandTerminator)

@write "$(TransferFileName)"

@write $(CommandTerminator)

@write $(CommandTerminator)

[-] exit
```

WCD3.0 Using Policies

In This Chapter

About policies.....	49
About the Policy Library.....	50
Configuring a policy	50
Auditing a policy.....	52

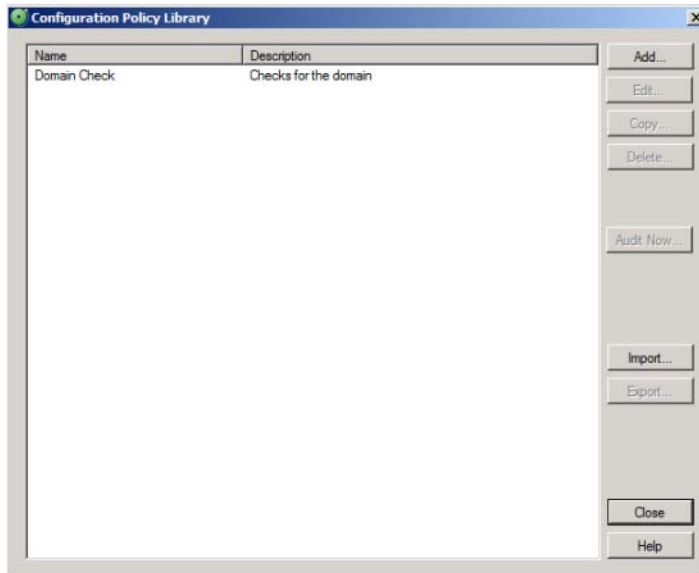
About policies

WhatsConfigured policies search through archived configuration files for strings that are either expected or not expected within the file(s).

Policies can be added to Alert Center Task Threshold's. When a scheduled task fails a policy, any associated notification policies alert you that the policy has failed due to unexpected content that has been flagged in an archived config file.

About the Policy Library

The WhatsConfigured Policy Library displays all policies currently configured for use with WhatsConfigured archive configuration files.



To access the Policy Library:

From the WhatsConfigured console's main menu, go to **Libraries > Policy Library**.

Use the WhatsConfigured Policy Library to configure new or existing policies.

- Click **Add** to configure a new policy.
- Select a policy, then click **Edit** to modify its configuration.
- Select a policy, then click **Copy** to make a duplicate of the selected policy.
- Select a policy, then click **Delete** to remove it from the library.
- Select a policy then click **Audit Now** to audit (test) a policy.
- Click **Import** to add a previously configured policy to the Policy Library.
- Click **Export** to save the policy as an .xml file to another location.

Configuring a policy

Use the Configuration Policy dialog to configure a WhatsConfigured Policy.

To configure a WhatsConfigured Policy:

- 1 From the main menu of the
- 2 Click **Add**. The Configuration Policy dialog appears.

- or -

Select an existing policy, then click **Edit**. The Configuration Policy dialog appears.

The screenshot shows the 'New Configuration Policy' dialog box. It has a title bar with a green icon and the text 'New Configuration Policy'. Inside, there are two text input fields: 'Name:' and 'Description:'. Below these are two sections: 'Include Patterns' and 'Exclude Patterns'. Each section contains a table with three columns: 'Pattern', 'Regex', and 'Ignore Case'. The 'Include Patterns' table has one row with empty fields. The 'Exclude Patterns' table has one row with empty fields. Below each table are 'Add' and 'Remove' buttons. At the bottom right are 'OK', 'Cancel', and 'Help' buttons.

- 3 Enter a **Name** for the policy. This name is displayed in the WhatsConfigured Policy Library.
- 4 Enter a short **Description** for the policy. This description is displayed next to the policy's name in the WhatsConfigured Policy Library.
- 5 In the following sections of the dialog, you have the opportunity to specify strings that you either expect or do not expect to see within the configuration files the policy audits. You can choose to enter only include patterns, only exclude patterns, or both.



Note: The more restrictive the audit criteria, the less audit results you may obtain as a result.

- 6 Under the **Include Patterns** section of the dialog, click **Add** to enter a string that you expect to see in the archived configuration files. Additionally,
 - Select **RegEx** if you want the string to be interpreted as a Regular Expression.
 - Select **Ignore Case** the case of the string is irrelevant to the string.



Tip: Select an include pattern, then click **Remove** to delete it from the list.

- 7 Under the **Exclude Patterns** section of the dialog, click **Add** to enter a string that you do not expect to see in the archived configuration files. Additionally,
 - Select **RegEx** if you want the string to be interpreted as a Regular Expression.
 - Select **Ignore Case** the case of the string is irrelevant to the string.



Tip: Select an include pattern, then click **Remove** to delete it from the list.

- 8 Click **OK** to save changes.

Auditing a policy

Use the Configuration Policy Audit dialog to configure the criteria by which the WhatsConfigured policy should be verified.

To audit a WhatsConfigured policy:

- 1 On the WhatsConfigured console's main menu, go to **Libraries > Policy Library**. The Policy Library appears.
- 2 Click **Add**. The Configuration Policy dialog appears.
- or -
Select an existing policy, then click **Edit**.
- 3 Select a policy, then click **Audit Now**. The Configuration Policy Audit dialog appears.
- 4 Under the **Audit Criteria** section of the dialog, click **Add** to select the device(s) against which to audit the policy.



Tip: To delete a device from the list, select it, then click **Remove**.

- 5 Select the **Archive Key** of the configuration files for which the policy will be audited. For example, to view audit results for running config archives, select the *running-config* key from the list. This list is populated with all of the keys from the configuration files archived for the selected device(s). To view all possible archives, select *All*.



Tip: To limit audit results to a device's most recently archived configuration file for a particular key, select **Latest Archive Only**.

- 6 After you have specified the appropriate audit criteria, click **Audit** to verify the policy. Results from the audit are displayed in the Audit Results section of the dialog:
 - The either successful or failed **Audit Result**.
 - The **Device Name** of the device by which the policy was audited.
 - Any relevant **Message** regarding the policy audit. For example, the number of archives that failed against the policy.



Tip: Select an audit result, then click **View** to see the details for that result.

- 7 Click **Close** to exit the dialog.

CHAPTER 12

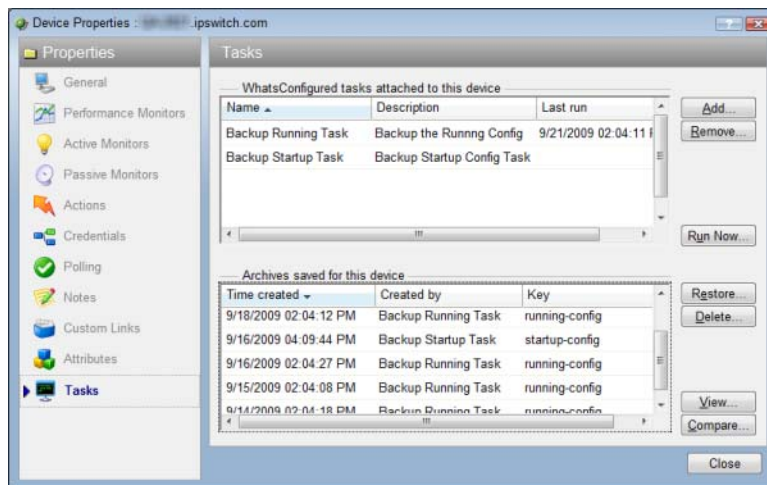
About Device Properties - Tasks

In This Chapter

Using Device Properties - Tasks.....	54
Viewing a configuration file.....	55
Comparing two configuration files	55
Restoring a device to an archived configuration	55
Deleting a configuration file from a device	55
Running a scheduled task immediately	56
Removing tasks from a device	56

Using Device Properties - Tasks

The Device Properties - Tasks dialog displays task and archived device data.



From this dialog you can add or remove device tasks, run a task immediately, restore a device to a previously archived configuration file, or delete an archived configuration. In addition to these management capabilities, the Device Properties - Tasks dialog allows you to view and compare archived configuration files by accessing the WhatsConfigured Diff Viewer.

Viewing a configuration file

To view an archived configuration file for a device:

- 1 In the Device List or Map View, right-click a device, then click **Properties**. The Device Properties dialog appears.
- 2 Click **Tasks**. The Tasks section of Device Properties appears.
- 3 Under **Archives saved for this device**, select an archived config file, then click **View**. The View Configuration Archive dialog appears.

Comparing two configuration files

To compare two device config files:

- 1 In the Device List or Map View, right-click a device, then click **Properties**. The Device Properties dialog appears.
- 2 Click **Tasks**. The Tasks section of Device Properties appears.
- 3 Under **Archives saved for this device**, select two configuration files, then click **Compare**. The WhatsConfigured Diff Viewer appears.

Restoring a device to an archived configuration

To restore a device to an archived configuration:

- 1 In the Device List or Map View, right-click a device, then click **Properties**. The Device Properties dialog appears.
- 2 Click **Tasks**. The Tasks section of Device Properties appears.
- 3 Under **Configuration archives saved for this device**, select a configuration, then click **Restore** to restore the device to the selected configuration.

Deleting a configuration file from a device

To delete an archived configuration file from a device:

- 1 In the Device List or Map View, right-click a device, then click **Properties**. The Device Properties dialog appears.
- 2 Click **Tasks**. The Tasks section of Device Properties appears.
- 3 Under **Archives saved for this device**, select an archived config file, then click **Delete**.

Running a scheduled task immediately

You can run a task on demand from either the Device Properties - Tasks dialog, or from the WhatsConfigured Task Library.



Note: If you run the task from Device Properties the task only runs for that specific device. If you run the task from the Task Library, the task runs for any device to which it is assigned.

To run a task immediately from Device Properties:

- 1 In the Device List or Map View, right-click a device, then click **Properties**. The Device Properties dialog appears.
- 2 Click **Tasks**. The Tasks section of Device Properties appears.
- 3 Under **Tasks attached to this device**, select a task, then click **Run Now** to perform the selected task immediately.

To run a task immediately from the Task Library:

- 1 On the WhatsUp Gold console, select **Configure > WhatsConfigured Task Library**. The WhatsConfigured Task Library appears.
- 2 Select the scheduled task that you would like to run at this time, then click **Run Now**.



Tip: To view the task's results, see the WhatsConfigured Task Log.

Removing tasks from a device

You can remove a task attached to a device from the Device Properties - Tasks dialog.

To remove a task from a device:

- 1 In the Device List or Map View, right-click a device, then click **Properties**. The Device Properties dialog appears.
- 2 Click **Tasks**. The Tasks section of Device Properties appears.
- 3 Under **Tasks attached to this device**, select a task, then click **Remove** to delete the task from this device.

CHAPTER 13

Using Alert Center with WhatsConfigured

In This Chapter

Assigning an Alert Center threshold to a task..... 57

Assigning an Alert Center threshold to a task

If you have WhatsUp Gold installed, you can assign an Alert Center threshold to a task to notify you on task activities. You do this from the Threshold tab of the WhatsConfigured Task dialog, accessed from the WhatsUp web interface.



Note: You cannot assign Alert Center thresholds to WhatsConfigured tasks from the WhatsConfigured Network Viewer.

The screenshot shows the 'Edit WhatsConfigured Task' dialog box with the 'Threshold' tab selected. The 'Name' field contains 'Backup Running Task' and the 'Description' field contains 'Backup the Running Config'. The 'Script' dropdown is set to 'Backup Running Config' with an 'Add Script' button next to it. The 'Threshold' tab is active, showing a list of thresholds. The first threshold is 'WCF AC 2', which is enabled. Below the list, the 'Alert Center Threshold' section shows three checked options: 'Detects configuration changes on a device', 'Fails to run for a device', and 'Successfully runs for device'. The 'Alert Center Notification Policy' dropdown is at the bottom. 'OK', 'Cancel', and 'Help' buttons are on the right.

Edit WhatsConfigured Task

Name: Backup Running Task

Description: Backup the Running Config

Script: Backup Running Config

Threshold

☒ Enable this threshold

Threshold name: WCF AC 2

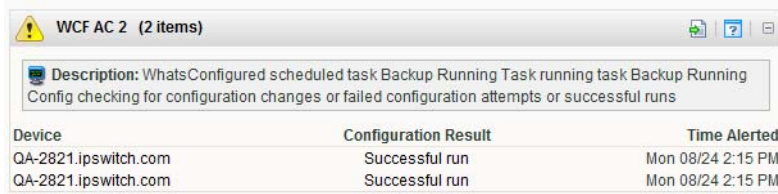
Alert Center Threshold

This threshold will alert when this task

- ☒ Detects configuration changes on a device
- ☒ Fails to run for a device
- ☒ Successfully runs for device

Alert Center Notification Policy

After you have assigned a threshold to a task, a custom threshold workspace report for the task threshold is displayed on the Alert Center tab.



Device	Configuration Result	Time Alerted
QA-2821.ipswitch.com	Successful run	Mon 08/24 2:15 PM
QA-2821.ipswitch.com	Successful run	Mon 08/24 2:15 PM

To assign a threshold to a WhatsConfigured task:

- 1 On the WhatsConfigured console, go to **Libraries > Task Library**. The Task Library appears.
- 2 Do one of the following:
 - Click **Add** to configure a new task. The Select Task type dialog appears.
 - Select *Schedulable Task*, then click **OK**. The New Task dialog appears.
 - or -
 - Select an existing scheduled task, then click **Edit**. The Edit Task dialog appears.
- 3 Select the **Threshold** tab. The threshold tab appears.
- 4 Select **Enable this threshold** to enable and configure the threshold options.
- 5 Enter a **Name** for the threshold.
- 6 Select to have the **Threshold** alert when the task **Detects configuration changes on a device**, if the task matches any of the selected conditions:
 - **Detects a successful execution of a task on a device**
 - **Fails to run for a device**
 - **Successfully runs for a device**
 - **Fails this policy**



Note: If you do not see the appropriate policy, or if the list is empty, browse (...) to the Policy Library to configure a new policy.

- 7 Select the **Notification** policy you would like WhatsConfigured to use to notify you when the threshold is met. If the list is empty or you want to configure a new notification policy, browse (...) to the Configuration Policy Library.

Using Archive Search

In This Chapter

About Archive Search	59
Performing an archive search	59

About Archive Search

The Archive Search feature allows you to search the content of device configuration archives. A configuration archive is any device output captured when running a configuration task or script. When a configuration script is run, the output from one or more commands may be captured and stored in a user or system specified key. The output is saved to the device using the key name and the time-stamp as a look-up key. The archive is persisted with the device in the discovery *.dis* file.

Performing an archive search

To perform an archive search:

- 1 Go to the Archive Search dialog:
From WhatsConfigured, select **Tools > Archive Search**. The Archive Search dialog appears.
- 2 Click **Add**. The Select Device dialog appears.
- 3 Select the device(s) for which you want to perform an archive search, then click **OK**.
- 4 Specify the Search Criteria:
 - Select an **Archive Key** for which to refine search results. For example, to view running config archives, select the *running-config* key from the list. This list is populated with all of the keys from the archived configuration files for the selected device(s). To view all possible archives, select *All*.
 - To view only the latest archives for the selected device(s), select **Lastest Archive Only**.
 - Enter a **Search Pattern** for which the search should attempt to find in the archived config files. This can be a phrase or regular expression.
 - Select **Regular Expression** for the contents of search pattern to be interpreted as a regular expression.

- If the contents of the search pattern are case insensitive, select **Ignore Case**.



Tip: Select a device, then click **Remove** to delete it from the list.

5 Click **Search**. The dialog displays the following Search Results in the bottom half of the dialog:

- The **Archive Key** under which the file was saved in the database.
- The **Device** for which the config file was saved.
- The time at which the configuration file was created (**Time Created**).
- The name of the configuration task for which the file was collected (**Created by**).



Tip: Select an archive file, then click **View** to see the specific archived file.

CHAPTER 15

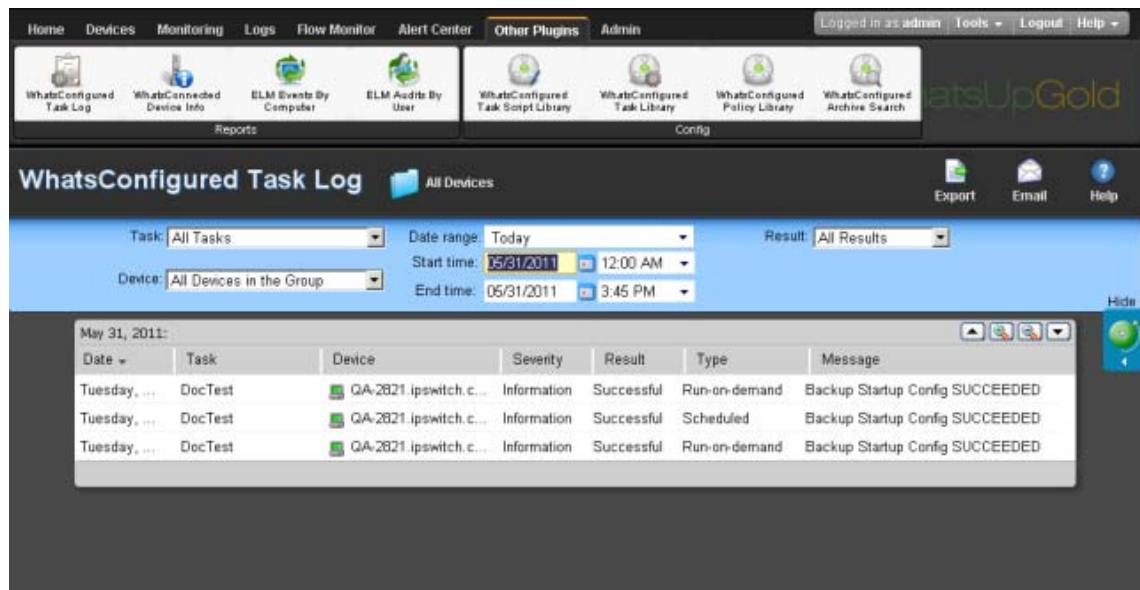
Using WhatsConfigured reports

In This Chapter

About the Configuration Task Log 61

About the Configuration Task Log

The Configuration Task Log displays log messages generated by WhatsConfigured tasks.



Report body

- **Date** displays the date the task ran.
- **Task** displays the name of the specific task.
- **Device** displays the network device for which the task ran.
- **Result** displays the outcome of the task.
- **Message** displays the log message that generated according to the task's result.

Filtering the report

Date range

Use the date/time picker at the top of the report to specify a date range and time.

Task

Use the **Task** list to select a specific task for which to view report data. This list is populated with scheduled tasks currently configured in the Scheduled Task Library.

Device Filter

Use the **Device Filter** list to select a specific network device for which to view report data. You can view data for all devices in the group.

Result

Use the **Result** list to select a specific result for which to view report data. You can choose to view data for all results.

Using the WhatsUp Services Controller

In This Chapter

About the WhatsUp Services Controller	63
Stopping the WhatsConfigured Service.....	64
Starting the WhatsConfigured Service	64
Restarting the WhatsConfigured Service.....	65
Stopping the TFTP server	65
Starting the TFTP server	65
Restarting the TFTP server.....	66

About the WhatsUp Services Controller

The WhatsUp Gold Services Controller application (`NMServiceManager.exe`) provides a single user interface to manage all Ipswitch WhatsUp Gold services. WhatsUp Gold services controller includes services that you can start, stop, or restart:



Note: Some services are optional. If the associated product is not licensed and enabled you may not be able to start and stop the service with the WhatsUp Services Controller dialog (Ipswitch Services Control Manager). Your license file determines whether or not you can access a plug-in. To update your license to purchase WhatsUp Gold Flow Monitor, VoIP plug-in, WhatsConnected, or WhatsConfigured, visit the *MyIpswitch portal* (<http://www.myipswitch.com>).

- Polling Engine (`nm-service.exe`)
- Flow Collector (`bwcollector.net.exe`)
- Alert Center (`alertcenterservice.exe`)
- Trivial File Transfer Protocol Server (`TFTPservice.exe`)
- Whats Configured (`networkconfigservice.exe`)
- Discovery (`discoveryservice.exe`)
- Failover Manager (`nmfailover.exe`)
- API (`nmapi.exe`)

- Whats Connected Data Service (`networkviewerdataservice.exe`)
- Whats Virtual Service (`whatsvirtualservice.exe`)

This application communicates with the Ipswitch Service Control Manager service (`ServiceControlManager.exe`) to issue start, stops, and restarts to the services used by WhatsUp Gold and its plug-in applications.

The following information is displayed in the WhatsUp Services Controller dialog:

- **Description.** Lists the description of the WhatsUp service, as gathered by the Ipswitch Service Control Manager service.
- **Process Name.** Lists the WhatsUp process .exe as listed in the Windows Task Manager Process tab.
- **Status.** Lists the current state of the service.

Stopping the WhatsConfigured Service

To stop the WhatsConfigured Service:

- 1 Go to the WhatsUp Services Controller dialog.
 - From the console, select **Tools > Services Manager**. The WhatsUp Services Controller dialog appears.
 - or -
- 2 From the the Programs menu, click **Ipswitch WhatsUp Gold > Utilities > Service Manager**. The WhatsUp Services Controller dialog appears.
- 3 In the WhatsUp Service Controller, select the **WhatsConfigured service** by clicking its service **Description**.
- 4 Click **Stop**.

Starting the WhatsConfigured Service

To start the WhatsConfigured Service:

- 1 Go to the WhatsUp Services Controller dialog.
 - From the console, select **Tools > Services Manager**. The WhatsUp Services Controller dialog appears.
 - or -
- 2 From the the Programs menu, click **Ipswitch WhatsUp Gold > Utilities > Service Manager**. The WhatsUp Services Controller dialog appears.
- 3 In the WhatsUp Service Controller, select the **WhatsConfigured service** by clicking its service **Description**.
- 4 Click **Start**.

Restarting the WhatsConfigured Service

To restart the WhatsConfigured Service:

- 1 Go to the WhatsUp Services Controller dialog.
 - From the console, select **Tools > Services Manager**. The WhatsUp Services Controller dialog appears.
 - or -
- 2 From the the Programs menu, click **Ipswitch WhatsUp Gold > Utilities > Service Manager**. The WhatsUp Services Controller dialog appears.
- 3 In the WhatsUp Service Controller, select the **WhatsConfigured service** by clicking its service **Description**.
- 4 Click **Restart**.

Stopping the TFTP server

To stop the TFTP server:

- 1 Go to the WhatsUp Services Controller dialog.
 - From the console, select **Tools > Services Manager**. The WhatsUp Services Controller dialog appears.
 - or -
- 2 From the the Programs menu, click **Ipswitch WhatsUp Gold > Utilities > Service Manager**. The WhatsUp Services Controller dialog appears.
- 3 In the WhatsUp Service Controller, select the **Trivial File Transfer Protocol server** (TFTP server) by clicking its service **Description**.
- 4 Click **Stop**.

Starting the TFTP server

To start the TFTP server:

- 1 Go to the WhatsUp Services Controller dialog.
 - From the console, select **Tools > Services Manager**. The WhatsUp Services Controller dialog appears.
 - or -
- 2 From the the Programs menu, click **Ipswitch WhatsUp Gold > Utilities > Service Manager**. The WhatsUp Services Controller dialog appears.
- 3 In the WhatsUp Service Controller, select the **Trivial File Transfer Protocol server** (TFTP server) by clicking its service **Description**.
- 4 Click **Start**.

Restarting the TFTP server

To restart the TFTP server:

- 1 Go to the WhatsUp Services Controller dialog.
 - From the console, select **Tools > Services Manager**. The WhatsUp Services Controller dialog appears.
- or -
- 2 From the the Programs menu, click **Ipswitch WhatsUp Gold > Utilities > Service Manager**. The WhatsUp Services Controller dialog appears.
- 3 In the WhatsUp Service Controller, select the **Trivial File Transfer Protocol server** (TFTP server) by clicking its service **Description**.
- 4 Click **Restart**.

Copyright notice

©1991-2011 Ipswitch, Inc. All rights reserved.

This document, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. Except as permitted by such license, no part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the expressed prior written consent of Ipswitch, Inc.

The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Ipswitch, Inc. While every effort has been made to assure the accuracy of the information contained herein, Ipswitch, Inc. assumes no responsibility for errors or omissions. Ipswitch, Inc., also assumes no liability for damages resulting from the use of the information contained in this document.

IMail, the IMail logo, WhatsUp, the WhatsUp Gold logo, WS_FTP, the WS_FTP logos, Ipswitch, and the Ipswitch logo are trademarks of Ipswitch, Inc. Other products and their brands or company names, are or may be trademarks or registered trademarks, and are the property of their respective companies.

This document was published on Tuesday, May 31, 2011 at 16:15.