

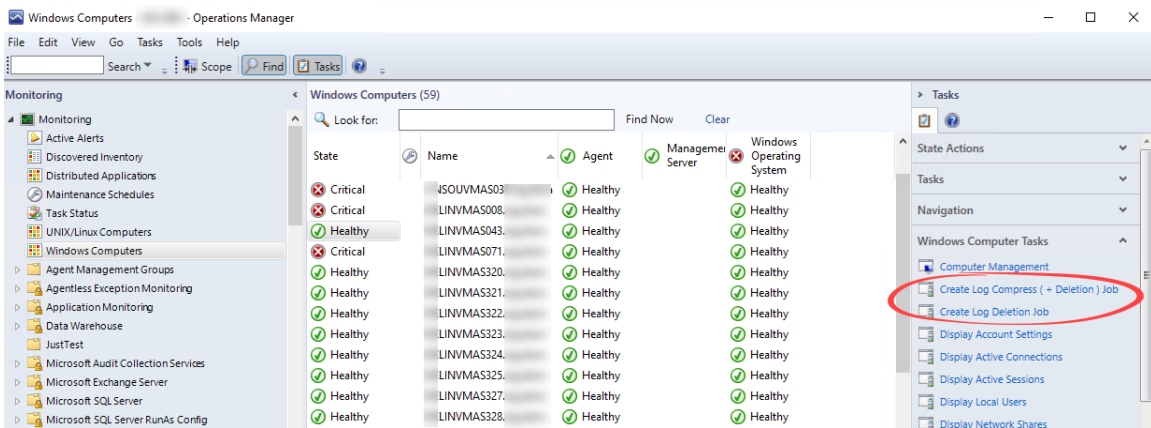
# SCOM – Agent Task – Create Log Deletion Job

## Summary

‘Create Log Deletion Job’ is a Management Pack that contains two SCOM – Agent Tasks to tame logfile growth.

- **Create Log Deletion Job** creates a scheduled task that deletes log files older than N days on the monitored computer.
- **Create Log Compress ( + Deletion ) Job** creates a scheduled task that compresses files older than N days and optionally also deletes the compressed files older than M days on the monitored computer.

It works on SCOM 2012 R2 and later.



## Table of Contents

Summary.....	1
Introduction.....	2
Realization .....	2
Usage.....	3
Checking the result.....	7
Result for Create Log Compress Job.....	7
Result for Create Log Compress ( + Deletion) Job .....	8

## Introduction

Many applications and services which run on a server are creating logs. Some are doing it by default and some others are configured to do so by an Administrator or Developer.

To prevent that logs occupy the complete disk space, either the concerned developer implemented a routine which rotates the files or the Administrator added a scheduled script to do so.

This Management Pack provides a convenient way to create a job for scheduled log handling directly from the SCOM console.

## Realization

This Management Pack consists of two PowerShell scripts which create the scheduled tasks and a PowerShell script that is launched by it on the monitored computer.

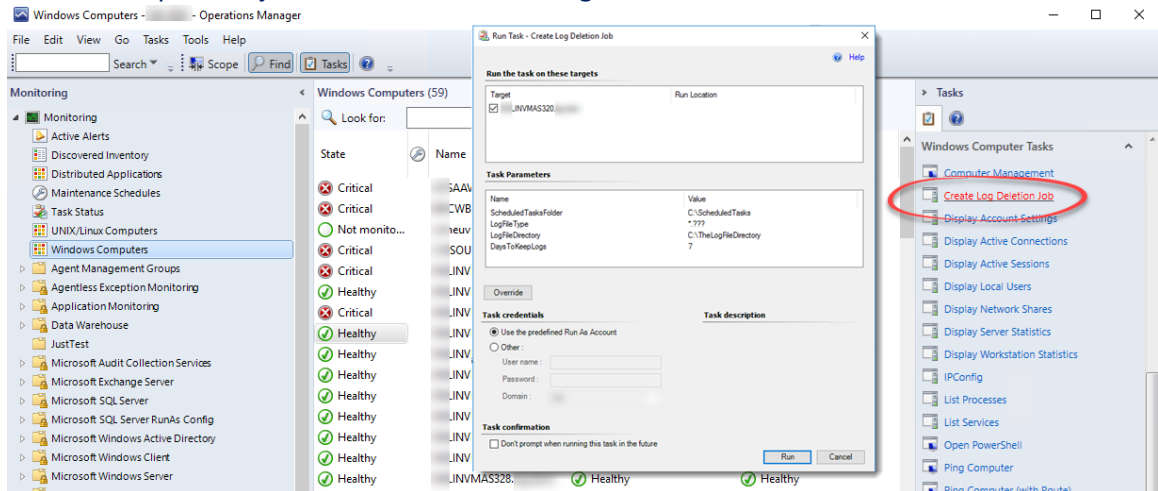
The scripts are embedded in the Management Pack file.

- The **\*.mp** file is the sealed version which offers the Administrators the usual experience. Importing it is the suggested way for this MP.
- The **\*.xml** file is the unsealed and targets Administrators which know the insights of the management pack. All parameters and the script are editable here.

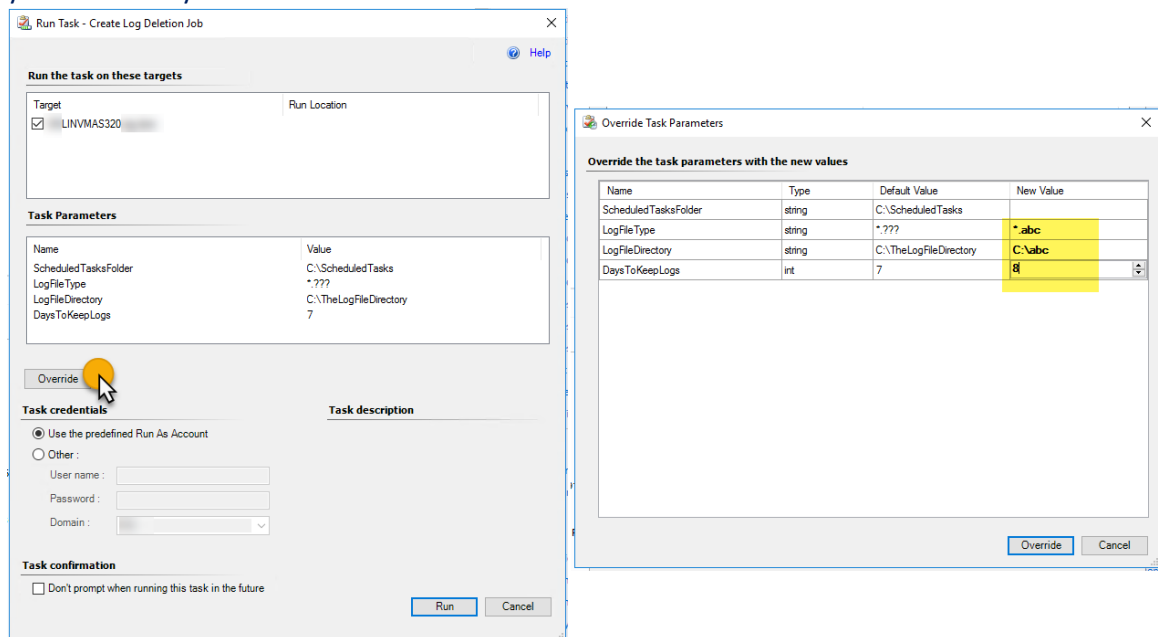
The script logs warnings, errors and informational messages directly in the Task – Window. Additionally, information is readable in the Operations Manager log file (eventvwr) on the computer that is chosen for job creation.

## Usage

- Download and Import the Management Pack from [URL].
- In the SCOM Console, 'Monitoring pane', click on 'Windows Computer' for instance.
- On the right, in the 'Tasks' pane below Windows Computer Tasks you can find **Create Log Deletion Job**
- Select a computer object and click on 'Create Log Deletion Job'.



- Click on the Override button to change the values according your requirement. E.g. the logs are in 'C:\abc', the extension of the logfile is \*.abc and you like to keep only logs your than 8 days:



- Set the credentials that should be used to run the task on the remote machine. – The account need to have administrative permissions on the target machine. Confirm with 'Run'. (The scheduled task on the remote machine will run as SYSTEM).

Run Task - Create Log Deletion Job

Run the task on these targets

Target	Run Location
<input checked="" type="checkbox"/> LINVNAS320	

Task Parameters

Name	Value
ScheduledTasksFolder	C:\ScheduledTasks
LogFile Type	*.abc
LogFileDirectory	C:\abc
DaysToKeepLogs	8

Override

Task credentials

Use the predefined Run As Account

Other :

User name : admin

Password : ••••••••

Domain : msft

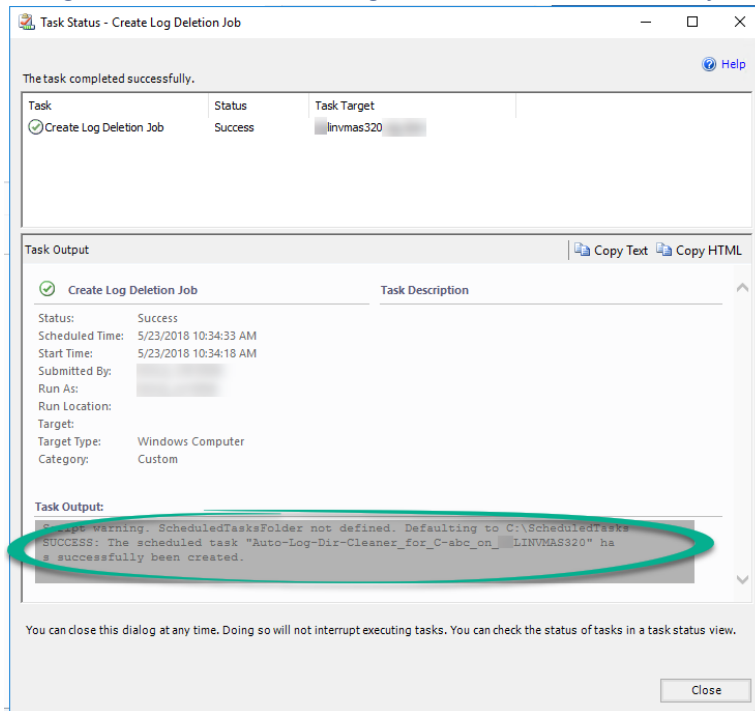
Task description

Task confirmation

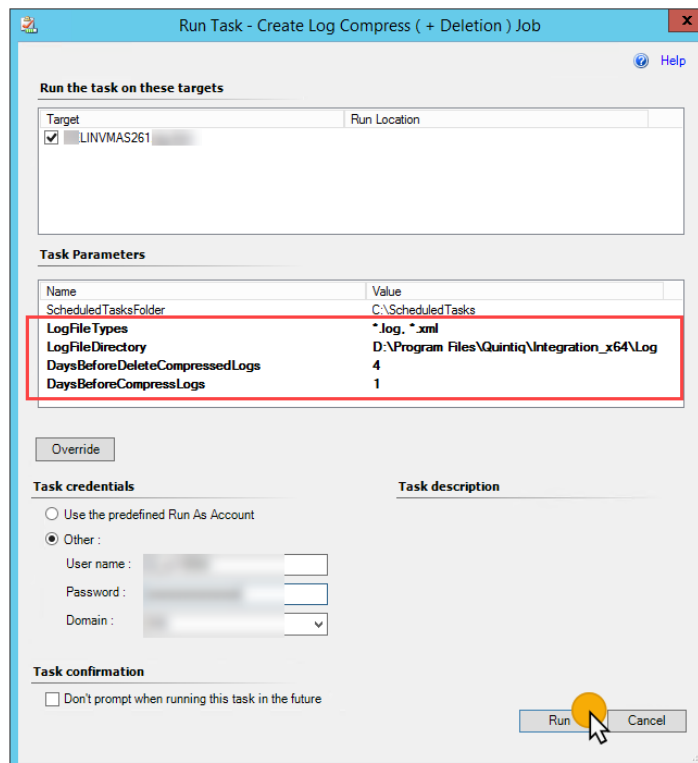
Don't prompt when running this task in the future

Run Cancel

- If all goes fine a success message will confirm that the task job was created:



- The task **Create Log Compress ( + Deletion ) Job** works exactly in the same way. It provides an parameter to specify from when logs shall be compressed and a parameter to specify how long to keep the compressed logs.



*Note: Set Delete **BeforeDeleteCompressedLogs** to **0** to avoid deletion.*

Depending on the .net framework version of the monitored computer the logs are either compressed via NTFS (< = Windows Server 2008 R2) or via ZIP.

## Checking the result

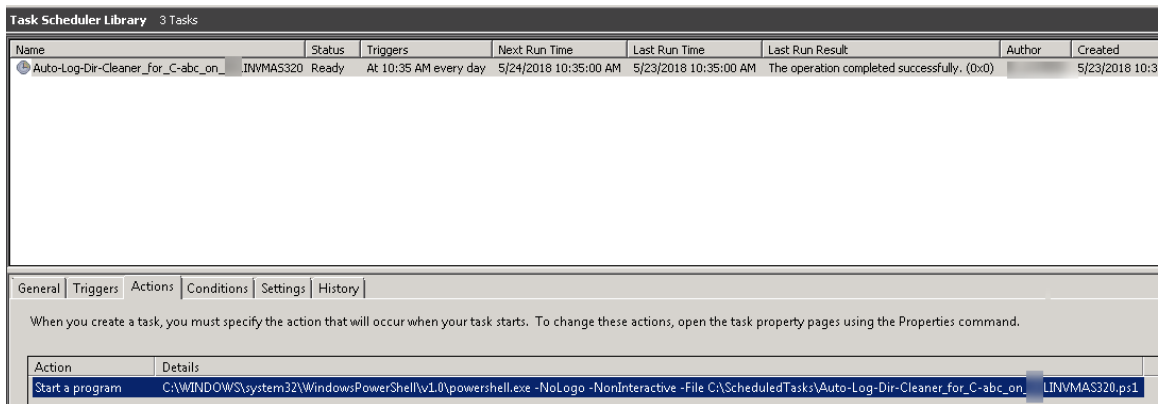
On the target machine then, a PowerShell script will be available in 'C:\ScheduledTasks' if the value was not changed to something different.

### Result for **Create Log Compress Job**

The PowerShell script which is handling the log deletion is shown in the picture below.

```
$now = Get-Date
Get-ChildItem -Path "C:\abc\*" -Include *.abc -ErrorAction SilentlyContinue |
  Where-Object { (New-TimeSpan -start $_.LastWriteTime -end ($now)).TotalDays -gt 8 } | Remove-Item -Force
```

A scheduled task is created which runs the script daily.



Note: The scheduled task runs as 'SYSTEM'.

## Result for Create Log Compress (+ Deletion) Job

The PowerShell script which is handling the log compression (+ deletion) is shown in the picture below.

```
$TimeStamp = Get-Date -Format 'yyyy-MM-dd_hh-mm'
$now = Get-Date

$StepLogFolder = "D:\Program Files\Quintiq\Integration_m04\Log" + "_CompressLogsOn_" + $TimeStamp

if (Test-Path -Path $StepLogFolder) {
    $now = 'No action required, folder exists.'
} else {
    mkdir $StepLogFolder
}

$logDirectoryParent = $StepLogFolder.SubString(0,$StepLogFolder.LastIndexOf('\')) + '\'

try {
    $targetLogZip = "D:\Program Files\Quintiq\Integration_m04\Log" + "_CompressLogsOn_" + $TimeStamp + ".zip"
    Add-Type -AssemblyName 'System.IO.Compression.FileSystem'
    Get-Childitem -Path "D:\Program Files\Quintiq\Integration_m04\Log\*" -Include *.log *.xml -ErrorAction SilentlyContinue | Where-Object { ((New-TimeSpan -start $_.LastWriteTime -end ($now)).TotalDays -gt 1) } | ForEach-Object {
        Move-Item -Path $_.FullName -Destination $StepLogFolder
    }
    [System.IO.Compression.ZipFile]::CreateFromDirectory($StepLogFolder,$targetLogZip)
    Get-Childitem -Path $StepLogFolder -Recurse | ForEach-Object {
        Remove-Item -Path $_.FullName
    }
    Remove-Item -Path $StepLogFolder
} catch {
    if ($?) {
        Get-Childitem -Path $logDirectoryParent -Filter 'Logs_CompressLogsOn_*.zip' -ErrorAction SilentlyContinue | Where-Object { ((New-TimeSpan -start $_.LastWriteTime -end ($now)).TotalDays -gt 4) } | Remove-Item -Force
    }
} catch {
    Get-Childitem -Path "D:\Program Files\Quintiq\Integration_m04\Log\*" -Include *.log *.xml -ErrorAction SilentlyContinue | Where-Object { ((New-TimeSpan -start $_.LastWriteTime -end ($now)).TotalDays -gt 1) -and ($_.Attributes -notmatch 'Compressed') } | ForEach-Object {
        Move-Item -Path $_.FullName -Destination $StepLogFolder
    }
    & COMPACT /I /Q /C /S:$StepLogFolder
} catch {
    if ($?) {
        Get-Childitem -Path "$logDirectoryParent\Logs_CompressLogsOn_*" -Recurse -ErrorAction SilentlyContinue | Where-Object { ((New-TimeSpan -start $_.LastWriteTime -end ($now)).TotalDays -gt 4) -and ($_.Attributes -match 'Compressed') } | Remove-Item -Force
        Get-Childitem -Path "$logDirectoryParent\Logs_CompressLogsOn_*" -ErrorAction SilentlyContinue | Where-Object { ((New-TimeSpan -start $_.LastWriteTime -end ($now)).TotalDays -gt 4) -and ($_.Attributes -match 'Compressed') } | Remove-Item -Force
    }
}
```

A scheduled task is created which runs the script daily.