Contacting BMC Software

You can access the BMC Software Web site at http://www.bmc.com. From this Web site, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

<table>
<thead>
<tr>
<th>United States and Canada</th>
</tr>
</thead>
</table>
| **Address** | BMC Software, Inc.  
2101 CityWest Blvd. Houston TX  
77042-2827 |
| **Telephone** | 1 (713) 918 8800 or  
1 (800) 841 2031 (Toll Free) |

Copyright 2015 BMC Software, Inc. or licensors, as an unpublished work. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc.

All other trademarks belong to their respective companies.

BMC Software considers information included in this documentation to be proprietary and confidential. Your use of this information is subject to the terms and conditions of the applicable End User License Agreement for the product and the proprietary and restricted rights notices included in this documentation.

Restricted Rights Legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC Software, Inc., 2101 CityWest Blvd., Houston, TX 77042-2827, USA. Any contract notices should be sent to this address.
Customer Support

You can obtain technical support by using the Support page on the BMC Software Web site or by contacting Customer Support by telephone or e-mail.

Support Web Site

You can obtain technical support from BMC Software 24 hours a day, 7 days a week at http://www.bmc.com/support_home. From this Web site, you can:

- Read overviews about support services and programs that BMC Software offers
- Find the most current information about BMC Software products
- Search a database for problems similar to yours and possible solutions
- Order or download product documentation
- Report a problem or ask a question
- Subscribe to receive e-mail notices when new product versions are released
- Find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

You can also access product documents and search the Knowledge Base for help with an issue at http://www.sentrysoftware.com

Support by Telephone or E-mail

In the United States and Canada, if you need technical support and do not have access to the Web, call 800 537 1813. Outside the United States and Canada, please contact your local support center for assistance. To find telephone and email contact information for the BMC Software support center that services your location, refer to the Contact Customer Support section of the Support page on the BMC Software Web site at http://www.bmc.com/support_home.
# Table of Contents

Product Hierarchy..................................................................................................................... 17
Credentials Management............................................................................................................ 19
String Search, Numeric Value Extraction and Value Map......................................................... 20
Thresholds and Alerts Actions.................................................................................................. 22
Requirements.......................................................................................................................... 25
Getting the BMC Software Installation utility.......................................................................... 26
Packages................................................................................................................................ 26
Extracting the Setup Files......................................................................................................... 27
Installing Monitoring Studio KM for PATROL.......................................................................... 30
Uninstall Monitoring Studio KM for PATROL........................................................................... 41
Integrating with BMC Portal...................................................................................................... 49
Integrating with BMC TrueSight OM........................................................................................ 50
1. Creating a Group.................................................................................................................. 55
2. Configuring a Host to Monitor.............................................................................................. 58
3. Specifying the Monitors Used............................................................................................... 65
   Monitoring Processes............................................................................................................. 67
      (Optional) Adding a Monitor Group.................................................................................... 67
      1. Specifying the Process to be Monitored......................................................................... 69
         Selecting the Process to Monitor from a List...................................................................... 70
         Specifying Search Criteria............................................................................................... 73
         Providing the PID file path............................................................................................... 75
      2. Selecting Parameters........................................................................................................ 76
      3. Configuring the Monitor Settings.................................................................................... 79
      4. Configuring the Process Cache Refreshing Frequency.................................................... 80
Running Command Lines.......................................................................................................... 81
Performing a Database Query..................................................................................................... 90
   Performing a Query on a Microsoft SQL Server Database...................................................... 91
   Performing a Query on a MySQL Server Database.................................................................. 93
   Performing a Query on an Oracle Database Server.................................................................. 95
   Performing a Query on a PostgreSQL Database....................................................................... 97
   Performing a Query on an Other Database............................................................................. 99
Monitoring Files....................................................................................................................... 101
### Monitoring a File System
- Monitoring a File System.................................104
- Monitoring Folders...........................................106
- Leveraging Values from Other KMs' Parameters.......................109
- Monitoring SNMP Agents & Traps ................................113
  - Polling SNMP Agents........................................113
  - Listening for SNMP Traps..................................118
- Performing Web Requests....................................122
- Monitoring Windows Event Logs.................................132
- Monitoring Windows Performance Counters.......................136
- Monitoring a Windows Service................................138
- Running PSL Commands......................................142
- Executing WBEM Queries.....................................144
- Executing WMI Queries........................................146
- Executing Nagios Plugins ..................................148
  - Importing a Nagios Configuration.........................152

### Cloning a Host
- Cloning a Host..................................................155

### Processing a Monitor Result Output
- 4. Processing a Monitor Result Output..........................157
  - 4.1 (Optional) Pre-Processing a Monitor Output..................158
  - 4.2 Splitting the Monitor Output..............................172
    - Dynamic Object Templates................................176
  - 4.3 Analyzing the Monitor Output..............................177
    - Searching for a Specific String..........................177
    - Extracting Numeric Values.................................182
    - Mapping Values..............................................188

### Executing a Basic Nagios Plugin
- Executing a Basic Nagios Plugin.................................198

### Parsing an XML Log File
- Parsing an XML Log File......................................204

### Acknowledging Alerts
- Acknowledging Alerts..........................................214
  - Acknowledge Alerts.........................................215
  - Acknowledge all and Reset..................................216
  - Reset CollectionErrorCount................................217

### Configuring Java Settings
- Configuring Java Settings.....................................218

### Configuring the SMTP Server
- Configuring the SMTP Server................................221

### Configuring the Proxy Settings
- Configuring the Proxy Settings...............................222

### Configuring Thresholds
- Configuring Thresholds.......................................223
  - Specifying the Threshold Mechanism Mode..................224
  - Setting or Modifying Alert Thresholds.......................225
Visualizing a Monitor ed File Content
Visualizing a Host Windows Event Log
Visualizing SNM P Traps
Setting the Discover y Inter val
Specifying Aler t Actions
Refr eshing Par am eter s
Temporarily Suspend the Monitoring of an Object
M odifying a Gr oup Display Nam e and/or  ID
For cing the Classic M ode
Editing a M onitor ed Object
Copy, Cut and Paste
I’m U nable to Poll an SN MP Agent (Getting a Warning)
Monitoring Studio Reads My L og File E ntirely
Can I Monitor the Processor Time U sage Made by a Windows Service?
Although SN MP Trap Listening Seems to Work, No Trap Matches My Criteria
How Do I Know Which Version of Monitoring Studio I Am Running?
How Do I Know Which Version of Monitoring Studio I Am Running?
Monitoring Studio Is Unable to Listen to SNM P Traps
Can I Monitor the Processor Time Usage Made by a Windows Service?
Monitoring Studio Reads My Log File Entirely
I’m Unable to Poll an SN M P Agent (Getting a Warning)
I’m Unable to See the Monitoring Studio Icons in PATROL Central
Infinite Loop Reported in the PATROL Agent Log
Is It Possible to Monitor the CPU/Memory Usage of a Process Tree?
Monitoring Studio Fails to Authenticate to My Web Server
Monitoring Studio Does Not Follow an HTTP Redirection (Web)
Monitoring Studio Is Not Able to Post a Web Form to My Web Server

Application Classes and Parameters

SEN_MS_COMMANDLINE
SEN_MS_DBQUERY
SEN_MS_DYNAMIC
SEN_MS_DYNAMIC_CONT
SEN_MS_DYNAMICDISCOVERY
SEN_MS_DYNAMICNUMBER
SEN_MS_DYNAMICSTRING
SEN_MS_DYNAMICVALUEMAP
SEN_MS_FILE
SEN_MS_FILESYSTEM
SEN_MS_FOLDER
SEN_MS_FORMULA
SEN_MS_GROUP
SEN_MS_HOST
SEN_MS_MAIN
SEN_MS_MONITORGROUP
SEN_MS_NAGIOSPERF
SEN_MS_NAGIOSPLUGIN
SEN_MS_NUMBER
SEN_MS_PROCESS
SEN_MS_PSLCOMMAND
SEN_MS_SNMPOLLING
SEN_MS_SNMPTRAP
SEN_MS_STRING
SEN_MS_TRANSFORM
SEN_MS_VALUEMAP
SEN_MS_WBEMQUERY
SEN_MS_WEBREQUEST
SEN_MS_WINEVENT
SEN_MS_WINPERF
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEN_MS_WINSERVICE</td>
<td>375</td>
</tr>
<tr>
<td>SEN_MS_WMIQUERY</td>
<td>377</td>
</tr>
<tr>
<td>Managing Baselines and Key Performance Indicators</td>
<td>379</td>
</tr>
<tr>
<td>Configuration Variables</td>
<td>380</td>
</tr>
<tr>
<td>Global Configuration Variables</td>
<td>380</td>
</tr>
<tr>
<td>Specific Configuration Variables</td>
<td>382</td>
</tr>
<tr>
<td>About...</td>
<td>385</td>
</tr>
<tr>
<td>About HTTP Authentication</td>
<td>385</td>
</tr>
<tr>
<td>About Internal Identifiers (IDs)</td>
<td>386</td>
</tr>
<tr>
<td>About Performance Counters</td>
<td>390</td>
</tr>
<tr>
<td>About Processes</td>
<td>390</td>
</tr>
<tr>
<td>About Regular Expressions</td>
<td>392</td>
</tr>
<tr>
<td>About WMI</td>
<td>394</td>
</tr>
</tbody>
</table>
Release Notes for v9.4.01
What's New

- **SWSY-3312**: Group macros can be used in Host monitors and Group & Host macros can be used in all other monitors (ex: %{SEN_HOSTNAME}).
- **SWSY-3399**: Support for Java 9.

Changes and Improvements

- **SWSY-3311** - Database Queries: The Database Query: Microsoft SQL Server uses Microsoft's JDBC driver version 6.2.1 instead of the JTDS driver. SSL Encryption and Authentication options are therefore deprecated as they are handled automatically by the driver.
- **SWSY-3304**: When converting XML files to CSV, quote, tab, space or whitespace characters are removed from the property name and value.
- **SWSY-3392**: It is now possible to connect and authenticate against a remote system with SSH and keyboard-interactive authentication (no actual interaction with the user is required while connecting to the system).
- **SWSY-3398**: The performance of the Process monitor is improved.

Fixed Issues

- **SWSY-3227**: Because the MySQL's JDBC driver version leveraged by version 9.4 was not stable enough, Monitoring Studio could not connect to MySQL Server databases running on Linux.
- **SWSY-3235**: Nagios Performance objects with long names were not monitored correctly and an error was displayed in the System Output Window (SOW).
- **SWSY-3245** - Process Monitoring: No process was found when using a regular expression to search for the command line that launched the process.
- **SWSY-3252**: A configuration file generated from v9.0.00 might not correctly import into v9.4.00, leaving invalid credentials, alert actions, etc.
- **SWSY-3289** - String Searches: On Windows, empty lines that contained the “\r” character (carriage return) were not properly handled by Monitoring Studio, which could lead to false alerts.
Overview
What is Monitoring Studio?

Monitoring Studio KM for PATROL is the essential tool for all IT administrators who want to fulfill their custom monitoring needs. This toolbox enables you to monitor almost any technology (application, server, device, etc.) for which there is no out-of-the-box monitoring solution. In a few clicks, you can cover up to 100% of your technologies in your BMC monitoring environment. Compatible with Linux/UNIX and Windows, using Monitoring Studio is a simple and effective way to rapidly deploy the monitoring of custom technologies without any coding. It also has the benefits of a “standard” solution: maintenance, updates, patches, etc. to further respond to growing technological needs for specific business-critical technologies.

With Monitoring Studio, it is possible to set up the monitoring of any technology, customize the way a notification is performed for an alert, or specify a recovery action to run when a problem occurs.

How Does it Work?

Monitoring Studio KM for PATROL is a Knowledge Module (KM) for BMC PATROL. Therefore, it must be installed on the following components of the PATROL framework:

- the PATROL Agents
- the PATROL Consoles
- the PATROL Console Servers (PATROL 7 framework only)

Monitoring Studio can be configured to monitor any technology locally and remotely.
The diagram below shows how Monitoring Studio integrates within your BMC framework and shows interaction between all the components that compose your monitored environment.
What to Monitor with Monitoring Studio?

Monitoring Studio KM for PATROL offers a large choice of Monitors easily configurable to monitor system elements, to query information about the targeted technology, and extract the relevant key metrics.

Monitoring Studio allows you to monitor:

- Command lines
- Files
- File systems
- Folders
- Multi-parameter formulas
- Processes
- PSL commands
- SNMP queries
- SNMP traps
- SQL queries
- WBEM queries
- Web requests
- Windows events
- Windows performance counters
- Windows services
- WMI queries

Monitoring Studio then allows you to analyze the results of some of the above Monitors by searching for strings and regular expressions, extracting numeric values, mapping values, and creating instances dynamically to represent the components of the monitored technology.

Then, administrators can easily define alert thresholds and alert actions to detect and react to critical conditions.

Monitoring Studio also offers integration capabilities with Nagios plugins and allows you to import an existing Nagios configuration from a Nagios server.
General Concepts
Introduction

This section provides general information about the following important concepts of Monitoring Studio KM for PATROL:

- **Product Hierarchy**
- **Credentials Management**
- **String Search, Numeric Value Extraction and Value Map**
- **Thresholds and Alerts Actions**
Product Hierarchy

The architecture of Monitoring Studio KM for PATROL has been designed in a logical manner to help you easily and intuitively manage all your technologies (application, server, device, etc.) and to reflect as accurately as possible the genuine hierarchy of the environment you want to monitor. The Monitoring Studio hierarchy has been established as follows:

1. Monitoring Studio
   The Monitoring Studio instance (SEN_MS_MAIN) is at the highest level of the hierarchy. It appears once the KM has been installed on a PATROL Agent and is loaded in a BMC PATROL Console. It is the starting point of the Monitoring Studio configuration. Groups are created from this instance.

2. Group
   A Group instance (SEN_MS_GROUP) is the main container which corresponds to the technology you need to monitor (application, server, device, etc.). Groups are displayed at the same level as the Monitoring Studio instance and contain one or several Hosts.

3. Host
   A Host instance (SEN_MS_HOST) includes information about the target server where the technology is running (hostname, system type, credentials list, SNMP information and host availability check information, etc.). Hosts are arranged under the Group instance and contain either Monitor Groups or Monitors.

4. Monitor Group
   A Monitor Group instance (SEN_MS_MONITORGROUP) is an optional container that you can use to group monitors. Monitor Groups are created under a Host and contain Monitors.
Monitor instances are at the lowest level of the hierarchy and corresponds to the diverse monitoring tools available in Monitoring Studio. Monitors are either grouped directly under a Host or under a Monitor Group. They can also be referred to as “monitored objects”.

Here is the complete list of Monitors:

- Command Line (SEN_MS_COMMANDLINE)
- Database Query (SEN_MS_DBQUERY)
- File (SEN_MS_FILE)
- File System (SEN_MS_FILESYSTEM)
- Folder (SEN_MS_FOLDER)
- Multi-Parameter Formula (SEN_MS_FORMULA)
- Nagios Performance Data (SEN_MS_NAGIOSPERF)
- Nagios Plugin (SEN_MS_NAGIOSPLUGIN)
- Performance Counter (SEN_MS_WINPERF)
- Process (SEN_MS_PROCESS)
- PSL Command (SEN_MS_PSLCOMMAND)
- SNMP Polling (SEN_MS_SNMPOLLING)
- SNMP Trap (SEN_MS_SNMPTRAP)
- WBEM Query (SEN_MS_WBEMQUERY)
- Web Request (SEN_MS_WEBREQUEST)
- Windows Event (SEN_MS_WINEVENT)
- Windows Service (SEN_MS_WINSERVICE)
- WMI Query (SEN_MS_WMIQUERY)
- Text Pre-Processing (SEN_MS_TRANSFORM)
- String Search (SEN_MS_STRING)
- Numeric Value Extraction (SEN_MS_NUMBER)
- Value Map (SEN_MS_VALUemap)
- Dynamic Instances (SEN_MS_DYNAMIC)
Credentials Management

Monitoring a system thoroughly requires connecting to it and accessing some of its resources, which itself requires to be properly authenticated with this system. This is even more true for a system monitored remotely.

In Monitoring Studio, the credentials required to access a monitored system are stored and managed at the host level. For each monitored host, you will be able to define the "System Credentials", to access standard system resources on this host. You will also be able to define additional credentials that may be required to access specific resources (a database, a Web application, etc.)

System Credentials

System Credentials are provided at the Host level and can be shared by several monitors. When the system credentials are changed, users only need to modify the credentials once and for all the monitors to inherit the changes.

The following Monitors can only use system credentials:

- File System (SEN_MS_FILESYSTEM)
- Process (SEN_MS_PROCESS)
- Windows Event (SEN_MS_WINEVENT)
- Windows Performance Counter (SEN_MS_WINPERF)
- Windows Service (SEN_MS_WINSERVICE)
- WMI Query (SEN_MS_WMQUERY)

If the system credentials are not specified when monitoring a localhost, Monitoring Studio will use the PATROL Agent's default account information. For remote monitoring, System Credentials are mandatory for the monitors listed above; failing to provide this authentication information will prevent the monitors from collecting any data.

Specific Credentials

Some technologies may require additional privileges to allow access to their data. In this case, Monitoring Studio enables users to provide specific credentials that will only apply to the monitor instance they relate to.

The following monitors may require specific credentials. If specific credentials are not provided, the solution will automatically use the system credentials provided at the Host level:
Some technologies report their bad health through repeated error messages or critical numbers which are hidden deep in log files, output of commands, database, Web page, etc. Monitoring Studio KM for PATROL is capable of parsing all this data by means of String Search, Numeric Value Extraction features, and Value Map to detect the source of potential problems and alert you when they occur.

String Search, Numeric Value Extraction and Value Map are available for the following monitors:
• Command Line (SEN_MS_COMMANDLINE)
• Database Query (SEN_MS_DBQUERY)
• Dynamic Instances (SEN_MS_DYNAMIC)
• File (SEN_MS_FILE)
• Multi-Parameter Formula (SEN_MS_FORMULA)
• Nagios Plugin (SEN_MS_NAGIOSPLUGIN)
• PSL Command (SEN_MS_PSLCOMMAND)
• SNMP Polling (SEN_MS_SNMP_POLLING)
• Text Pre-Processing (SEN_MS_TRANSFORM)
• WBEM Query (SEN_MS_WBEMQUERY)
• Web Request (SEN_MS_WEB_REQUEST)
• WMI Query (SEN_MS_WMQUERY)

Searching for a Specific String

The String Search (SEN_MS_STRING instance) enables you to run fast and powerful searches for strings on some of the monitored objects that you previously configured (flat or log files, the output of a Web request or a database query, OID content, etc). You can then decide to trigger an alert and send a notification to your operators when the string specified is found or not found in the return output of monitored instances.

Extracting Numeric Values

The Numeric Value Extraction (SEN_MS_NUMBER instance) enables you to extract numeric values from a text output (the output of a command, a Web page, the result of a SQL query, or WBEM query, etc). All you need to do is indicate how to find the numeric values within the monitored object and Monitoring Studio will extract and report them as graphs in the Console.

Mapping Values

The Value Map (SEN_MS_VALUE_MAP instance) enables you to extract values from result outputs generated by other Monitors and map up to 25 of these values to user-defined status that match your specific needs. Based on these statuses, it becomes easy to apply thresholds and have Monitoring Studio trigger PATROL standard alerts directly in your PATROL environment.
Thresholds and Alerts Actions

What Are Thresholds?

Thresholds are used to define acceptable values for the parameters collected. You can set thresholds for each parameter of any monitor. Breaching these thresholds will trigger an alert.

Monitoring Studio KM for PATROL offers default thresholds that can be very easily tailored to your needs through wizards. Monitoring Studio accepts up to two alert thresholds in addition to defining the acceptable range values.

What Are Alert Actions?

When a threshold is breached, an alert is raised. Alert Actions enable you to configure specific actions to be executed when an alert is raised by a parameter. With Alert Actions, it is possible to customize the way a problem notification is performed, or specify a recovery action to be run when a problem occurs.
Preparing for Installation
This chapter provides information about the tasks that must be performed before you start installing Monitoring Studio KM for PATROL. Make sure to carefully read the following sections requirements:

- **Requirements**
- **Getting the BMC Software Installation Utility**
- **Packages**
- **Extracting the Setup Files**
Requirements

BMC Framework

Console Systems

- PATROL Consoles

Optional

- BMC ProactiveNet
- BMC TrueSight Operations Management

PATROL Agent

Monitoring Studio KM for PATROL supports any version of the PATROL Agent.

JAVA

Monitoring Studio KM for PATROL requires Java 1.8.00 or higher and a Java Runtime Environment (JRE) to be installed on the same system that runs the PATROL Agent.

You can download the Java Runtime Environment along with the KM on the Sentry Software Web site.
Getting the BMC Software Installation Utility

In order to install Monitoring Studio KM for PATROL, you need the latest version of the BMC Software Installation Utility. You can download the latest Installation Utility from the Monitoring Studio KM for PATROL page on the Sentry Software Web site.

Choose the appropriate package depending on the platform on which you plan to install Monitoring Studio KM for PATROL:

- For UNIX/Linux systems: **ins_ALL_<version>.tar**
- For Windows systems: **ins_WINDOWS_<version>.zip**

The packages are valid for all the PATROL components: Agent, Console, Console Server, etc.

Place the Installation Utility package in the same folder as the Monitoring Studio KM for PATROL packages.

Packages

The Monitoring Studio KM for PATROL packages are available on the Sentry Software Web site.

There are two packages, each of which can be used to install the KM on every PATROL component (Agent, Console, etc.) according to the platform you are using:

- For UNIX/Linux systems: **monitoring-studio-<version>-thorium.tar**
- For Windows systems: **monitoring-studio-<version>-thorium.zip**

Java Runtime Environment Package

Java Runtime Environment version **1.8.00** or higher is required for the KM to operate properly.

A package with a suitable Java Runtime Environment, designed for use with the BMC Software Installation Utility, is made available to download on the Sentry Software Web site. It can be installed along with the Monitoring Studio package. The KM will be able to leverage either this instance of the Java Runtime Environment or another installed separately on the system.

⚠️ The Monitoring Studio KM for PATROL packages are designed for use with the BMC Software Installation Utility (formerly known as Thorium). They need to be merged with the latest available version of the BMC Software Installation Utility. Please see instructions.
Extracting the Setup Files

Like most Knowledge Modules for PATROL, you need to install Monitoring Studio on the following components of your PATROL architecture:

- Every managed system with a PATROL Agent
- Every PATROL Console (Classic)
- The PATROL Console Server
- PATROL Central – Web Edition

On Windows

The packages for Windows are to be extracted in the following order:

1. Unzip the BMC Installation Utility: `ins_WINDOWS_<version>.zip`
2. Unzip the Monitoring Studio KM package in the same folder: `monitoring-studio-<version>-thorium.zip`
3. Unzip the `senjre_<version>_WINDOWS.zip` package if you need to install a Java Runtime Environment program

The content of the packages merges into a subfolder named `bmc_products`.

⚠️ Windows may ask for your confirmation to overwrite files and folders. This is normal behavior and you can accept all overwrite confirmation requests.

On UNIX/Linux

The packages for UNIX/Linux are to be extracted with the `tar` utility in the following order:

1. Untar the BMC Installation Utility: `tar xvf ins_ALL_<version>.tar`
2. Untar the Monitoring Studio KM package named: `tar xvf monitoring-studio-<version>-thorium.tar`

The files are extracted into a sub-directory named `bmc_products`. 
Installing/Uninstalling the KM
Introduction

This section describes how to install and uninstall Monitoring Studio KM for PATROL on any PATROL component (Agents, Consoles, Console Server, etc.). As the setup is based on the BMC Software Installation Utility, this procedure applies to all operating systems supported by Monitoring Studio KM for PATROL. Although not covered by this section, the installation through the Distribution Server is fully supported. Please refer to the Distribution Server documentation for a detailed description of the deployment procedure.
Installing Monitoring Studio KM for PATROL

Installing Monitoring Studio KM for PATROL is an automated process managed by a wizard. The wizard goes through the necessary steps to properly install Monitoring Studio and all files associated with it. You are simply prompted for the product's folder location and the product/components to install. Browse to the `bmc_products` folder where the files have been extracted (both Monitoring Studio and the Common Installer).

- On Windows: Launch the `setup.exe` program.
- On UNIX: Launch the `setup.sh` script

On the more recent versions of Windows, you may get this security alert dialog: Windows detected that the BMC Software Installation Utility is listening on the 50001 port.

⚠️ An automatic migration is performed from Monitoring Studio KM for PATROL v9.x to the current version, but not from v8.x.

To install Monitoring Studio KM for PATROL

1. The welcome screen of the installer appears. Click **Next** to continue.
2. Read the license agreement, click the **Accept** option and then click **Next** to continue.
3. Select **Install products on this computer now** and click **Next** to continue.
4. Specify the **BMC Software Products** folder. See the BMC Software documentation for more information about the BMC Software Products folder. Click **Next** to continue.
5. Select Default. Installing Monitoring Studio KM for PATROL does not require any customization. Click Next to continue.

6. Select one or several of the following options:
   - Managed System to install Monitoring Studio KM for PATROL on a PATROL Agent.
   - Console Systems to install Monitoring Studio KM for PATROL on a PATROL Classic Console.
   - Common services to install Monitoring Studio KM for PATROL on the Console Server or on PATROL Central – Web Edition.
   Click Next to continue.
Select System Roles

Select one or more roles for the computer on which you are planning to install products. The products available for installation as displayed on the Select Products and Components to Install screen are based on the system role that you select. If you do not select a role or if you select all roles, all products will be displayed.

- **Managed System**
  The role of this system is to host software that manages resources on the system. Examples of products installed on these systems include the PATROL Agent and PATROL Knowledge Modules.

- **Console Systems**
  The role of this system is to host user desktop applications such as consoles, user interfaces, viewers, and browsers. Examples of products installed on these systems include the PATROL Classic Console, PATROL Central - Microsoft Windows Edition, and the PATROL Migration Tools. You must install Knowledge Modules on the PATROL Classic Console system (in addition to installing them on managed systems). Please refer to the PATROL Installation Release Notes for more information.

- **Common Services**
  The role of this system is to host services that are shared among managed systems and application systems.
7. Check the **Monitoring Studio KM for PATROL** box to install the KM (it should be selected by default). Click **Next** to continue. Select the **Java Runtime Environment** program if you want to install it along with the KM (this option is only provided when you have previously downloaded the [package](#)). Click **Next** to continue.
8. Review the installation parameters and click **Start Install** to launch the installation procedure.
9. The setup program displays the actions performed and indicates the percentage of completion. Once the installation is complete, click **Next** to view the installation results.
10. The wizard displays the installation results. Click **View Log** to display a detailed log of the installation procedure. Click **Next** and then **Finish** to exit the setup program.

The following products and components have been successfully installed on your machine.
To review the details of the installation, click **View Log**.

Monitoring Studio KM for PATROL

Log File Location:
C:\Users\cyril.SENTRY\AppData\Roaming\BMCinstall\PC-CPWIN7-1483374160.log

Please click **Next** to continue.
11. Click the **Finish** button to complete and close the installation wizard.

THANK YOU!

Thank you for using the Installation Utility.

To exit the Installation Utility, click **Finish**.
Uninstall Monitoring Studio KM for PATROL

Monitoring Studio uninstall is an automated process managed by a wizard. The wizard goes through the necessary steps to remove Monitoring Studio and all files associated with it. You are simply prompted for the product's folder location and the product/components to uninstall.

To uninstall Monitoring Studio KM for PATROL
1. Locate the Uninstall folder under the BMC products directory (typically under C:\Program Files\BMC Software, or /opt/bmc):
   - On Windows systems, launch uninstall.exe
   - On UNIX, launch uninstall.sh
2. Specify the BMC Software products folder. See the BMC Software documentation for more information about the BMC Software products folder.
3. To uninstall the KM, select **Monitoring Studio KM for PATROL**.

![Uninstall Wizard — Selecting Products and Components Directory](image_url)
4. Click **Start Uninstall**.
5. A page displays the list of products/components processed and the percentage of completion. Click **Next** to continue.
6. A page displaying SUCCESS indicates that Monitoring Studio KM for PATROL is now uninstalled.
Integrating Monitoring Studio KM for PATROL
Introduction

This section provides detailed information about the integration of Monitoring Studio KM for PATROL into the BMC framework.

- Integrating with BMC Portal
- Integrating with BMC TrueSight Operations Management
Integrating with BMC Portal

An integration component for BMC Portal is released along with the Knowledge Module (KM) in the form of a PAR file. It enables visualization of parameter data of all objects monitored by the KM, in the BMC Portal environment.

⚠️ This integration component is valid on BMC portal v2.8 upwards.

To integrate Monitoring Studio with BMC Portal:

1. Install the KM on top of a PATROL Agent as detailed in the Installation Guide. A PATROL Console will be required to configure the KM
2. In the PATROL Console, make sure the discovery is properly performed and data is collected
3. Upload the PAR file in BMC Portal:
   - Place the SentrySoftware-PM-PATROL-SENMS-<version>.par file in a known location on your file system.
   - Log in to BMC Portal with the superadmin credentials
   - Click on Portal, then under Tasks click Performance Managers
   - Click Upload and then click Browse to open a file selection dialog box
   - Select SentrySoftware-PM-PATROL-SENMS-<version>.par and click Upload
4. Create a new managed element:
   - Log in to BMC Portal with administrator credentials
   - Click the Configure tab > Elements > Add
   - Select Infrastructure Element and click Next
   - Select an RSM to collect data about the infrastructure element and click Next
   - Enter the name of the machine on which the PATROL Agent is running, the Host Name and click Next
   - Select the group to which your PM will belong and click Next
   - Select the PATROL category, the Application Classes to add and click Next
   - Set the Properties and Credentials that the RSM must have to access the specified elements and click Next. The list of thresholds is displayed. You can update these settings without impacting the values set in PATROL
5. Click Finish

⚠️ Thresholds are set in the PATROL Console. To get the last configurations made, click the Refresh PATROL Integration button available in the Elements page.
Integrating with BMC TrueSight Operations Management

No specific integration steps need to be performed. To know how to use Monitoring Studio KM for PATROL in BMC TrueSight Operations Management, please refer to the TrueSight Operations Management - Monitoring Studio documentation.

ℹ️ Monitoring Studio needs to detect a CMA policy with a Monitoring Studio configuration to switch to CMA mode.
Loading Monitoring Studio KM for PATROL
In order to start monitoring any technology (application, server, device, etc.) within your PATROL Console, Monitoring Studio KM for PATROL must be loaded on the appropriate managed system. Loading a Knowledge Module in PATROL Central Operator both loads the Knowledge Module on the managed system and adds the name of the Knowledge Module to your management profile.

Loading Monitoring Studio KM for PATROL on PATROL Classic
1. In the Console menu bar, click File > Load KM.
2. In the Files of Type list, select KM List Files (*.kml).
3. Select SEN_MS.kml.
4. Click Open.

Loading Monitoring Studio KM for PATROL on PATROL Central
1. In the Common Tasks tab of the PATROL Central Operator Taskpad, click the Load Knowledge Modules icon.
2. To start the wizard, click Next.
3. Select the check boxes for the managed systems that you want to load Monitoring Studio KM for PATROL on. Click Next.
4. Select the SEN_MS.kml check box in the Knowledge Module Lists tab.
5. Click Finish.
Configuring Groups, Hosts and Monitors
The configuration of Monitoring Studio KM for PATROL will mostly depend on the type of technology to be monitored. It is however recommended to follow these guidelines:

1. **Create a Group**
2. **Add the Hosts to be monitored**
3. **Specify objects to be monitored**
4. **Specify the information to be searched**

Once configured, the technology (application, server, device, etc.) is displayed in the PATROL Console. This technology is then monitored just like any other standard component of the system (hardware, OS, middleware, etc.).

⚠️ The entire configuration of Monitoring Studio is stored in the agent configuration tree (under /SENTRY/STUDIO).
1. Creating a Group

A Group is designed to facilitate the management of the monitored technologies (application, server, device, etc.). Typically, the name of a Group is the name of the technology you wish to monitor, for example 'My Company's Web Site'. A Group is displayed at the same level as the Monitoring Studio instance and contains one or several Hosts.

To create a group

1. In the PATROL Console, right-click the Monitoring Studio icon and select KM Commands > New Group...

2. Name your Group:
   - **Group name**: Enter the name of the group to be created.
   - (Optional) **Description**: Describe the group to be monitored.
   - (Optional) **Owner contact details**: Enter the name, E-mail, phone number, etc. of the person to be alerted or who can provide support if an alerts occurs.

3. Click **Next**.
Group Constants

4. Define the **Group Constants** for the Group. Constants are very useful for monitoring a technology whose properties may change from one system to another. Constants are defined at the **Group** level and can be reused in the **Monitors** related to the group. They facilitate the monitoring across various systems of a technology whose properties may change from one system to another. Here is an example of use:

**Example:** You want to specify the monitoring of a device through a command line interface (CLI). Depending on where the PATROL Agent is running, this CLI may be installed in a different directory. To avoid editing every Monitor using this CLI to change the path, you create a Group Constant `MYCLI_PATH` with the path to the CLI executable. Then you create the "Command Line Analysis" Monitor by specifying, for example: `"%{MYCLI_PATH}" - option1 - option2

⚠️ Please note that Groups Constants are case sensitive and should always be used in uppercase.

Fill in the following fields:

- **Constant name**: Enter the name of the constant. Example: `TECHNOLOGY_PATH`
- **Value**: Set a value for the constant. Example: `/opt/MyTechnology`. If you enter the word "PASSWORD" in a Constant name, the corresponding value will be automatically encrypted once you click the **Finish** button at the last step of the wizard.
- **Required**: Check the box to activate this newly-set Group Constant. The monitoring of the Group and all its dependent objects will be offline if the value of the Group Constant is not set. This ensures that no monitoring operation is performed until the required Group Constants are properly set. This feature is particularly useful if you create a group monitoring template where all the defined Group Constants are empty – in this case, you first import the configuration template, then enter values to all required Group Constants before the monitoring actually starts.
5. Click Next.

6. The new group is defined by an **Internal ID** and a **Display name**. These values, set by default by Monitoring Studio, can however be modified while configuring the group. Modify the fields of your choice:

   - **Internal ID**: PATROL internal identifier of this **Group**. The PATROL internal identifier of every object belonging to this Group (Hosts, Monitors, etc.) will include the internal ID of the Group. It is therefore recommended to keep this ID short.
   - **Display name**: Label that will appear next to the Group in the PATROL Console.
   - **What thresholds do you want to set for the newly created instance?**
     - **I want to use the default thresholds**: Uses the default thresholds set by Monitoring Studio. (see [SEN_MS_GROUP](#))
     - **I want to set custom thresholds**: Allows you to customize the thresholds for all parameters of the instance.
     - **I want to use the default thresholds and customize them**: Sets the default Monitoring Studio thresholds on all parameters and then allows you to customize any/all of them. This is mainly intended to help save time if you wish to customize the thresholds of just one or many parameters for the instance, and leave the default settings for the others.
     - **I do not want to set any thresholds for now**: No thresholds will be set on any parameter of the instance, and as a result no alerts will be triggered. Monitoring Studio will poll the object and return the output of the polling – but will not raise any alerts until you set thresholds.
7. Click **Finish**.

The **Group** instance is automatically created and its icon 📦 is displayed in the PATROL Console. Collected parameters for Group instances are listed in the **SEN_MS_GROUP** chapter.

You can now:

- Add **Hosts** by right-clicking the **Group** instance and selecting **KM Commands > New > Host...**
- Edit your **Group** by right-clicking the **Group** instance and selecting **KM commands > Edit**.
- Remove your **Group** by right-clicking the **Group** instance and selecting **KM commands > Delete**.

### 2. Configuring a Host to Monitor

Once you have created the **Group** associated to your targeted technology, the second action you need to perform is to provide Monitoring Studio with the information related to the **Host** on which the technology you want to monitor is running. A **Host** instance contains all the information about the server where the technology to monitor is running (hostname, system type, connection credentials list, SNMP information, host availability check information, etc.). **Hosts** are grouped under the **Group** instance and contain **Monitors**.

#### To create a host

1. In the PATROL Console, right-click the **Group icon** and select **KM Commands > New > Host...**
2. Provide the name of the Host:
- **Hostname/IP address/FQDN**: Enter the Hostname, IP address or Fully Qualified Domain Name of the host on which the technology you wish to monitor is running.
- *(Optional) Display name*: Enter the name that will be displayed in the console for this host.
- *(Optional) Description*: Enter a unique description that will be used to describe this host.

   \*The description is particularly useful to easily identify the host when importing a configuration through the Import Configuration wizard.\*

- **System type**: Select the type of the operating system that is running on the host that will be monitored (Windows, UNIX/Linux, or Other). Select Other for any host that will not behave as a regular Linux or UNIX system would. For example, while a network device is likely to run a customized version of Linux of BSD, its restricted shell will not allow Monitoring Studio to use standard UNIX commands to perform the monitoring.

   \*Monitoring Studio requires you to specify the type of the operating system to avoid a costly identification phase and to offer more robust monitoring options.\*

3. **Create a separate device in TrueSight OM**: Check this option if you want the remote monitored host to appear as a separate device in TrueSight OM.

4. Click **Next**.

### System Credentials

![Adding the Hosts to be Monitored - Credentials](image)
5. **(Optional) System Credentials**: Enter the **Username** and **Password** required to establish a connection with the targeted host. These credentials are used for **all system-related monitoring tasks** to collect data.

6. **(Optional) Associated OpenSSH Private Key File Path**: When monitoring remote hosts running UNIX, Linux or other types of operating systems that support SSH authentication key file, you may need to provide an OpenSSH private key file to establish a secured connection with the remote host. Enter the path of the OpenSSH private key file you wish to use to establish this connection and enter the optional PassPhrase in the **Password** field.

   ! **The Private Key File should exists on the PATROL Agent node.**

7. **SNMP version used by <Managed System>**: Select the SNMP version to be used when performing SNMP monitoring and/or an SNMP availability check. Select **None** if no SNMP agent is running on the host, or if you do not plan to use SNMP to monitor this host.

8. Click **Next**.

9. If you have previously selected SNMP version **1**, specify:
   - the **Community** to be used.
   - the **Port** number (default: 161).

10. If you have previously selected SNMP version **2c**, specify:
    - the **Community** to be used.
    - the **Port** number (default: 161).
    - the number of seconds Monitoring Studio KM for PATROL will wait for the completion of the SNMP polling (default: 120 seconds). This timeout must be long enough to complete the polling of an entire SNMP table.
11. If you have previously selected SNMP version 3, specify:
   - the **Username** to be used to perform the SNMP query.
   - the **Authentication protocol** to be used to authenticate the SNMP v3 messages. Possible options are: **None**, **MD5**, **SHA**.
   - the **Authentication password** to be used to authenticate the SNMP v3 messages.
   - the **Privacy protocol** to be used to encrypt SNMP v3 messages. Possible options are: **None**, **AES**, **DES**.
   - the **Privacy password** associated with the privacy protocol.
   - the **Context name** accessible to the SNMP entity.
   - the **Port** number (default: 161).
   - the number of seconds Monitoring Studio will wait for the completion of the SNMP polling (default: 120 seconds). This timeout must be long enough to complete the polling of an entire SNMP table.
12. Click **Next**.
13. (Optional) Select the method Monitoring Studio should use to test the availability of the monitored host:

- **Ping check**: Select this option to have Monitoring Studio ping the monitored host. To be successful, the targeted host must respond to at least 1 ping command out of 4, during each collection cycle.

- **Hostname Resolution Check**: Select this option to check that Monitoring Studio can resolve the hostname of the monitored host to an IP address.

- **SNMP check**: Select this option to have Monitoring Studio check the monitored host availability via an SNMP session. For the SNMP availability check to be successful, the targeted host must respond to a "GETNEXT" request on either the OID 1.3.6.1 or the OID 1.3.6.1.4.1.

- **(Windows OS only) WMI check**: Select this option to have Monitoring Studio check the monitored host availability via the WMI protocol.

- **(Remote UNIX/Linux or other OS only) SSH check**: Select this option to have Monitoring Studio check the targeted host availability (UNIX, Linux, or other systems) via the SSH protocol. For the SSH availability check to be successful, Monitoring Studio must be able to connect to the host using the provided System Credentials.

- **TCP Port check**: Select this option to have Monitoring Studio check the targeted host availability via a TCP connection. You must provide the port number the protocol must use to access the host.

- **(Optional) Signature Files Check**: Click to open the Signature Files Check dialog box. If at least one of the signature files is present on the targeted host, the check is considered successful:
2. Configuring a Host to Monitor

Adding the Hosts to be Monitored - Signature Files

- Provide the name and full path of the signature file on the targeted host.
- Click **Accept**.

⚠️ **Wildcards are not supported in signature files**

- **Disable related monitors when the host is unreachable**: Select this option if you wish Monitoring Studio to disable the monitoring of all Monitors attached to the Host.

⚠️ **The monitoring will automatically resume when the host is once again available.**

- **What thresholds should be set on the newly created object?** Mode to be used to set alert thresholds:
  - **I want to use the default thresholds**: Uses the default thresholds set by Monitoring Studio
  - **I want to set custom thresholds**: Allows you to customize the thresholds for all parameters of the instance
  - **I want to use the default thresholds and customize them**: Sets the default Monitoring Studio thresholds on certain parameters and then allows you to customize any/all of them. This is mainly intended to help save time if you wish to customize the thresholds of just one of many parameters for the instance, and leave the default settings for the others.
  - **I do not want to set any thresholds for now**: No thresholds will be set on any parameter of the instance, and as a result no alerts will be triggered. Monitoring Studio will poll the object and return the output of the polling – but will not raise any alerts until you set thresholds.

14. Click **Finish**.

The Host icon 🏛 is automatically created and displayed in the PATROL Console. Collected parameters for Host instances are listed in the **SEN_MS_HOST** chapter. You can now specify the Monitors to use or first create a Monitor Group.
3. Specifying the Monitors Used

Once you have created the Group Icon, the Host(s) and the Monitor Group(s) (Optional) related to your technology, you need to specify what must be monitored for this technology.

You can choose among a large choice of tools that can be easily configured to monitor:

If Monitoring Studio is running on a Windows system:

<table>
<thead>
<tr>
<th>Monitors</th>
<th>Monitored Hosts</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Windows</td>
</tr>
<tr>
<td>Command Lines</td>
<td>X</td>
</tr>
<tr>
<td>Database Queries</td>
<td>X</td>
</tr>
<tr>
<td>File Systems</td>
<td>X</td>
</tr>
<tr>
<td>Files</td>
<td>X</td>
</tr>
<tr>
<td>Folders</td>
<td>X</td>
</tr>
<tr>
<td>Multi-Parameter Formulas</td>
<td>X</td>
</tr>
<tr>
<td>Nagios Plugins</td>
<td>X</td>
</tr>
<tr>
<td>Processes</td>
<td>X</td>
</tr>
<tr>
<td>PSL Commands</td>
<td>X</td>
</tr>
<tr>
<td>SNMP Agents</td>
<td>X</td>
</tr>
<tr>
<td>SNMP Traps</td>
<td>X</td>
</tr>
<tr>
<td>WBEM Queries</td>
<td>X</td>
</tr>
<tr>
<td>Web Requests</td>
<td>X</td>
</tr>
<tr>
<td>Windows Events</td>
<td>X</td>
</tr>
<tr>
<td>Windows Performance Counters</td>
<td>X</td>
</tr>
<tr>
<td>Windows Services</td>
<td>X</td>
</tr>
<tr>
<td>WMI Queries</td>
<td>X</td>
</tr>
</tbody>
</table>
If Monitoring Studio is running on a UNIX/Linux system:

<table>
<thead>
<tr>
<th>Monitors</th>
<th>Monitored Hosts</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Windows</td>
</tr>
<tr>
<td>Command Lines</td>
<td></td>
</tr>
<tr>
<td>Database Queries</td>
<td></td>
</tr>
<tr>
<td>File Systems</td>
<td></td>
</tr>
<tr>
<td>Files</td>
<td></td>
</tr>
<tr>
<td>Folders</td>
<td></td>
</tr>
<tr>
<td>Multi-Parameter Formulas</td>
<td></td>
</tr>
<tr>
<td>Nagios Plugins</td>
<td></td>
</tr>
<tr>
<td>PSL Commands</td>
<td></td>
</tr>
<tr>
<td>SNMP Agents</td>
<td></td>
</tr>
<tr>
<td>SNMP Traps</td>
<td></td>
</tr>
<tr>
<td>WBEM Queries</td>
<td></td>
</tr>
<tr>
<td>Web Requests</td>
<td></td>
</tr>
<tr>
<td>Windows Events</td>
<td></td>
</tr>
<tr>
<td>Windows Performance Counters</td>
<td></td>
</tr>
<tr>
<td>Windows Services</td>
<td></td>
</tr>
<tr>
<td>WMI Queries</td>
<td></td>
</tr>
</tbody>
</table>
Monitoring Processes

Guarantying optimal server, application, or any technology performance cannot only rely on monitoring system resources such as CPU or memory utilization, storage space availability or temperature levels. The performance of each process underlying critical technologies needs to be accurately evaluated to understand the origin of the load on a system.

The Process Monitor is designed to identify and monitor any process running on the servers of your IT environment, locally or remotely. In a few clicks, set up configuration options to retrieve processes’ status, specify critical parameters to monitor or manage alert thresholds.

To monitor Windows, UNIX, or Linux processes, simply specify one or more of the following criteria:

- The Process Name (as it appears in `ps` or in the Windows Task Manager)
- The Command Line that was used to spawn the process
- The Username the process is run as
- The PID file path that contains the process ID

⚠️ These criteria will be automatically pre-filled when selecting the process to monitor from the current running processes list.

Once you have specified the process to be monitored, simply indicate which parameters should be monitored.

(Optional) Adding a Monitor Group

Once you have added the Hosts to be monitored, you can decide to add an optional container that will be displayed under the Hosts instance in the PATROL tree view: the Monitor Group.

The Monitor Group offers you a higher level of classification. Thanks to this additional container, you can keep Monitors together, organize, and arrange them.
To add a monitor group

1. In the PATROL Console, right-click the Host icon and select KM Commands > New > Monitor Group...

2. Indicate the:
   - PATROL internal identifier (Internal ID field)
   - label to be displayed in the PATROL console (Display name field)

3. Click OK.

Your Monitor Group has been successfully created. You can now specify the Monitors to use.
1. Specifying the Process to be Monitored

The **Process** Monitor allows you to identify the process to be monitored by either:

- [Selecting the process from a list](#)
- [Entering search criteria](#)
- [Providing the PID file path](#)

---

*Specifying the Process to Be Monitored - Three Methods*
Selecting the Process to Monitor from a List

To select the process to be monitored from a list:

1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Process from the drop-down list and click Next.
3. Check Select the process from current process list and click Next. It may take a few seconds for Monitoring Studio to retrieve the list of all the processes currently running on the host.
4. Click Current processes to display the list of all the processes running on the monitored host.
5. Click the process you wish to monitor and click **Accept**.

6. Click **Next**. The process criteria panel is displayed with all information already filled out.

7. Remove or modify any information you do not wish to monitor and click **Next**.

   - **All fields are optional but at least one field must be completed. The process to monitor must match all the criteria if more than one is specified.**
Examples:

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Processes that match the criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example 1</td>
<td></td>
</tr>
<tr>
<td>• Process name MUST BE EXACTLY patrolagent.exe</td>
<td>PatrolAgent.exe</td>
</tr>
<tr>
<td>• Command-line = &lt;nothing&gt;</td>
<td>C:\Patrol\PatrolAgent.exe</td>
</tr>
<tr>
<td>• User = &lt;nothing&gt;</td>
<td>-p 3181</td>
</tr>
<tr>
<td></td>
<td>C:\Patrol\PatrolAgent.exe</td>
</tr>
<tr>
<td>Example 2</td>
<td></td>
</tr>
<tr>
<td>• Process name MUST BE EXACTLY patrolagent.exe</td>
<td>C:\Patrol\PatrolAgent.exe</td>
</tr>
<tr>
<td>• Command-line MUST MATCH THE REGULAR EXPRESSION -[pP] 3181</td>
<td>-p 3181</td>
</tr>
<tr>
<td>• User = &lt;nothing&gt;</td>
<td></td>
</tr>
<tr>
<td>Example 3</td>
<td></td>
</tr>
<tr>
<td>• Process name MUST CONTAIN Pat</td>
<td>PatrolAgent.exe</td>
</tr>
<tr>
<td>• Command-line = &lt;nothing&gt;</td>
<td>PatProcess.exe</td>
</tr>
<tr>
<td>• User = &lt;nothing&gt;</td>
<td>PatrolPerf.exe</td>
</tr>
</tbody>
</table>

⚠️ To monitor all processes of a selected user, fill out the “Process runs as this user” field and leave “Process name” and “Command-line” empty.

8. Select the parameters to be monitored for this process and click Next.
9. Configure the Monitor settings
10. Click Finish.
Specifying Search Criteria

The **Process** Monitor also allows you to enter search criteria to identify the process to be monitored:

1. In the PATROL Console, right-click the **Host** icon and select **KM Commands > New > Monitor...**

2. Select **Process** from the drop-down list and click **Next**.

3. Check **Enter the criteria to select processes** and click **Next**. The following dialog box is displayed:

   ![Entering Search Criteria - Enter criteria to identify and monitor processes](image)

4. Specify your search criteria:
   - (Optional) In the **Process name** section, select a condition option and enter the name of the process you wish to monitor.
   - (Optional) In the **Command line that launched the process** section, select a condition option and enter a **regular expression**. Only processes that have been launched by a command-line matching this regular expression will be monitored.
   - (Optional) In the field **Process runs as this user**, enter the user name the monitored process is running as.
   - Click **Next**.

   **⚠️** All fields are optional but at least one field must be completed. The process to monitor must match all the criteria if more than one is specified.
Examples:

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Processes that match the criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example 1</strong></td>
<td></td>
</tr>
<tr>
<td>• Process name MUST BE EXACTLY patrolagent.exe</td>
<td>PatrolAgent.exe</td>
</tr>
<tr>
<td>• Command-line = &lt;nothing&gt;</td>
<td>C:\Patrol\PatrolAgent.exe</td>
</tr>
<tr>
<td>• User = &lt;nothing&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>C:\Patrol\PatrolAgent.exe</td>
</tr>
<tr>
<td>• PatrolAgent.exe</td>
<td></td>
</tr>
<tr>
<td>• C: \Patrol\PatrolAgent.exe</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example 2</strong></td>
<td></td>
</tr>
<tr>
<td>• Process name MUST BE EXACTLY patrolagent.exe</td>
<td>C:\Patrol\PatrolAgent.exe</td>
</tr>
<tr>
<td>• Command-line MUST MATCH THE REGULAR EXPRESSION -[pP] 3181</td>
<td></td>
</tr>
<tr>
<td>• User = &lt;nothing&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example 3</strong></td>
<td></td>
</tr>
<tr>
<td>• Process name MUST CONTAIN Pat</td>
<td>PatrolAgent.exe</td>
</tr>
<tr>
<td>• Command-line = &lt;nothing&gt;</td>
<td>PatProcess.exe</td>
</tr>
<tr>
<td>• User = &lt;nothing&gt;</td>
<td>PatrolPerf.exe</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

⚠️ To monitor all processes of a selected user, fill out the "User" field and leave "Process name" and "Command-line" empty.

⚠️ Search criteria are case-sensitive on UNIX and Linux.

5. **Select the parameters to be monitored** for this process and click **Next**.
6. **Configure the Monitor settings**
7. Click **Finish**.
Providing the PID file path

Some technologies record their PID (process ID) in a pre-defined file. Monitoring Studio can read the PID from this file and monitor the corresponding process.

1. In the PATROL Console, right-click the Host icon and select KM Commands > New > Monitor...

2. Select Process from the drop-down list and click Next.

3. Check Provide a PID file path and click Next.

4. Provide the PID file path and click Next. At each polling, Monitoring Studio reads this file, retrieves the PID number and checks whether this process PID exists or not. Normally, the process PID is dynamically allocated. The process PID number should be at the very beginning of the file’s content.

5. Select the parameters to be monitored for this process and whether you want to include all child processes. Click Next.

6. Configure the Monitor settings

7. Click Finish.

All processes that match the provided criteria will be monitored and identified with an individual icon in the PATROL Console.
2. Selecting Parameters

Once you have specified a process, you need to select the parameters to be monitored among the following ones:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Displays the number of processes that match the criteria.</td>
<td>Processes</td>
</tr>
<tr>
<td>ChildCount</td>
<td>Displays the number of children of the matching process(es).</td>
<td>Processes</td>
</tr>
<tr>
<td>ProcessorTime</td>
<td>Displays the processor time percentage used by the matching process(es).</td>
<td>Percentage (%)</td>
</tr>
<tr>
<td>WorkingSet (Windows only)</td>
<td>Displays the working set size of the matching process(es).</td>
<td>Megabytes</td>
</tr>
<tr>
<td>PageFileBytes (Windows only)</td>
<td>Displays the page file size used by the matching process(es).</td>
<td>Megabytes</td>
</tr>
<tr>
<td>PrivateBytes (Windows only)</td>
<td>Displays the amount of memory that has been allocated by matching processes and that cannot be shared with others.</td>
<td>Megabytes</td>
</tr>
<tr>
<td>VirtualBytes</td>
<td>Displays the virtual memory used by the matching process(es).</td>
<td>Megabytes</td>
</tr>
<tr>
<td>Status (UNIX/Linux only)</td>
<td>Provides the status of the process according to the selected Status Interpretation.</td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
</tr>
<tr>
<td>PageFaultsPerSec (Windows only)</td>
<td>Displays the number of page faults per second caused by the matching process(es).</td>
<td>Page fault/sec</td>
</tr>
<tr>
<td>HandleCount (Windows only)</td>
<td>Displays the number of handles opened by the matching process(es).</td>
<td>Handles</td>
</tr>
<tr>
<td>ThreadCount (Windows only)</td>
<td>Displays the number of threads of the matching process(es).</td>
<td>Threads</td>
</tr>
</tbody>
</table>

Check the Include child processes box if you want to include all the child processes associated to the defined main process.

When several processes match the provided criteria, the selected parameters will be valued in the PATROL Console (under the single process icon) by summing up the value of the parameters of each matching process.
Status Interpretation (UNIX/Linux)

If you have previously selected the Status parameter, the wizard illustrated below appears to let you define the status interpretation.

![Status Interpretation Wizard]

(Optional) For each of the process status, select the value of the Status parameter: OK, Suspicious or Failed. By default, the overall process status will be interpreted as shown in the table below:

<table>
<thead>
<tr>
<th>Process Status</th>
<th>Interpreted Parameter Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Running</td>
<td>OK</td>
</tr>
<tr>
<td>Sleeping</td>
<td>OK</td>
</tr>
<tr>
<td>Queued</td>
<td>OK</td>
</tr>
<tr>
<td>Waiting</td>
<td>OK</td>
</tr>
<tr>
<td>Intermediate</td>
<td>OK</td>
</tr>
<tr>
<td>Stopped</td>
<td>Suspicious</td>
</tr>
<tr>
<td>Growing</td>
<td>Suspicious</td>
</tr>
<tr>
<td>Unknown</td>
<td>Suspicious</td>
</tr>
<tr>
<td>Terminated</td>
<td>Failed</td>
</tr>
<tr>
<td>Not Running</td>
<td>Failed</td>
</tr>
</tbody>
</table>
The ultimate status reflects the worst status of all selected processes. If the child processes are selected for inclusion, they will also influence the ultimate process status.
3. Configuring the Monitor Settings

Monitoring Studio creates a Monitor instance in the PATROL Console for each configured element. This instance is defined by a name and an internal identifier. These values, set by default by Monitoring Studio, can however be modified while configuring the element:

1. Modify the fields of your choice:
   - **Internal ID**: PATROL internal identifier of the Monitor instance.
   - **Display name**: Label that will be displayed in the PATROL Console for the Monitor instance.
   - **What thresholds should be set on the newly created object?**: Mode to be used to set alert thresholds:
     - [ ] I want to use the default thresholds: Uses the default thresholds set by Monitoring Studio
     - [ ] I want to set custom thresholds: Allows you to customize the thresholds for all parameters of the instance
     - [ ] I want to use the default thresholds and customize them: Sets the default Monitoring Studio thresholds on certain parameters and then allows you to customize any/all of them. This is mainly intended to help save time if you wish to customize the thresholds of just one of many parameters for the instance, and leave the default settings for the others.
     - [ ] I do not want to set any thresholds for now: No thresholds will be set on any parameter of the instance, and as a result no alerts will be triggered. Monitoring Studio will poll the object and return the output of the polling – but will not raise any alerts until you set thresholds.
If you select the “I want to use the default thresholds and customize them” option, the Set Thresholds panel will be prompted when you click Finish (certain parameters for the instance may appear with an asterisk symbol indicating that they already have thresholds). Thresholds can be set or modified at anytime by right-clicking on the instance > KM commands > Thresholds.

2. Click Finish. The corresponding Process instance (<Process: Display Name>) is created in the PATROL Console. The collected parameters for the Process Monitor are listed in the SEN_MS_PROCESS chapter.

4. Configuring the Process Cache Refreshing Frequency

Monitoring Studio relies on a cache mechanism to share the information among the Monitors in order to use as little resources as possible on the target Host and over the network. The cache will be refreshed if one of the Monitors need to collect data (polling interval reached) and if the cache is older than the minimum cache refresh selected.

All process data on the selected Host are cached and shared by all Process Monitors defined for this Host. By default, the process cache is refreshed minimum every 15 seconds. It is however possible to change this minimum cache refresh interval as described in the Setting the Polling Interval section.
Running Command Lines

Even though Monitoring Studio KM for PATROL offers a variety of monitoring methods (Monitors), there might be an in-house script or command that you need to run and analyze on a regular basis to monitor a specific technology.

The Command Line Monitor allows you to trigger the periodic execution of a specified command line on the targeted host. This command can be a shell command, a shell script or an executable file with arguments.

Once the Command Line Monitor is executed and its result displayed in the PATROL Console, you can define String Searches, Numeric Value Extractions and Value Map criteria to retrieve strings or numeric value from the result output.

Monitoring Studio enables you to create Command Line Monitors to execute the OS command/script on the local system where the PATROL Agent KM is installed through the Command Line: Local Execution Monitor; or if the command is run on the remote target, different from the PATROL Agent system, through the Command Line: Remote Execution Monitor.

To run a command line (Local Execution)
1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Command Line: Local Execution from the drop-down list and click Next.
3. **Credentials**: Select the credentials that you want to use to execute the command line:
   - **Agent's Default Account**: Uses the PATROL Agent credentials for localhost monitoring (Default).
   - **System Credentials (default)**: Uses the credentials previously specified when creating the Host.
   - **Add new credentials...**: Lets you provide new credentials for the current command line instance. Refer to the Setting Credentials chapter for detailed information.

4. Enter the command line to execute:
   - **Command line**: Enter the command line or the path to the script that Monitoring Studio will execute. You can use the following macros in the command line that will be executed:
     - %{SEN_TIME:<date-time-format>}: Use this macro to insert the current date and time in the command line. You can specify the format of the date and time string that will be inserted, which follows UNIX `asctime()` format (%Y for year, %m for month, %D for day, %H for hours, %M for minutes, %S for seconds, etc.).
     - %{SEN_LASTTIME:<date-time-format>}: Use this macro to insert the date and time at which the command was last executed. This can be particularly useful when you need to specify a time range for the command, like listing events since the last time we checked. The format is the same as the UNIX `asctime()` format (%Y for year, %m for month, %D for day, %H for hours, %M for minutes, %S for seconds, etc.).
     - %{SEN_SCRIPTPATH:<local-script-path>}: Use this macro to copy a file stored on the PATROL Agent's system to the monitored host before the command is executed. When the command is executed, the macro is replaced by the path to the copied file on the targeted host. This is particularly useful to trigger the execution of scripts that are stored on the PATROL Agent's system without having to install these scripts on each monitored host. This macro is irrelevant when monitoring the localhost.
       - When using the %{SEN_LASTTIME:<date-time-format>} macro, the execution is skipped entirely the first time the Monitor runs (after the PATROL Agent starts). This is to ensure that an actual date and time is inserted with an actual value for this macro.
       - The %{SEN_SCRIPTPATH} macro should provide the script file path on the local Agent system.
       - The %{SEN_SCRIPTPATH} macro does not support local environment variables (e.g., %PATROL_HOME%).

     - %{HOSTNAME}: This macro inserts the hostname of the targeted system, as specified in the host configuration (it may be an IP address, a FQDN or a short name).
     - %{USERNAME}: Use this macro to insert the username of the specified credentials in the command line to be executed.
     - %{PASSWORD}: Use this macro to insert the password of the specified credentials in the command line to be executed. The password is inserted in clear text.
       - Passwords should never be sent in clear text. Passwords in command lines may be visible to non-root users. Use at your own risk.

     - **Command launched once and runs continuously**: Check this option if you want Monitoring Studio to automatically run the command line continuously without having to repeatedly relaunch it manually.
     - **Click Next**.
5. If you have enabled the *Command launched once and runs continuously* option, Monitoring Studio prompts you to confirm your choice:

![Running Command Lines - Confirmation Message](image_url)
- Click Yes, and configure the **Command Termination** settings:

  ![Command Termination Settings](image)

  - **Command line:** ping
    
    In some cases, Monitoring Studio may need to execute a termination command to stop commands running in the background.
    
    **Termination command line:** (optional)
    
    - **Terminate at Agent restart**

  - **(Optional) Termination command line:** Enter a specific command that will be used to stop the execution of the never-ending command. This command runs when the command instance is deleted from the PATROL configuration or when the **Terminate at Agent restart** option is selected. It is required to properly end the execution of the command.
  - **Terminate at Agent restart:** Select this option if you wish the never-ending command to be stopped upon the next PATROL Agent restart.
  - Click **Next**.
6. If you have not enabled the Command launched once and runs continuously option, configure the Command Execution Validation settings:

7. Set the following Command Execution Validation options:
   - **Timeout**: Enter the time in seconds after which the command will be stopped (Default: 30 seconds). If the timeout is reached, the value of the Status parameter will be set to 2 (Failed), indicating that the command failed to execute properly. No further analysis will be performed (String or Numeric Value searches).
   - **(Optional) Recovery Command**: Click this button to type a command similar to a recovery/cleaning action that will be executed when the timeout is reached. The %{SEN_PID} macro can be used to indicate the PID of the command line process being interrupted.

   - **(Optional) Execution is validated when output contains**: Enter the regular expression to be found in the command output that will confirm that the command is successful. The regular expression entered here will be searched in the output of the command. If it is not found, the value of the Status parameter will be set to 2 (Failed), indicating that the
command failed to execute properly. No further analysis will be performed (String and Numeric Value searches).

This option can be particularly useful to ensure that the command has been properly executed and avoid false alerts triggered by the associated String Search/Numeric Value Extraction when an error is encountered during the command execution. For example, if the specified command must print a text banner, you will want to make sure that the text banner is found in the command output instead of an error message).

- **Exit codes below mean the command execution**: Select an execution option (succeeded/failed), to state if the command line was properly executed or not. When one of the exit codes is found or not found, depending on the option selected, the **ExitStatus** parameter of the Command Line monitor is automatically set to 1 (Failed) or 0 (Succeeded) indicating that the command failed or succeed to execute properly.

- **(Optional) Exit codes separated by commas**: Enter one or several exit codes separated by commas. When one of the exit codes is found or not depending on the selected execution option, the **ExitStatus** parameter will be set to 1 (Failed) or 0 (Successful).

The **ExitCode** and **ExitStatus** parameters are deactivated for never-ending commands and commands for which no exit code is specified or when the exit code is ignored. Also, the ExecutionTime parameter is deactivated for never-ending command lines to avoid meaningless alerts (SEN_MS_COMMANDLINE).

- **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the **CollectionErrorCount** parameter of the Group reflect possible alerts triggered upon the Command Line execution. The **CollectionErrorCount** parameter of the Group reports on the collection errors of any Monitor, associated to the Group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.
8. Click **Next**.
9. **Configure the Monitor settings**.
10. Click **Finish**. The corresponding Command Line instance (Command Line: <Display Name>) is created in the PATROL Console. The collected parameters for Command Line Monitors are listed in the `SEN_MS_COMMANDLINE` chapter.

To run a command line (Remote Execution)

3. In the PATROL Console, right-click the **Host** or **Monitor Group** icon and select **KM Commands > New > Monitor...**
4. Select **Command Line: Remote Execution** from the drop-down list and click **Next**.

5. **Credentials**: Select the type of credentials that you want to use to execute the command line:
   - **System Credentials**: Uses the credentials previously specified when creating the Host (Default).
   - **Add new credentials...**: Lets you provide new credentials for the current command line instance. Refer to the Setting Credentials chapter for detailed information.

6. Enter the command line to execute:
   - **Command line**: Enter the command line or the path to the script that Monitoring Studio will execute. You can use the following macros in the command line that will be executed:
     - `%{SEN_TIME:<date-time-format>`: Use this macro to insert the current date and time in the command line. You can specify the format of the date and time string that will be inserted, which follows UNIX asctime() format (%Y for year, %m for month, %D for day, %H for hours, %M for minutes, %S for seconds, etc.).
     - `%{SEN_LASTTIME:<date-time-format>`: Use this macro to insert the date and time at which the command was last executed. This can be particularly useful when you need to specify a time range for the command, like listing events since the last time we
checked. The format is the same as the UNIX asctime() format (%Y for year, %m for month, %D for day, %H for hours, %M for minutes, %S for seconds, etc.).

⚠ When using the %{SEN_LASTTIME:<date-time-format>} macro, the execution is skipped entirely the first time the Monitor runs (after the PATROL Agent starts). This is to ensure that an actual date and time is inserted with an actual value for this macro.

- %{SEN_SCRIPTPATH:<local-script-path>}: Use this macro to copy a file stored on the PATROL Agent's system to the monitored host before the command is executed. When the command is executed, the macro is replaced by the path to the copied file on the targeted host. This is particularly useful to trigger the execution of scripts that are stored on the PATROL Agent's system without having to install these scripts on each monitored host. This macro is irrelevant when monitoring the localhost.
  
  The %{SEN_SCRIPTPATH} macro should provide the script file path on the local Agent system.

- %{HOSTNAME}: This macro inserts the hostname of the targeted system, as specified in the host configuration (it therefore may be an IP address, a FQDN or a short name).
- %{USERNAME}: Use this macro to insert the username of the specified credentials in the command line to be executed.
- %{PASSWORD}: Use this macro to insert the password of the specified credentials in the command line to be executed. The password is inserted in clear text.

  Passwords should never be sent in clear text. Passwords in command lines may be visible to non-root users. Use at your own risk.

7. Configure the settings for the **Command Execution Validation**:

![Image of Command Execution Validation](image-url)
8. Set the following **Command Execution Validation** options:

- **Timeout**: Enter the time in seconds after which the command will be stopped (Default: 30 seconds). If the timeout is reached, the value of the Status parameter will be set to 2 (Failed), indicating that the command failed to execute properly. No further analysis will be performed (String or Numeric Value searches).

- **(Optional) Execution is validated when output contains**: Enter the regular expression that needs to match the command output for the command to be considered successful. The regular expression entered here will be searched in the output of the command. If it is not found, the value of the Status parameter will be set to 2 (Failed), indicating that the command failed to execute properly. No further analysis will be performed (String and Numeric Value searches).

This option can be particularly useful to ensure that the command has been properly executed and avoid false alerts triggered by the associated String Search/Numeric Value Extraction when an error is encountered during the command execution. For example, if the specified command must print a text banner, you will want to make sure that the text banner is found in the command output instead of an error message).

- **Exit codes below mean the command execution**: Select an execution option (succeeded/failed), to state if the command line was properly executed or not. When one of the exit codes is found or not found, depending on the option selected, the ExitStatus parameter of the Command Line monitor is automatically set to 1 (Failed) or 0 (Succeeded) indicating that the command failed or succeed to execute properly.

- **(Optional) Exit codes separated by commas**: Enter one or several exit codes separated by commas. When one of the exit codes is found or not depending on the selected execution option, the ExitStatus parameter will be set to 1 (Failed) or 0 (Successful).

- **The ExitCode and ExitStatus parameters are deactivated for never-ending commands and commands for which no exit code is specified or when the exit code is ignored. Also, the ExitTime parameter is deactivated for never-ending command lines to avoid meaningless alerts (SEN_MS_COMMANDLINE).**

- **Report execution errors in Group's CollectionErrorCount**: Select this option to have the CollectionErrorCount parameter of the Group reflect possible alerts triggered upon the Command Line execution. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor, associated to the Group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.

9. Click **Next**.

11. **Configure the Monitor settings**.

12. Click **Finish**. The corresponding Command Line instance (Command Line: <Display Name>) is created in the PATROL Console. The collected parameters for Command Line Monitors are listed in the **SEN_MS_COMMANDLINE** chapter.
Performing a Database Query

The **Database Query** Monitor executes SQL queries on the most popular database servers currently available on the market (SQL Server, MySQL, Oracle and PostgreSQL) and monitors their return output within your PATROL environment. You can then run string or numeric value searches, or perform value mapping on the return output to monitor the result in myriad ways.

If the technology you wish to monitor uses a database server, you can test this database by sending applicative queries to the database server, or by testing the content of some critical data tables. As query results are stored by Monitoring Studio in a pipe-separated table format, it is easy to search for strings or numeric values and extract these values from a database query result. Refer to the [String Search](#), [Numeric Value Extraction](#) and [Value Map](#) sections for more information.

Supported databases are:

- Microsoft SQL Server
- MySQL Server
- Oracle Database Server
- Other Database
- PostgreSQL
Performing a Query on a Microsoft SQL Server Database

This section details the various connection settings available for performing queries on Microsoft SQL database server.

To configure a connection to a Microsoft SQL database server

1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Database Query: Microsoft SQL Server from the drop-down list and click Next.

3. Define the Connection Information to the Microsoft SQL database server:
   - **Port or Instance name**: Specify the SQL server instance name if there are several SQL Server instances installed. Leave “default” if there is a single instance; or enter the port to be used to access the Microsoft SQL database (Default: port 1433).
   - **(Deprecated) SSL Encryption**: Since version 9.4.01 of Monitoring Studio, the SSL Encryption option is deprecated as it is handled automatically by the Microsoft's JDBC driver.

⚠️ These settings apply to all Microsoft SQL Server connections defined for the host.
4. Click Next.

5. Define the **Query Information** to the Microsoft SQL database server:
   - **Credentials**: Select the type of credentials that you want to use:
     - **System Credentials (default)**: Uses the credentials previously specified when creating the Host.
     - **Add new credentials...**: Lets you set new specific credentials.
   - **(Deprecated) Authentication**: Since version 9.4.01 of Monitoring Studio, the SQL Server Authentication option is deprecated as it is handled automatically by the Microsoft's JDBC driver. To use Windows-integrated Authentication, you need to specify new Credentials with empty username and empty password. In this case, the connection will be done using the PATROL Agent's default account.
   - **Database name**: Enter the name of the database. (Default: master)
   - **SQL Query**: Enter the SQL query you wish to perform.
   - **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the query times out, the Status parameter will be set to 2 (Failed) and an alarm will be triggered.
   - **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the CollectionErrorCount parameter of the Group reflect possible alerts triggered upon the query execution. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.
6. Click **Next**.
7. **Configure the Monitor settings**.
8. Click **Finish**. The corresponding Database Query instance (Database Query: <Display Name>) is created in the PATROL Console. The collected parameters for Database Query Monitors are listed in the **SEN_MS_DBQUERY** chapter.

**Performing a Query on a MySQL Server Database**

This section details the various connection settings available for performing queries on MySQL database server.

**To configure a connection to a MySQL server database**

1. In the PATROL Console, right-click the **Host** or **Monitor Group** icon and select **KM Commands > New > Monitor...**
2. Select **Database Query: MySQL Server** from the drop-down list and click **Next**.

   ![New Database Query for Host 1](image)

3. Define the **Connection Information** to the MySQL database server:
   - **Database name**: Enter the name of the database.
   - **Port number**: Enter the port to be used to access the MySQL database (Default: port 3306).

   ![](image)  

   *These settings apply to all MySQL Server connections defined on pc-cpwin7.*
4. Click **Next**.

![New Database Query for Host](image)

**Performing a Query on a MySQL Server Database - MySQL Server Query Information**

5. Define the **Query Information** to the MySQL database server:
   - **Credentials**: Select the type of credentials that you want to use:
     - **System Credentials (default)**: Uses the credentials previously specified when creating the Host.
     - **Add new credentials...**: Lets you set new specific credentials.
   - **SQL Query**: Enter the SQL query you wish to perform.
   - **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the query times out, the **Status** parameter will be set to 2 (Failed) and an alarm will be triggered.
   - **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the **CollectionErrorCount** parameter of the Group reflect possible alerts triggered upon the query execution. The **CollectionErrorCount** parameter of the Group reports on the collection errors of any Monitor associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.

6. Click **Next**.

7. **Configure the Monitor settings**.

8. Click **Finish**. The corresponding Database Query instance (Database Query: <Display Name>) is created in the PATROL Console. The collected parameters for Database Query Monitors are listed in the **SEN_MS_DBQUERY** chapter.
Performing a Query on an Oracle Database Server

This section details the various connection settings available for performing queries on Oracle database server.

To configure a connection to an Oracle database server

1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Database Query: Oracle Database Server from the drop-down list and click Next.

3. Define the Connection Information to the Oracle database server:
   - **Database name**: Enter the name of the database.
   - **Port number**: Enter the port to be used to access the Oracle database (Default: port 1521).

   ![Database Query Window]

   These settings apply to all Oracle Database Server connections defined on pc-cpwin7.

   *These settings apply to all Oracle Database Server connections defined for the host.*
4. Click Next.

5. Define the Query Information to the Oracle database server:
   - **Credentials**: Select the type of credentials that you want to use:
     - **System Credentials (default)**: Uses the credentials previously specified when creating the Host.
     - **Add new credentials...**: Lets you set new specific credentials.
   - **SQL Query**: Enter the SQL query you wish to perform.
   - **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the query times out, the Status parameter will be set to 2 (Failed) and an alarm will be triggered.
   - **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the CollectionErrorCount parameter of the Group reflect possible alerts triggered upon the query execution. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.

6. Click Next.

7. **Configure the Monitor settings**.

8. Click Finish. The corresponding Database Query instance (Database Query: <Display Name>) is created in the PATROL Console. The collected parameters for Database Query Monitors are listed in the **SEN_MS_DBQUERY** chapter.
Performing a Query on a PostgreSQL Database

This section details the various connection settings available for performing queries on PostgreSQL database server.

To configure a connection to a PostgreSQL database server:
1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Database Query: PostgreSQL from the drop-down list and click Next.

3. Define the Connection Information to the PostgreSQL database server:
   - **Database name**: Enter the name of the database.
   - **Port number**: Enter the port to be used to access the PostgreSQL database (Default: port 5432).

   *These settings apply to all PostgreSQL connections defined on the host.*
4. Click Next.

5. Define the **Query Information** to the PostgreSQL database server:
   - **Credentials**: Select the type of credentials that you want to use:
     - **System Credentials** (default): Uses the credentials previously specified when creating the Host.
     - **Add new credentials...**: Lets you set new specific credentials.
   - **SQL Query**: Enter the SQL query you wish to perform.
   - **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the query times out, the Status parameter will be set to 2 (Failed) and an alarm will be triggered.
   - **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the CollectionErrorCount parameter of the Group reflect possible alerts triggered upon the query execution. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.

6. Click Next.

7. Configure the Monitor settings.

8. Click Finish. The corresponding Database Query instance (Database Query: <Display Name>) is created in the PATROL Console. The collected parameters for Database Query Monitors are listed in the **SEN_MS_DBQUERY** chapter.
Performing a Query on any Other Database

This section details the various connection settings available for performing queries on database servers other than MS SQL, MySQL, Oracle or PostgreSQL.

To configure a connection to any other database server
1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Database Query: Other (JDBC) from the drop-down list and click Next.

### Performing a Query on an Other Database - Other (JDBC) Connection Information

3. Specify the **Connection Information**:
   - **JDBC URL**: Enter the URL to use for connecting to the JDBC database. You can use the `%{SEN_USERNAME}`, `%{SEN_PASSWORD}` and `%{SEN_PASSWORD}` macros to insert the username and password (Example: jdbc:postgresql://hostname:5432/dbname).
   - **Driver class**: Enter the driver class for the JDBC database (Example: org.postgresql.Driver).

⚠️ These settings apply to all Other (JDBC) connections defined on the host.
4. Click **Next**.

5. Define the **Query Information** to the Other (JDBC) database server:
   - **Credentials**: Select the credentials that you want to use:
     - **System Credentials (default)**: Uses the credentials previously specified when creating the Host.
     - **Add new credentials...**: Lets you set new specific credentials.
   - **SQL Query**: Enter the SQL query you wish to perform.
   - **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the query times out, the **Status** parameter will be set to 2 (Failed) and an alarm will be triggered.
   - **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the **CollectionErrorCount** parameter of the Group reflect possible alerts triggered upon the query execution. The **CollectionErrorCount** parameter of the Group reports on the collection errors of any Monitor associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.

6. Click **Next**.
7. **Configure the Monitor settings**.
8. Click **Finish**. The corresponding Database Query instance (Database Query: <Display Name>) is created in the PATROL Console. The collected parameters for Database Query Monitors are listed in the **SEN_MS_DBQUERY** chapter.
Monitoring Files

The File Monitor is designed to monitor the presence, content, growth and change of a specific file. Therefore, the solution is able to instantly detect and alert when a critical file goes missing or if the file size is growing too fast for example. It is one of the most important monitoring tools offered by Monitoring Studio KM for PATROL as a lot of technologies deal with files and many of them are critical. The most typical usage of file monitoring is parsing a log file. Most technologies use log files to trace their operations and notify operators when failures occur.

Selecting the right file type to monitor (flat or log) is essential to allow Monitoring Studio to read the monitored file correctly. Note that searching strings in flat or log files is also performed slightly differently: Flat files are entirely updated and therefore need to be parsed entirely, as opposed to log files where new lines are appended at the end of the file - and hence only these new lines need to be analyzed.

To monitor a file (flat and log)
1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select File from the drop-down list and click Next.
3. Specify the type of file that you wish to monitor:
   - Log file: if you only want the new lines to be parsed when searching for strings or numeric values, or performing value mapping
   - Flat file: If you want the entire file to be parsed when searching for strings or numeric values, or performing value mapping

![File Type Selection](image_url)
4. Click **Next** to continue.

5. **Credentials**: Select the type of credentials that you want to use for monitoring the file:
   - **Agent's Default Account**: Uses the PATROL Agent credentials for localhost monitoring
   - **System Credentials (default)**: Uses the credentials previously specified when creating the Host
   - **Add new credentials...**: Lets you set new credentials for this specific monitoring. Refer to the Setting Credentials chapter for detailed information

6. Identify the file to monitor:
   - **Filename**: Indicate the path and name of the file to be monitored. You can use:
     - the ? wildcard to replace one character; the * wildcard to replace one or more characters. Example: C:\Program Files\MyTechnology\*.log
   
   ![Monitoring a File - File Information](image)

   "The filename full path supports environment variables such as %PATROL_HOME%.

   "If multiple files match the file mask, Monitoring Studio will monitor the most recent file matching this mask.

   - a date/time format to dynamically add the current date or time to the file name. Simply insert the following string in the "Filename" field, where the date/time format appears, replacing the three dots by date format symbols: %{SEN_TIME:]. For the complete list of format symbols, meaning and examples, please refer to the Format Symbols for Macros section. Example: %{SEN_TIME:%Y-%m-%d %H:%M:%S}
7. Select the parameters you want to monitor. Refer to the `SEN_MS_FILE` section for details on parameters.

8. Click **Next**.
9. **Configure the Monitor settings**.
10. Click **Finish**. The corresponding File instance (Log/Flat File: <Display Name>) is created in the PATROL Console. The collected parameters for File Monitors are listed in the `SEN_MS_FILE` chapter.

**To restart the scan of Log Files**

When looking for strings or numbers in a log file, the file content is scanned as new content is added. This means that the information that is in the file is only scanned once and then skipped over by the next polling. This option offers the possibility to restart the scanning from the beginning of the file. To do so:

1. Right-click the **Log File > KM commands > Restart Scan**
2. A message prompts for confirmation.
3. Click the **Yes** button to proceed. At the next polling, the entire file will be scanned.

⚠️ Once this option has been selected and confirmed, it cannot be canceled.
Monitoring a File System

Because file systems (or Windows logical disks) are often a critical resource for technologies, Monitoring Studio provides a **File System** Monitor to rapidly identify which technologies are impacted when a file system is full.

**To monitor a file system**

1. In the PATROL Console, right-click the **Host** or **Monitor Group** icon and select **KM Commands** > **New** > **Monitor**...
2. Select **File System** from the drop-down list and click **Next**.

3. Specify how you want to select the file system to monitor:
   - Select the file system from current file system list
   - Enter the file system path
4. Select the file system you wish to monitor or enter the file system path and click **Next**.

5. **Configure the Monitor settings.**

6. Click **Finish**. The corresponding File System instance (File System: <Display Name>) is created in the PATROL Console. The collected parameters for File System Monitors are listed in the **SEN_MS_FILESYSTEM** chapter.

**Configuring the File System Cache Refreshing Frequency**

Monitoring Studio relies on a **cache mechanism** to share the information among the Monitors in order to use as little resources as possible on the target Host and over the network. The cache will be refreshed if one of the Monitors needs to collect data (polling interval reached) and the cache is older than the selected minimum cache refresh.

All file systems usage data on the selected Host are cached and shared by all File System Monitors defined on this Host. By default, the file system cache is refreshed minimum every 15 seconds. It is however possible to change this **minimum cache refresh interval** as described in the Setting the Polling Interval section.
Monitoring Folders

Many applications store critical data as files in the filesystem (for example, each pending query is stored as a separate file in a specific folder). In such cases, monitoring the folders (directories) containing these files can prove very useful to detect an abnormal behavior (for example: many files are piling up in the "queue" folder).

The Folder Monitor tracks changes in folders (directories) that store files processed by the application. It measures their size, growth and flow (including how many files moved in, how many moved out, etc.).

This feature allows you to:

- Ensure that your application is not overloaded (number of files to be processed, e.g.).
- Measure the application activity (how many removed files, that is, how many have been processed).
- Check the age of the newest file (whether the data is coming in properly...).
- Check the age of the oldest file (whether the application is running late in processing queued files).

By default, the Folder Monitor automatically times out after 30 seconds.

To monitor a folder

1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Folder from the drop-down list and click Next.
3. Credentials: Select the type of credentials that you want to use for monitoring the folder:
   - Agent's Default Account: Uses the PATROL Agent credentials for localhost monitoring
   - System Credentials (default): Uses the credentials previously specified when creating the Host
   - Add new credentials...: Lets you set new credentials for this specific monitoring. Refer to the Setting Credentials chapter for detailed information.
4. Identify the folder to be monitored:
   - Folder name: Enter the name and path of the folder (directory) to be monitored. You can use a date/time format to dynamically assign the current date or time in the file name. Simply insert the following string in the "Folder name" field, where the date/time format appears, replacing the three dots by date format symbols: %{SEN_TIME:...}. For the complete list of format symbols, meaning and examples, please refer to the Format Symbols for Macros section. Example: %{SEN_TIME:%Y-%m-%d %H:%M:%S}
   - The folder name full path supports environment variables such as %PATROL_HOME%
   - Include sub-folders: Check this option to monitor all the sub-folders of the specified folder
   - (Optional) Monitor only files matching this mask: Specify the file types or enter masks, and only these files will be monitored (e.g.:*.txt;myFiles?.log;file.*). You can use wildcards such as "*" to replace any number of characters, or "?" to replace just one character. You may
also use several masks separated by ";". Monitoring Studio will only take into account the files matching the masks entered.

- Click Next.

5. Select the **parameters** to be monitored for this folder.

6. Select an option from the **When folder is empty** field to have Monitoring Studio perform the corresponding action on the time-based parameters when the monitored folder is empty (the time-based parameters are: `OldestModifiedFileElapsedTime`, `LastModified`, `FileElapsedTime` and `LongestTimeFileRemainsInFolder`):

   - **Do not update the time-based parameters**: The parameters' values are not updated, and the alerts are not cleared (Default). In this case, the parameters keep the same value as previously set upon the last collect. If the last value set was within an alarm range, the alert remains active until the value is set again, that is when the folder is no longer empty.

   - **Suspend the time-based parameters**: The parameters' values are not set, but any alert is cleared when the folder becomes empty. In this case, the parameters are suspended (i.e. deactivated) and immediately enabled again. The value of these parameters are not updated and the parameters remain offline in the PATROL Console until a new value is set, that is when the folder is no longer empty. If the parameters were previously in alarm, the alert is not cleared (No PATROL event (STD/9) is triggered).

   - **Set the time-based parameters to zero**: The parameters' values are reset to zero and all alerts are automatically acknowledged. In this case, the parameters are set to zero as long as the folder remains empty. While the value could be considered incorrect, it ensures that previous alerts are cleared and that the corresponding PATROL event (STD/9) is triggered (assuming that zero is out of the alarm range).
3. Specifying the Monitors Used

7. Click **Next**.
8. **Configure the Monitor settings**.
9. Click **Finish**. The corresponding File instance (Folder: <Display Name>) is created in the PATROL Console. The collected parameters for Folder Monitors are listed in the **SEN_MS_FOLDER** chapter.
Leveraging Values from Other KMs' Parameters

The **Multi-Parameter Formula** Monitor can monitor all the KMs that are loaded in your PATROL Console. This feature more precisely extracts other KMs' parameter values and computes them with a mathematical formula or a pre-defined PSL function. The returned value can be used to perform additional operations such as converting units, performing correlation, etc.

**To create a multi-parameter formula**

1. In the PATROL Console, right-click the **Host** or **Monitor Group** icon and select **KM Commands > New > Monitor...**
2. Select **Multi-Parameter Formula** from the drop-down list and click **Next**.
3. Specify the parameters to use in the formula:
   - In the list of parameters currently available on the Agent, select the parameters required (The maximum number of parameters supported by Monitoring Studio is 26)
   
   **Offline parameters and parameters with no values are displayed as "OFFLINE" in the list. Avoid selecting these parameters if you want your formula to be properly applied.**

   - (Optional) Check the **Use the objects display name to build the PATROL IDs** option to allow the use of instance labels instead of PATROL IDs to determine a parameter's path. This option may be particularly useful when an object ID is unknown
   - Click **Next**
3. Specifying the Monitors Used

4. Define the formula to apply to the monitored parameter(s):

- Command Line: application_ping.sh; ExecutionTime
- Database Query: select * from products; ExecutionTime
- Web Request: http://www.sentrysoftware.com; ExecutionTime
- **Formula**: Enter the formula you wish to apply to the parameter(s). The parameters are identified by a letter listed in the dialog box. Build the required formula using:
  - The uppercase letter preceding the parameter (A, B, etc.).
  - PSL operators: addition (+), subtraction (-), multiplication (*), division (/), concatenation (.), bitwise (&), etc.
  - Predefined PSL functions: `swTableJoin()`, `swGetMinimum()`, `swGetMaximum()`, and `swGetAverage()`.

### Examples for predefined PSL functions

**To join two different tables from text parameters**

The `swTableJoin()` PSL function is used to join two different tables from text parameters:

```plaintext
swTableJoin(tableA, separatorsA, keyColumnA, tableB, separatorsB, keyColumnB, defaultRightLine, keyType, timeout)
```

where:
- `tableA` is the left table derived from text parameter A
- `separatorsA` are the separators that separate the columns in `tableA`
- `keyColumnA` is the key column number in `tableA` used for matching the key columns in `tableB`
- `tableB` is the right table derived from text parameter B
- `separatorsB` are the separators that separate the columns in `tableB`
- `keyColumnB` is the key column number in `tableB` used for matching the key columns in `tableA`
- `defaultRightLine` (Optional) default rightTable line, when a match is not found
- `keyType` (Optional) key type such as wbem used by Matsya TableJointClient used by the Java client.
- `timeout` (Optional) timeout for the table joint query.

**Example:**

```plaintext
swTableJoin(A, "|", 1, B, "|", 1)
```

where:
- **Table 1 (in parameter A: path1/Result):**
  - key 1|A|B|C|
  - key 2|aa|bb|cc|
  - key 3|1|2|3|

- **Table 2 (in parameter B: path2/Result):**
  - key 1|X|Y|Z|
  - key 2|xx|yy|zz|
  - key 3|4|5|6|
3. Specifying the Monitors Used

### 3.1 Specifying the Monitors

#### 3.1.1 Specifying the Monitors Type

1. **Monitoring Studio KM for PATROL 9.4.01**

#### 3.1.2 Monitor Settings

- **To find the minimum value among several parameters:**
  - Use the `swGetMinimum()` PSL function to find the minimum value for the chosen list of parameters (where A and B are number parameters):
    ```plaintext
    swGetMinimum([A, B])
    ```

- **To find the maximum value among several parameters:**
  - Use the `swGetMaximum()` PSL function to find the maximum value for the chosen list of parameters (where A and B are number parameters):
    ```plaintext
    swGetMaximum([A, B])
    ```

- **To find the average value among several parameters:**
  - Use the `swGetAverage()` PSL function to find the average value for the chosen list of parameters (where A and B are number parameters):
    ```plaintext
    swGetAverage([A, B])
    ```

The returned output of the `swGetMinimum`, `swGetMaximum`, and `swGetAverage` PSL functions will be displayed by the **Value** parameter.

**Note:** If the formula or parameters entered are not PSL-compatible, an error will be reported by the Group’s **CollectionErrorCount** parameter.

- **Do not collect if one or more parameter has no value:** Select this option if you wish to skip the collect when one or more parameters are not populated (usually after a PATROL Agent restart).
- **Click Next.**

5. **Configure the Monitor settings.**

6. **Click Finish** to start monitoring the selected parameters. The corresponding Multi-Parameter Formula instance (Multi-Parameter Formula: <Display Name>) is created in the PATROL Console. The collected parameters for Multi-parameter Formula Monitors are listed in the **SEN_MS_FORMULA** chapter.

---

**The returned output is a table that is set to the Result text parameter:**

```plaintext
key 1;A;B;C;key 1;X;Y;Z;
key 2;aa;bb;cc;key 2;xx;yy;zz;
key 3;1;2;3;key 3;4;5;6;
```
Monitoring SNMP Agents & Traps

Many devices use SNMP to report their health and operations. They often use SNMP traps to report failures or they can also embed a true SNMP agent and a documented MIB which specifies the meaning of each SNMP OID. Listening to SNMP traps and, even better, polling SNMP agents constitute an efficient way to ensure that your device is operating properly.

Monitoring Studio supports SNMP v1, v2c and v3 for SNMP polling. For SNMP traps, only SNMP v1 is supported.

Polling SNMP Agents

The SNMP Polling Monitor is designed to poll the SNMP agent and retrieve the values of a given OID (object identifier), or the values of an SNMP table, thereby enabling you to easily identify the source of the problem or just be informed of the status of the monitored device/attribute.

SNMP polling supports SNMP v1, v2c, and v3.

To poll an SNMP agent

1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select SNMP Polling from the drop-down list and click Next.

If SNMP is deactivated on the PATROL Agent, a warning message is displayed, inviting you to activate it using the following configuration variable: /snmp/support. Besides, Monitoring Studio will trigger an alert on the CollectionErrorCount parameter under the Monitoring Studio icon if SNMP is disabled in the configuration and one or several SNMP Monitors have been created.
3. Specifying the Monitors Used

3. Specifying the Monitors Used

Monitoring Studio KM for PATROL 9.4.01

Polling SNMP Agents - Value Selection

3. Select the type of value you want to poll:
   - **A single value from one OID**: Select this option to poll a value from one OID.
   - **Multiple values from rows of an SNMP table**: Select this option to poll several values from an SNMP table.
   - **Click Next**.

4. If you have selected **A single value from one OID**, the following dialog box is displayed:

   - **SNMP Polling: Single Value**
   - **SNMP Version**: 2
   - **Port**: 161
   - **OID to poll**: `1.3.6.1.4.1`

   You need to enter the numeric-formatted OID of the value you want to retrieve from the SNMP agent.
   - **Report execution errors in Group's CollectionErrorCount**
The **SNMP version** and **Port** number configured at the host level is displayed at the top of the dialog box.

- **OID to poll**: Enter the OID (object identifier) to poll, as indicated in the Management Information Base (MIB).

- **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the CollectionErrorCount parameter of the Group reflect possible alerts triggered upon the SNMP monitoring execution. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor, associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.

- **Click Next**.

Monitoring Studio will try to poll the OID to identify its type. This will determine which parameters are activated ("Content" for strings, "Value" for numbers). If, for some reason, the OID cannot be polled, a warning message is displayed and invites you to select an OID type, either “string” or “number,” to let Monitoring Studio know what type of value is expected. If you are not sure about:

- the OID type, it is recommended to select “The OID content is a string.”
- the exact OID to poll, you should use a SNMP MIB Browser tool to identify which OID corresponds to which variable. You can also use the Monitoring Studio built-in SNMP Browser tool to list the available OID variables and their values. To do so, right-click the main Monitoring Studio icon > KM Commands > Tools > SNMP Browser.

If the OID returns a number, select the parameters you wish to collect:

- **Delta**: Calculates the difference between values collected during two consecutive polling.
- **Delta per second/minute/hour**: Reports the value resulting of the division of the Delta...
3. Specifying the Monitors Used

by the number of seconds/minutes/hours elapsed between the collection intervals.

- **Raw Value**: Reports the actual value collected upon data polling.
- **Discard negative Delta values**: Select this option if you want to exclude negative Delta values from other collected values.

*Discarding negative Delta values can be particularly relevant when the raw value consists in a counter that keeps increasing: if the raw value decreases, it indicates that the counter has been reset and that the value should be discarded.*

- **Rescaling**: Use this option to configure Monitoring Studio to rescale the value that is being extracted in order to have a more readable graph in the PATROL Console. The available rescaling options are:
  - No rescaling
  - Divide the value by a value that gives you the reading in terms best suited to you
  - Multiply the extracted value by a constant factor

This can be useful if you extract numeric values in bytes but prefer to see the graph in megabytes. In such a case, you would divide the values by 1048576 (1024*1024).

- Click Next.

5. If you have selected **Multiple values from rows of an SNMP table**, the following dialog box is displayed:

![SNMP Polling Multiple Values Dialog Box](image)

The **SNMP version** and **Port** number configured at the host level is displayed at the top of the dialog box.

- **SNMP Table OID**: Enter the Table OID (object identifier) to poll, as indicated in the Management Information Base (MIB). This OID should always end with ".1".
- **Columns to retrieve, separated by commas**: Enter the column numbers from which values
should be retrieved. Enter “ID” to retrieve the row identifier. Leave the field blank to retrieve values from the entire row. Enter "ID" to retrieve the row identifier

- Click Next.

⚠️ If for some reason, the OID cannot be polled, a message is displayed stating that the table seems to be empty. In such a case, verify that you have entered the right OID and/or that the table really exists and contains values.

6. Configure the Monitor settings.

7. Click Finish. The corresponding SNMP Polling instance (SNMP Polling: <Display Name>) is created in the PATROL Console. The collected parameters for SNMP Polling Monitors are listed in the SEN_MS_SNMP POLLING chapter.

Monitoring Studio will get the value and allow you to store it in a graph, or search for strings in the OID content.
Listening for SNMP Traps

Many devices use SNMP to report their health and operations. They often use SNMP traps to notify failures but the best case is when they embed a true SNMP agent and a documented MIB which specifies the meaning of each SNMP OID. Listening for SNMP traps is a safe method to ensure that your device is operating properly.

The **SNMP Trap Listening** Monitor monitors and listens for SNMP traps and enables rapid recovery actions depending on the traps received, thereby ensuring optimal functioning of applications or devices that send SNMP traps.

⚠️ Only SNMP version 1 is supported for SNMP trap listening. SNMP v2c and v3 traps are not supported.

⚠️ The SNMP Agent emitting the traps should be configured to send them to the PATROL Agent where Monitoring Studio is installed and running, otherwise, Monitoring Studio will not be able to receive the SNMP trap and will only listen for the SNMP traps on the localhost. No other Trap Listener should be running at the same time.

**To listen for SNMP traps**

1. In the PATROL Console, right-click the **Host** or **Monitor Group** icon and select **KM Commands > New > Monitor...**
2. Select **SNMP Trap** from the drop-down list and click **Next**.

⚠️ If SNMP is deactivated on the PATROL Agent, a warning message is displayed, inviting you to activate it using the following configuration variable: /snmp/support. Besides, Monitoring Studio will trigger an alert on the CollectionErrorCount parameter under the Monitoring Studio icon if SNMP is disabled in the configuration and one or several SNMP monitors have been created.
3. Identify the traps to listen to:
   - **Enterprise OID**: Enter the Enterprise OID of the SNMP Trap. You can use regular expressions (Example: `1.3.6.1.4.44.141.32.*`)
   - **(Optional) Trap numbers**: Enter the number identifying the SNMP Trap
   - **Optional Varbinds**: Click this button to configure the Varbinds settings. A Varbind or Variable Binding is a sequence of two specific fields. The first field is an OID, which addresses a specific parameter; the second field contains the value of the specified parameter

   - **OID 1 & 2**: First and second attached variables OID that should be contained within the SNMP trap. You can also enter the text that should be found (or not) within the OID content.
   - **Case sensitive**: Indicates whether or not the search will be case sensitive.
   - **Click Accept**.
All these pieces of information define the SNMP Trap that is expected. Monitoring Studio will react to the SNMP Trap received only if this information is found within the SNMP Trap. All other SNMP Traps will be ignored.

⚠️ If you are unsure about the characteristics of the SNMP trap you want to detect, you will need to use a SNMP MIB Browser tool to understand the exact meaning of each trap. You can also use the Monitoring Studio built-in SNMP trap listener tool to view in real-time the SNMP traps and their characteristics that are received by the PATROL Agent. Right-click on the main Monitoring Studio icon > KM commands > Tools > Real-time SNMP Trap Listener.

⚠️ The SNMP trap listening port is actually a PATROL Agent configuration variable: /snmp/trap_port=162. You can set this variable to whatever port you want Monitoring Studio to listen on.

⚠️ Make sure that only one device is listening for traps on the localhost or Monitoring Studio will not be able to run the SNMP Trap listener.

4. Click Next.

5. Set the automatic acknowledgment:
   - **Acknowledge alert if the following SNMP Trap is received**: Check this option if you wish to acknowledge an alert when the SNMP trap received matches the criteria set.
   - **Trap number**: Enter the trap number that will acknowledge the alerts triggered by the specified trap.
   - **Varbinds**: Click this button to configure the first and second attached variables OID that should be contained within the SNMP trap to acknowledge a matching trap received.
   - **Acknowledge alert after**: Check this option if you wish to automatically acknowledge a matching SNMP trap and then specify the time in seconds after which the matching SNMP Trap received will be acknowledged (Default: 120 minutes).
3. Specifying the Monitors Used

- Select the action to be taken when acknowledging: **Reset MatchingTrapCount**, i.e. clear all previous alerts or **Decrease MatchingTrapCount by one**; i.e. clear the previous alert.

6. Click **Next**.
7. **Configure the Monitor settings**.
8. Click **Finish**. The corresponding SNMP Trap instance (SNMP Trap: <Display Name>) is created in the PATROL Console. The collected parameters for SNMP Trap Listening Monitors are listed in the [SEN_MS_SNMPTRAP](#) chapter.
Performing Web Requests

The **Web Request** Monitor allows you to extract data from any Web-based interface. You can either monitor the availability of a Web page, extract useful information from a Web-based administration UI, or extract data from a Web service or a REST API. If your IT environment uses a proxy server, please make sure that the **Proxy Settings** are properly configured.

* Web Requests are always executed locally (on the PATROL Agent node).

The Web Request Monitor enables you to configure the most common type of requests:

- **GET** that requests data from a specified resource.
- **POST**, **PUT** and **DELETE** that submits data to be processed to a specified resource.

Refer to the procedure corresponding to the type of Web request you wish to perform.

To perform a Web request (GET)

The **GET** method is commonly used to retrieve information from a given server using a specific URI. Requests using GET should only retrieve data and should have no other effect on the data.

The example illustrated in this section shows how to retrieve information about volumes on an EMC XtremIO server.
1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Web Request from the drop-down list and click Next.

Performing Web Requests (GET) - Web Request Information

- Enter the URL of the resource that needs to be polled and monitored. It is possible to poll a secure web site by using the "https" method.
- Select the GET request type.

⚠ Some Web forms can be passed to the server with the HTTP GET method. In this case, the form data goes through the URL (http://server/form.php?varA=valueA&varB=valueB&...).

- **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the request times out, the Status parameter will be set to 2 (Failed) and an alarm will be triggered.
- **Content to be parsed**: Select the option corresponding to the information you want to retrieve from the Web page output returned by the Web server:
  - Entire HTTP response - header + body
  - HTTP response body
  - Extract text from HTML (Default)
  - Conversion of JSON to flat map

The current example polls the URL of an EMC XtremIO server using a GET request type. The "Extract Text from HTML" option is used to retrieve the required volume information.
3. (Optional) Click the **Authentication** button to configure the proxy settings.

![Authentication](image)

**Proxy Settings**

- **Bypass the proxy (lagrange)**

**HTTP Authentication**

**Credentials:** [No authentication]

Once credentials are configured, you can use the `%{SEM_USERNAME}`, `%{SEM_PASSWORD}` and `%{SEM_PASSWORD_BASE64}` macros in any relevant field (URLs, headers, body, etc.) for reference.

**Performing Web Requests - Proxy Settings**

- **Bypass the proxy**: Check this option if you have configured **Proxy Settings** and wish to bypass the proxy for this Web Request specifically. This can be useful when the resource is located on the internal network and the proxy refuses to serve it.

- **HTTP Authentication**: Select the method you want to use to provide the credentials for HTTP authentication:
  - **No authentication**: The Web Request does not require any authentication (Default).
  - **System Credentials**: Uses the system credentials previously specified when creating the Host.
  - **Add new credentials**: Lets you set new specific credentials.

The current example uses the default option **No Authentication**; the authentication credentials will be provided through a header.

*Refer to the [HTTP authentication](#) section on for more details.*

4. Click **Accept** to save your changes.

- **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the `CollectionErrorCount` parameter of the group reflect possible alerts triggered upon the Web Request execution. The `CollectionErrorCount` parameter of the Group reports on the collection errors of any Monitor associated to the Group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.
5. Click **Next** to continue.
6. (Optional) Specify **Request Headers**, that is **Key** (name) and **Value** pairs that are displayed in the request and response of message headers for Hypertext Transfer Protocol (HTTP). HTTP headers are an integral part of HTTP requests and responses and are mainly intended for the communication between the server and client in both directions.

    ![Image of Request Headers](image)

    **Performing Web Requests (GET) - Request Headers**

    The current example uses a basic authentication header specific to EMC XtremIO REST API.

7. Click **Next** to continue.
8. **Configure the Monitor settings**.
9. Click **Finish**. The corresponding Web Request instance (Web Request: <Display Name>) is created in the PATROL Console. The collected parameters for Web Request instances are listed in the **SEN_MS_WEBREQUEST** chapter.
10. Click the **Result** parameter to display the output of the request.

    The current example shows a list of all the volumes retrieved from the target server.
3. Specifying the Monitors Used

Monitoring Studio KM for PATROL 9.4.01

Performing Web Requests (GET) - Request Headers

Web Request (GET) output is displayed as list that might be difficult to read or extract valuable information. Use the Text Pre-processing tool to format the result output of your request.

To perform a Web request (POST, PUT, DELETE)

The example illustrated in this section shows a POST request performed on an XtremiO server to modify the size of a specific volume.
1. In the PATROL Console, right-click the **Host** or **Monitor Group** icon and select **KM Commands > New > Monitor...**
2. Select **Web Request** from the drop-down list and click **Next**.

- **Web Request Information**
  - **URI**: `https://10.0.11.97/api/json/v2/types/volumes`
  - **HTTP request type**: POST
  - **Timeout**: 30 seconds
  - **Content to be parsed**: Extract text from HTML
  - **Authentication**: Off
  - **Report execution errors in Group’s CollectionErrorCount**

**Performing Web Requests - Web Request Information**

- **Enter** the **URI** of the resource that needs to be polled and monitored. It is possible to poll a secure web site by using the "https" method.
- **Select a Request Type**, such as **POST** in the current example:
  - **POST**: A POST request is used to send data to the server: customer information, file upload, or volumes as in the current example, etc. using HTML forms.
  - **PUT**: Replaces all the current representations of the target resource with the uploaded content.
  - **DELETE**: Removes all the current representations of the target resource given by URI.
  - **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the request times out, the **Status** parameter will be set to 2 (Failed) and an alarm will be triggered.
- **Content to be parsed**: Select the option corresponding to the information you want to retrieve from the Web page output returned by the Web server.

**The current example polls the URL of an EMC XtremIO server using a GET request type. The "Extract Text from HTML" option is used to retrieve the information about volumes for which a specific parameter (size) will be modified by a request body (step 8).**

- **Entire HTTP response - header + body**
- **HTTP response body**
- **Extract text from HTML (Default)**
- **Conversion of JSON to flat map**
• Click the **Authentication** button to configure the proxy settings.

![Authentication](image)

- **Bypass the proxy**: Check this option if you have configured Proxy Settings and wish to bypass the proxy for this Web Request specifically. This can be useful when the resource is located on the internal network and the proxy refuses to serve it.
- **HTTP Authentication**: Select how you want to enter the credentials for HTTP authentication.

  - **No authentication**: The Web Request does not require any authentication.
  - **System Credentials**: Uses the system credentials previously specified when creating the Host.
  - **Add new credentials**: Lets you set new specific credentials.

  > Refer to the [HTTP authentication](#) section on for more details.

3. Click **Accept** to save your changes.

- **To post a form to a Web server**, you need to enter the URL of the script/CGI/page that will actually process the data, which may be different from the Web page URL that shows the form itself. Again, you need to check the Web form HTML source to identify the URL to query.

- **A Web form HTML source should contain a statement like the following one**: `<FORM METHOD="POST" ACTION="/urlToQuery.php">`.

- **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the CollectionErrorCount parameter of the group reflect possible alerts triggered upon the Web Request execution. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor associated to the Group, for which this option is available.
and selected, providing a global view of the collection errors for the whole Group.

4. Click **Next** to continue.

5. (Optional) Specify **Request Headers**, that is **Key** (name) and **Value** pairs that are displayed in the request and response of message headers for Hypertext Transfer Protocol (HTTP). HTTP headers are an integral part of HTTP requests and responses and are mainly intended for the communication between the server and client in both directions.

The current example uses a basic authentication header specific to EMC XtremIO REST API.

6. Click **Next** to continue.

7. Configure **Request Body Type** Settings:

   Provide the required information according to the type of **Request Body** you have selected:

   **A - Request Body (None)**

   No additional information is needed. Simply click **Next** to Configure the Monitor settings

   **B - Request Body (Form)**

   Configure the required settings:
3. Specifying the Monitors Used

Performing Web Requests - Request Body (Form) Settings

- **Content-Type**: Select the option corresponding to the type of content for the request:
  - use *multipart/form-data* to transmit binary (non-alphanumeric) data
  - use *application/x-www-form-urlencoded* otherwise (Default).
- **Variables**: Provide the Variables' Key (name) and Value pairs that compose your query.
- Click **Next** to Configure the Monitor settings.

C - Request Body (Free)

Configure the required settings:

![Request Body Configuration](image)

- **Content-Type**: Select the option corresponding to the type of content for the request.
  
  The current example uses the application/json body content-type that provides the name of the target volume and the parameter to modify in JSON format. In this case the size will be set to 10000m.

- Provide the **Body of the request** (single line) or the **path to the file containing the request Body** (on the PATROL Agent) if the body contains multiple lines. The file path can include environment variables. If a file is provided, the Monitoring Studio discovery will read the file on the localhost and use its content as the request body.
- Click **Next** to Configure the Monitor settings.
8. Click **Finish**. The corresponding Web Request instance (Web Request: <Display Name>) is created in the PATROL Console. The collected parameters for Web Request instances are listed in the **SENM_5_WEBREQUEST** chapter.

9. Click the **Result** parameter to display the output of the request.

Performing Web Requests - Request Body (Free) Settings

For **Web Request (DELETE)**, a value set to 200 in the graph of the **HTTPStatusCode** parameter, indicates that the query has been performed successfully. The **Result** parameter will show 'No Data'. If the query fails, the graph of the **HTTPStatusCode** parameter will display a value of 400.
Monitoring Windows Event Logs

The **Windows Event** Monitor tracks events posted by your technology to consolidate the monitoring under a single icon and avoid you to look up in the Windows Event Log. It also enables you to define automatic acknowledgment of previously triggered alerts by specifying the Windows event that will acknowledge the alert.

*Windows events on Windows 2003 systems can only be monitored locally.*

**To monitor a Windows event log**

1. In the PATROL Console, right-click the **Host** or **Monitor Group** icon and select **KM Commands > New > Monitor...**
2. Select **Windows Event** from the drop-down list and click **Next** to retrieve all the available Windows event logs from your local or remote system.

![Monitoring Windows Events — Welcome Page](image)
3. Select the Windows event log that you want to monitor from the Event log drop-down menu and click Next.

4. All available providers are listed in the Provider drop-down menu. Select one from the list and click Next.
5. (Optional) Define the **Event Settings**:
   - **Event level**: Select the event level(s) to monitor (Critical, Error, Warning, Information)
   - **Count events with these event IDs**: Enter the ID(s) of the event(s) to be considered for alerting.
   - **But exclude these event IDs**: Enter the ID(s) of the event(s) to be excluded from alerting.
   - **Event message must contain/must not contain**: Enter the string or regular expression to look for, and specify whether or not it should be found in the event message.
   - **About Event IDs**: Click this button to get further details on the event ID syntax.

   *Use a comma (,) to separate several IDs or a hyphen (-) between start and end values to indicate a range (Example: 4372,4375,4380-4385).*

   *If you are unsure about the characteristics of the Windows event you want to detect, you may use the Monitoring Studio built-in Windows Event Log Reader tool to view content of the event Logs and the characteristics of the events. Right-click a Host icon > KM commands > Tools > Windows Event Log Reader.*

6. Click **Next**.

7. (Optional) Define the **Acknowledgment Rule**:
   - **Acknowledge alert after**: Check this option and then specify the time in minutes after which the alerts will be acknowledged (Default: 120 minutes).
   - **Acknowledge on these event IDs**: Enter the ID(s) of the event(s) that you wish to acknowledge on. Use a comma (,) to separate several IDs or a hyphen (-) between start and end values to indicate a range.
   - **Event message must contain/must not contain**: Enter the string or regular expression to look for, and specify whether or not it should be found in the event message.
   - **When acknowledging**: Choose between resetting the `MatchingEventCount` parameter, i.e.
clear all previous alerts or simply **decreasing it by one**; i.e. clear the previous alert.

8. Click **Next**.

9. **Configure the Monitor settings**.

10. Click **Finish**. The corresponding Windows Event Log instance (Windows Event Log: <Display Name>) is created in the PATROL Console. The collected parameters for Windows Event Log instances are listed in the **SEN_MS_WINEVENT** chapter.

### Configuring the Windows Event Cache Refreshing Frequency

Monitoring Studio relies on a **cache mechanism** to share the information among the Monitors in order to use as little resources as possible on the target Host and over the network. The cache will be refreshed if one of the Monitors needs to collect data (polling interval reached) and the cache is older than the selected minimum cache refresh frequency.

All Windows event data on the selected host are cached and shared by all Windows Event Monitors defined for this host. By default, the Windows event cache is refreshed minimum every 15 seconds. It is however possible to change this **minimum cache refresh interval** as described in the **Setting the Polling Interval** section.
Monitoring Windows Performance Counters

The Windows Performance Counter Monitor is designed to monitor any commercial Windows-based technology or any custom technology relying on a Windows-based middleware which is instrumented through Windows Performance Counters.

The Windows Performance Counter Monitor collects information about elements on your Windows systems and measures them. These elements can be processors, threads, processes, memory, etc., with each one having an associate set of counters and possibly instances. The data gathered by the Windows performances about specific components can be used to identify problems and bottlenecks within your technology and plan ahead for your future needs.

The Windows Performance Counter Monitor brings Windows performance data, important to the proper functioning of your technology, within your BMC framework and automatically notifies you when a value breaches a specific threshold.

To monitor a Windows performance counter
1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Performance Counter from the drop-down list and click Next.
3. Select the **Windows Performance Object** that contains the counter and click **Next**.

![New Performance Counter for Host 1](image)

4. Select the counter and instances:
   - **Instance name**: Select the **Windows Performance object** to be monitored.
   - **Performance counter**: Select the **counter** to be monitored.
   - *(Optional) Rescaling*: A scale can be used to divide or multiply the Windows performance value by this number. To do so, enter the number you wish the value to be divided/multiplied by (Example: the committed memory is expressed in bytes, so, to obtain the value in mega-bytes (MB), type "1024" in the field for scale. By default, the scale value is 1 (no scale).

5. Click **Next**.
6. **Configure the Monitor settings**.
7. Click **Finish**. The corresponding Performance Counter instance (Performance Counter: <Display Name>) is created in the PATROL Console. The collected parameters for Performance Counter instances are listed in the **SEN_MS_WINPERF** chapter.
Monitoring a Windows Service

Many software technologies that run on any Windows operating system run as Windows services, running as background processes with no direct user interface and no logged-on user. Typically, they start automatically and are expected to keep on running without any human intervention. When one of these critical services fails, many users and external services can immediately be affected. Therefore, making sure that these services are seamlessly running is a key requirement for most system administrators.

To monitor a Windows service

1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select Windows Service from the drop-down list and click Next.

Please note that the System Credentials entered at the host level will be used to retrieve Windows Service information.

You can either:

- select a service from the list, or
- manually enter the service short name
To display the list of installed services

3. Select the **Select a service from the list of installed services** option and click **Next**.
4. Select the service to be monitored and click **Next**.
To manually enter the short name of a service

3. Select the **Enter manually the short name of a service** option and click **Next**.
4. Enter the Windows **service short name** to be monitored and click **Next**.
5. For each possible service state, you can select the value of the Status parameter: OK, Suspicious or Failed. By default, the service state will be interpreted as listed in the table below:

<table>
<thead>
<tr>
<th>Service State</th>
<th>Default Parameter Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Running</td>
<td>0 (OK)</td>
</tr>
<tr>
<td>Paused</td>
<td>1 (Suspicious)</td>
</tr>
<tr>
<td>Stopped</td>
<td>2 (Failed)</td>
</tr>
<tr>
<td>Start Pending</td>
<td>1 (Suspicious)</td>
</tr>
<tr>
<td>Continue Pending</td>
<td>1 (Suspicious)</td>
</tr>
<tr>
<td>Pause Pending</td>
<td>1 (Suspicious)</td>
</tr>
<tr>
<td>Stop Pending</td>
<td>1 (Suspicious)</td>
</tr>
<tr>
<td>Not Installed</td>
<td>2 (Failed)</td>
</tr>
<tr>
<td>Unknown</td>
<td>1 (Suspicious)</td>
</tr>
</tbody>
</table>

6. **Configure the Monitor settings.**

7. Click Finish. The corresponding Windows Service instance (Windows Service: <Display Name>) is created in the PATROL Console. The collected parameters for Windows Service instances are listed in the SEN_Memons chapter.
Running PSL Commands

PSL (PATROL Script Language) is a language developed by BMC for writing complex application discovery procedures, parameters, arbitrary commands, and tasks. This language is commonly used for KM development but can also be useful for IT administrators as they can access the internal information of the PATROL Agent through PSL commands.

The PSL Command Monitor fills the gap of what other monitors cannot offer you. This feature allows you to run PSL commands on the local PATROL Agent system using the default PATROL Agent credentials. You can then define String Searches, Numeric Value Extractions and Value Map that will help you parse and analyze the output of PSL commands.

⚠️ The PSL Command Monitor is intended for advanced users with PSL coding experience.

To run a PSL command
1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select PSL Command from the drop-down list and click Next.

3. Set the following PSL Command Information options:
   - **PSL Command**: Enter the PSL command or the full path to the PSL file that Monitoring Studio will execute. PSL commands are executed on local host, where the PATROL Agent is running.
The PSL file path can include environment variables (example: %PATROL_HOME%/lib/psl/patroldiags.psl).

You can for example execute simple PSL commands to check the health of the PATROL Agent:

**Examples:**

- `system("%PSLPS");` # Reports the PSL processes and their status
- `system("%DUMP RUNQ");` # Reports the list of items scheduled in the run queue
- `system("%STAT ALL");` # Reports all Agent memory usage statistics

**Timeout:** Enter the time in seconds after which the PSL command will be stopped (Default: 30 seconds). If the timeout is reached, the value of the *Status* parameter will be set to 2 (Failed), indicating that the PSL command failed to execute properly. No further analysis will be performed.

4. Click **Next**.
5. **Configure the Monitor settings.**
6. Click **Finish**. The corresponding PSL Command instance (PSL Command: <Display Name>) is created in the PATROL Console. The collected parameters for PSL Command instances are listed in the **SEN_MS_PSLCOMMAND** chapter.
Executing WBEM Queries

WBEM (Web-Based Enterprise Management) is a set of systems management technologies developed to unify the management of distributed computing environments that provides users with information about the status of local or remote computer systems.

Monitoring Studio can execute WBEM queries on your system and consolidate these queries under a single WBEM Query instance in your PATROL environment. It can also query the WBEM repository for class and instance information.

To execute a WBEM query

1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select WBEM Query from the drop-down list and click Next.

3. Select the type of Credentials that you want to use for the WBEM query:
   - **Agent’s Default Account**: Uses the PATROL Agent credentials for monitoring the localhost.
   - **System Credentials (default)**: Uses the credentials previously specified when creating the Host.
   - **Add new credentials...**: Lets you set new credentials for this specific query monitoring.
4. Set the **Common WBEM Settings**:
   - **Port**: Enter the port number you wish to use for the connection. By default, in standard environments, the port 5988 is used for non-encrypted data, while port 5989 is used for
encrypted data.

- **Encrypt data using the HTTPS protocol**: Select this option to encrypt the query with the HTTPS protocol.

5. Provide the following WBEM query Information:

- **Namespace**: Enter the WBEM namespace. A namespace is a logical group of related classes representing a specific technology or area of management (Example: root/cimv2)
- **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the query times out, the Status parameter will be set to 2 (Failed) and an alarm will be triggered.
- **WBEM query**: Enter your WBEM query (Example: SELECT DeviceID, Speed, Description, PermanentAddress, OperationalStatus FROM HPUX_EthernetPort).
- **Report execution errors in Group's CollectionErrorCount**: Select this option to have the CollectionErrorCount parameter of the Group reflect possible alerts triggered upon the query execution. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor, associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.

6. Click **Next**.

7. **Configure the Monitor settings**.

8. Click **Finish**. The corresponding WBEM Query instance (WBEM Query: <Display Name>) is created in the PATROL Console. The collected parameters for WBEM Query instances are listed in the **SEN_MS_WBEMQUERY** chapter.

You can now run **String Searches**, **Extract Numeric Values** and **Value Map** operations on the WBEM query result output.
Executing WMI Queries

WMI (Windows Management Instrumentation) is the Microsoft implementation of WBEM (Web Based Enterprise Management) that provides users with information about the status of local or remote computer systems.

Monitoring Studio can execute WMI queries on your system and consolidate them within your PATROL environment along with the technology monitoring under a single icon. It can also query the WMI repository for class and instance information. You can for example query the WMI that returns all the objects representing shut-down events from your desktop system.

This function is only available on Windows hosts. It can be remotely used from PATROL Agents running on Windows only.

To execute a WMI query

1. In the PATROL Console, right-click the Host or Monitor Group icon and select KM Commands > New > Monitor...
2. Select WMI Query from the drop-down list and click Next.

Please note that the System Credentials entered at the host level will be used to retrieve WMI Queries information.

3. Identify the host:
   - **Namespace**: Enter the WMI namespace. A namespace is a logical group of related classes representing a specific technology or area of management (Example: root\cimv2).
3. Specifying the Monitors Used

- **WQL Query**: Enter your query. (Example: SELECT DeviceID, Speed, Description, PermanentAddress, OperationalStatus FROM HPUX_EthernetPort). If you need help to build your WMI query, download WMI CIM Studio, which is one of the WMI Administrative tools on the Microsoft site.

4. Provide the following WMI Query Information:
   - **Timeout**: Specify the time in seconds after which the query will be stopped (Default: 30 seconds). If the query times out, the Status parameter will be set to 2 (Failed) and an alarm will be triggered.
   - **Report execution errors in Group’s CollectionErrorCount**: Select this option to have the CollectionErrorCount parameter of the Group reflect possible alerts triggered upon the query execution. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor, associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.

5. Click **Next**.

6. **Configure the Monitor settings**.

7. Click **Finish**. The corresponding WMI Query instance (WMI Query: <Display Name>) is created in the PATROL Console. The collected parameters for WMI Query instances are listed in the SEN_MS_WMIQUERY chapter.

You can now run **String Searches** and **Extract Numeric Values** on the WMI query result output.
Executing Nagios Plugins

Monitoring Studio provides powerful monitoring capabilities to support your existing custom scripts built for Nagios. Through **Nagios Plugin** Monitors, Monitoring Studio extends the monitoring coverage of your IT environment by supporting any existing Nagios Plugins. This Monitor is designed to execute any Nagios Plugin and monitor the results directly from your PATROL console without requiring a native Nagios environment.

To import an entire Nagios configuration, please refer to the Importing Nagios Configuration topic.

By default, alert actions are set at the **Nagios Plugin** Monitor level and shared by all **Nagios Performance Data** instances below. To set different alert actions for a specific Nagios Performance Data instance, you need to create a separate Nagios Plugin Monitor and then specify the alert actions that will only apply to all its related Nagios Performance Data instances.

To monitor Nagios plugins

1. In the PATROL Console, right-click the **Host** icon and select **KM Commands > New > Monitor...**
2. Select **Nagios Plugin** from the drop-down list and click **Next**.
3. Enter the path of the folder where the required Nagios plugin file is stored and click Next.

4. Configure the Nagios Plugin command line to be executed:

- **Credentials**: Select from the drop-down list the type of credentials that you want to use for the command line execution:
  - **Agent’s Default Account**: Uses the PATROL Agent credentials for executing Nagios plugin
commands on the local Nagios server. The Run this command locally option should also be selected to launch the command/plugin execution on the local PATROL Agent system.

- **System Credentials (default):** Uses the credentials previously specified when creating the Host.
- **Add new credentials...** Lets you set new credentials for this specific monitoring. Refer to the Setting Credentials chapter for detailed information.

- Enter the **Nagios Plugin command line** to execute.
- **Run this command locally (on <PATROL Agent's host>):** Select this option if you want the command line to be executed on the PATROL Agent’s system and not on the targeted host.
- **Timeout:** Enter the time in seconds after which the command will be stopped (Default: 30 seconds). If the timeout is reached, a new error is logged in CollectionErrorCount parameter of the monitored group, indicating that the command failed to execute properly.
- **Monitor Performance Data:** Check this option if you want each performance data returned by the plugin to be instantiated and monitored.
- **Report Unknown (3) service status:** Select how you want an unknown status to be reported in PATROL:
  - as a new error in the CollectionErrorCount parameter of the monitored Group.
  - as a specific value in the Status parameter.
- **Click Next.**

5. Select the parameters to be monitored. Refer to the SEN_MS_NAGIOSPERF chapter for details on parameters.

- Use the **Rescaling** option to rescale the value retrieved in order to have a more readable graph in the PATROL console:
  - **No Rescaling:** The value remains unchanged and is reported as collected.
• **Convert to MB**: The value is converted into megabytes.
• **Convert to GB**: The value is converted into gigabytes.
• **Convert to TB**: The value is converted into terabytes.
• **Divide by the value below**: The value is divided by the number you provide.

6. Click **Next**.
7. Specify a time after which Nagios Performance instances with missing performance data will be deleted:
   - **After number of hours**: Enter the number of hours after which the instance will be deleted.
   - **Never**: To prevent instances with missing performance data to be deleted.
   - **Immediately**: To delete instances with missing performance data immediately.

8. Click **Next**.
9. **Configure the Monitor settings**.
10. Click **Finish**. The corresponding Nagios Plugin instance (Nagios Plugin: <Display Name>) is created in the PATROL Console. The collected parameters for Nagios Plugin instances are listed in the **SEN_MS_NAGIOSPLUGIN** chapter.
Importing a Nagios Configuration

In addition to the Nagios Plugin Monitor that allows you to integrate and monitor specific Nagios plugins into your PATROL console, Monitoring Studio also enables you to import an entire Nagios configuration (nagios.cfg and icinga.cfg). Monitoring Studio detects and interprets Nagios groups, hosts and services configured on the Nagios server, imports them, and creates corresponding instances (Groups, Hosts and Nagios Plugin Monitors) in your Monitoring Studio environment.

⚠️ Nagios Plugin Monitors imported from a Nagios configuration (.cfg) use the source plugin files and should not be removed even after the import process is completed.

To import an entire Nagios configuration

1. Right-click the Monitoring Studio icon > KM commands > Configuration > Import Nagios Configuration
2. Enter the path of the folder containing the Nagios or Icinga 1 configuration file(s) you wish to import (Default: /usr/local/nagios/etc).

⚠️ Provide a valid Nagios configuration path to ensure a successful import. The full path to the Nagios configuration file can be provided as: /usr/local/nagios/etc/custom_nagios.cfg
3. The Nagios folder can contain multiple resources and cfg files that defines the objects in a chronological order. Each object will be used to build the Group, Host & Nagios Plugin Monitor structure in Monitoring Studio.

4. Click **Next** to create the objects in Monitoring Studio and proceed to the next step.

![Importing Nagios Configuration - Hosts List](image)

5. Click the **Hosts** button to view the list of the hosts for which you will need to provide information.

6. Click **Next** to continue.

7. Provide the Nagios credentials that Monitoring Studio will use to run all Nagios Plugin commands on the Nagios server:
3. Specifying the Monitors Used

Importing Nagios Configuration - Provide System Credentials

- **Username**: Enter the username to connect to the Nagios server.
- **Password**: Enter the password to connect to the Nagios server.
- **Currently Used by**: Lists all objects that are currently using system credentials.

8. Click **Next** to continue.
9. When some objects (Groups, Hosts, or Monitors) defined in the Nagios configuration already exist in the PATROL Agent configuration, the system prompts you to provide additional information.

Import Summary

- Total number of objects: 8
- Number of groups: 2
- Number of Nagios Plugins: 3

Objects are renamed with prefix: **NAGIOS**

- Pause the monitoring of the imported objects
- Click **Next**) to start the import process.
3. Specifying the Monitors Used

Importing Nagios Configuration - Conflicting Objects

- Click the **Conflicting Objects** button to display the complete list of conflicting objects.
  
  Then, select an option according to the action you wish to perform:
  
  - **Overwrite**: To overwrite existing objects. Overridden objects will be lost.
  
  - **Rename by adding this prefix**: To add a prefix to the existing objects' names that will help you identify the imported objects. For example, enter 'NAGIOS'. All concerned Groups, Hosts and Monitors found will be imported and named NAGIOS<existing-name>.

10. Click **Next** to continue.

11. Verify the information displayed in the **Import Summary**:

![Import Summary](image)

- **Pause the monitoring of the imported objects**: The imported configuration may need to be customized once created in PATROL. In the meantime, we recommend you to select this option to avoid potential unnecessary error messages and events.

12. Click **Next** to start the import process.

13. Once the import process is completed, a message is displayed and a full discovery is performed. Click **Finish** to close the wizard.

### Cloning a Host

Cloning a Host consists in taking an exact copy of the settings, credentials and related Monitors of a source Host and applying these settings to a target Host. Cloning is an easy way to produce alternate copies of existing monitoring infrastructure. This tool is particularly convenient for managing large environments where several Hosts need to be monitored rapidly and in a similar manner.
To clone a host

1. In the PATROL Console, right-click the Host icon you wish to clone and select KM Commands > Clone...

2. Provide a Hostname, IP address or FQDN (Fully Qualified Domain Name) for the cloned Host. Two Hosts with the same name cannot coexist in the same group.

3. Click OK.

4. The cloned Host instance is automatically created in the same group as the source Host and, by default, uses the same credentials. If the credentials of the cloned Host need to be changed, we recommend you to select the Pause the monitoring of the cloned host option to suspend the monitoring of the cloned Host while you modify its credentials and avoid unnecessary error messages and events.

To edit the credentials of a cloned Host, right-click the cloned Host and select KM Commands > Credentials > Edit Credentials... To edit any other settings, right-click the cloned Host and select KM Commands > Edit...
4. Processing a Monitor Result Output

Once a Monitor instance is created in the PATROL Console, you can process the result output to retrieve specific information or to detect the presence of strings or numeric values that should or not be present.

Prior to perform analysis of a Monitor's output, you may have to pre-process the data in order to make it manageable by the search/extraction tools. Typically, the usual procedure consists in:

- **Pre-Processing the Monitor Output**
- **Split the Monitor Output**
- **Analyzing the Monitor Output:**
  - Searching for a specific string
  - Extracting a numeric value
  - Mapping Values

The features described in this chapter are exclusively available for the following Monitors:

- Command Line (SEN_MS_COMMANDLINE)
- Database Query (SEN_MS_DBQUERY)
- Dynamic Instance (SEN_MS_DYNAMIC)
- File (SEN_MS_FILE)
- Multi-Parameter Formula (SEN_MS_FORMULA)
- Nagios Plugin (SEN_MS_NAGIOSPLUGIN)
- PSL Command (SEN_MS_PSLCOMMAND)
- SNMP Polling (SEN_MS_SNMPOLLING)
- Text Pre-Processing (SEN_MS_TRANSFORM)
- WBEM Query (SEN_MS_WBEMQUERY)
- Web Request (SEN_MS_WEBSERVER)
- WMI Query (SEN_MS_WMISERVER)
4.1 (Optional) Pre-Processing a Monitor Output

Because **String Search**, **Numeric Value Extractions** and **Value Map** operations cannot always be performed from the raw result output of instances, you may need to first process the multi-line, XML, JSON, or HTML content by using the **Text Pre-Processing** tool. This tool transforms the result output to simpler format that can be easily parsed with the **String Search**, **Numeric Values Extraction**, and **Value Map** tools.

**Example of converting multi-line records to single lines**

The "ipconfig /all" command under Windows reports various information about each network card, and each "paragraph" is about one network card:

![Image of ipconfig /all output]

The aim here is to detect any disconnected card. So we add a monitoring instance for the command `ipconfig /all`. But as the text is in paragraphs, a direct String Search will not get the desired result in this case – which is why we run the Text Pre-Processing tool to convert the multi-line text to single lines.

In the screenshot below, the "ipconfig / all" command is executed and its output is pre-processed to transform its paragraphs into single lines, which in turn enables an efficient parsing with a String Search that looks for "disconnected" network cards.

![Screenshot of Text Pre-Processing tool](image-url)
To convert multi-line records into single line:

1. In the PATROL Console, right-click the monitor icon (File, Command Line, Web Request, etc.) and select KM Commands > New > Text Pre-Processing...

2. (Optional) If you need to filter the Monitor output to remove unwanted text before parsing it, select which lines of the result should be scanned:
   - **all lines/only the following line numbers**: Specify whether all the lines of the result should be processed or enter the list of line numbers you wish to process separated by a semicolon (;). Lines are specified as follows:
     - x, y: line x and line y
     - x-y: all lines from x to y inclusive
     - x: Only line x
     - x-: all lines from x to the end of the file inclusive
   - **contain/do not contain**: Enter the string or regular expression to look for, and specify whether or not it should be contained in the selected lines.
3. Select **Convert multi-line records into single line** and click **Next**.

![Text Pre-Processing Wizard: Convert multi-line records into single lines — Start/End of line definition](image)

4. Define the first and/or last lines of the paragraphs:
   - **This RegExp marks the beginning of a new record**: Enter the regular expression or string that marks the beginning of a new record. Please note that the regular expression can match with any part of the first line of each paragraph. If needed, include the first line in the result.
   - **This RegExp marks the end of a record**: Enter the regular expression or string that marks the end of a new record. Please note that the regular expression can match with any part of the first line of each paragraph. If needed, include the last line in the result.

   The entire line containing the specified RegExp will be returned.

   - The regular expression `^$` can be used to match an empty line.
   
   - You can specify a regular expression that only identifies the beginning of a new paragraph (record). In this case, Monitoring Studio skips the content until it finds a line matching with the specified criteria.
     The text that follows this line (and optionally including this first line) is concatenated in a single line by using the specified separator, until Monitoring Studio finds another line that matches with the specified regular expression. Each line in the original content that matches with this regular expression produces a new line in the result content. The same is true for the regular expression that marks the end of a paragraph (or record).

   - If you specify both regular expressions to identify the beginning and the end of a record, Monitoring Studio will only take into account the text content that is in between lines that matches these regular expressions (i.e. between the start line and the end line). Lines in the original text between a line matching the end marker and the next line matching the beginning marker will be skipped and not integrated in the text result.
   
   - Specify the character to be used as a separator.
5. Click **Next**.
6. **Configure the Monitor settings**.
7. Click **Finish**. The output will be displayed at the next discovery by the `TransformResult` parameter.

You can now **perform string searches, extract numeric values, map values** or **build dynamic objects** from the `TransformResult` parameter output.

**To convert XML to CSV**

1. In the PATROL Console, right-click the Monitor instance (File, Command Line, Web Request, etc.) and select **KM Commands > New > Text Preprocessing...**

2. (Optional) If you need to filter the Monitor output to remove unwanted text before parsing it, select which lines of the result should be scanned:
   - **all lines/only the following line numbers**: Specify whether all the lines of the result should be processed or enter the list of line numbers you wish to process separated by a semicolon (;). Lines are specified as follows:
     - x, y: line x and line y
     - x-y: all lines from x to y inclusive
     - x: Only line x
     - x-: all lines from x to the end of the file inclusive
   - **contain/do not contain**: Enter the string or regular expression to look for, and specify whether or not it should be contained in the selected lines.
3. Select **Convert XML to CSV (Comma-Separated Values)** and click **Next**.

![Image of Convert XML to CSV wizard](image)

4. Enter the XML tag that defines the record, sub-objects, and properties, the character to be used as a separator and click **Next**.

5. **Configure the Monitor settings**.

6. Click **Finish**. The output will be displayed at the next discovery by the *TransformResult* parameter.

You can now **perform string searches, extract numeric values, map values** or **build dynamic objects** from the *TransformResult* parameter output.
To convert JSON to CSV

1. In the PATROL Console, right-click the Monitor instance (File, Command Line, Web Request, etc.) and select **KM Commands > New > Text Preprocessing...**

   ![Image of Text Pre-Processing Wizard]

   **Text Pre-Processing wizard**

   This wizard helps you setup a text pre-processing on "Flat File: C:\Windows\System32\FNTCACHE.DAT".

   **From the result, select**
   
   - all lines
   - Process only the lines that
     - contain

   **Select the conversion you wish to perform on the selection:**
   - Convert JSON to CSV (Comma-Separated Values)

2. (Optional) If you need to filter the Monitor output to remove unwanted text before parsing it, select which lines of the result should be scanned:
   - **all lines/only the following line numbers**: Specify whether all the lines of the result should be processed or enter the list of line numbers you wish to process separated by a semicolon (;). Lines are specified as follows:
     - x, y: line x and line y
     - x-y: all lines from x to y inclusive
     - x: Only line x
     - x-: all lines from x to the end of the file inclusive
   - **contain/do not contain**: Enter the string or regular expression to look for, and specify whether or not it should be contained in the selected lines.
3. Select Convert JSON to CSV (Comma-Separated Values) and click Next.

4. Enter the JSON entry key that defines the record, a list of properties (delimited by a semi-colon) and click Next.

5. List the properties defined for the provided JSON entry. Separate each property with a semi-colon (;).

6. Specify which separator to use to isolate each property in the output file, such as dot (.), dash (-), double semi-colon (;;), comma (,), pipe (| ), plus sign (+), etc.

7. Configure the Monitor settings.

8. Click Finish. The output will be displayed at the next discovery by the TransformResult parameter.

You can now perform string searches, extract numeric values, map values or build dynamic objects from the TransformResult parameter output.
To convert JSON to Flat Map

1. In the PATROL Console, right-click the Monitor instance (File, Command Line, Web Request, etc.) and select KM Commands > New > Text Preprocessing...

   ![Text Pre-Processing Wizard — Welcome Page](image)

   - **All lines/only the following line numbers**: Specify whether all the lines of the result should be processed or enter the list of line numbers you wish to process separated by a semicolon (;). Lines are specified as follows:
     - \(x, y\): line \(x\) and line \(y\)
     - \(x-y\): all lines from \(x\) to \(y\) inclusive
     - \(x\): Only line \(x\)
     - \(x-\): all lines from \(x\) to the end of the file inclusive

   - **Contain/do not contain**: Enter the string or regular expression to look for, and specify whether or not it should be contained in the selected lines.

2. (Optional) If you need to filter the Monitor output to remove unwanted text before parsing it, select which lines of the result should be scanned:
   - **All lines/only the following line numbers**: Specify whether all the lines of the result should be processed or enter the list of line numbers you wish to process separated by a semicolon (;). Lines are specified as follows:
     - \(x, y\): line \(x\) and line \(y\)
     - \(x-y\): all lines from \(x\) to \(y\) inclusive
     - \(x\): Only line \(x\)
     - \(x-\): all lines from \(x\) to the end of the file inclusive
   - **Contain/do not contain**: Enter the string or regular expression to look for, and specify whether or not it should be contained in the selected lines.
3. Select **Convert JSON to Flat Map (Property=Value Pairs)** and click **Next**.

![Image of the Text Pre-Processing Wizard: Convert JSON to Flat Map — Extraction Confirmation](image)

4. Click **Next**.
5. **Configure the Monitor settings**.
6. Click **Finish**. The output will be displayed at the next discovery by the `TransformResult` parameter.

You can now **perform string searches**, **extract numeric values**, **map values** or **build dynamic objects** from the `TransformResult` parameter output.
To extract text from HTML

1. In the PATROL Console, right-click the Monitor instance (File, Command Line, Web Request, etc.) and select **KM Commands > New > Text Preprocessing**...

![Text Pre-Processing Wizard — Welcome Page](image)

2. (Optional) If you need to filter the Monitor output to remove unwanted text before parsing it, select which lines of the result should be scanned:
   - **all lines/only the following line numbers**: Specify whether all the lines of the result should be processed or enter the list of line numbers you wish to process separated by a semi-colon (;). Lines are specified as follows:
     - x, y: line x and line y
     - x-y: all lines from x to y inclusive
     - x: Only line x
     - x-: all lines from x to the end of the file inclusive
   - **contain/do not contain**: Enter the string or regular expression to look for, and specify whether or not it should be contained in the selected lines.
3. Select Extract text from HTML (Comma-Separated Values) and click Next.

4. Click Next.
5. Configure the Monitor settings.
6. Click Finish. The output will be displayed at the next discovery by the TransformResult parameter.

You can now perform string searches, extract numeric values, map values or build dynamic objects from the TransformResult parameter output.

To process text through an external command

This option should be selected if text inputs (files, output of commands, Web requests, etc.) need to be transformed in a special way in order to be parsed with Monitoring Studio’s String Searches and Numeric Value Extractions. If the built-in text transformation features of Monitoring Studio cannot handle such "specially formatted" text, you will have to process the content through a custom script or utility that performs the required transformation. The main advantage of processing the text through an external command feature is that it enables you to customize the processing of almost any source of information important to your technology.

1. In the PATROL Console, right-click the Monitor instance (File, Command Line, Web Request, etc.) and select KM Commands > New > Text Preprocessing...
2. (Optional) If you need to filter the Monitor output to remove unwanted text before parsing it, select which lines of the result should be scanned:

- **all lines/only the following line numbers**: Specify whether all the lines of the result should be processed or enter the list of line numbers you wish to process separated by a semicolon (;). Lines are specified as follows:
  - x, y: line x and line y
  - x-y: all lines from x to y inclusive
  - x: Only line x
  - x-: all lines from x to the end of the file inclusive

- **contain/do not contain**: Enter the string or regular expression to look for, and specify whether or not it should be contained in the selected lines.
3. Select **Text processing through an external command** and click **Next**.

4. Specify the command to be executed to transform the text:
   - **Command to be executed**: Enter the command to be executed on the localhost where your PATROL Agent is installed. The principle is very similar to the "pipe" mechanism of the UNIX shell except that the content is not passed directly but is stored in a temporary file and then the result needs to be stored in another temporary file. Hence the command line you specify needs to take the `%{INPUTFILE}` macro as an argument (the `%{INPUTFILE}` macro is replaced by the real temporary input file location at run time) as well as `%{OUTPUTFILE}`.
   - **This command is executed on the host where the PATROL Agent is installed, not on the target Host.**

   - **The output of the command must match this RegExp to be considered as successful**: Enter a RegExp to avoid typical path problems such as getting "... not found" error messages instead of the properly transformed text

   - **If your command line redirects its output to `%{OUTPUTFILE}`, the validation regular expression is likely to fail because the standard output is empty and thus matches with nothing. Use a validation regular expression only if your command line is able to produce both the `%{OUTPUTFILE}` and some text to its standard output.**

   - **Command lines are executed locally, therefore it is not possible to use the awk command when monitoring a UNIX/Linux system from Windows.**

   - **Macros are case sensitive and should then always be written in uppercase.**

   - **Click Next.**
5. **Configure the Monitor settings.**

6. Click **Finish**. The output will be displayed at the next discovery by the `TransformResult` parameter.

You can now **perform string searches, extract numeric values, map values or build dynamic objects** from the `TransformResult` parameter output.
4.2 Splitting the Monitor Output

The **Dynamic Object Builder** Monitor splits the return output of objects and applies a set of user-defined rules on this return output data.

Once you have created a **Dynamic Object Builder** instance, Monitoring Studio:

- applies the user-defined set of rules to the output data
- extracts each matching line and creates a dynamic object for each of them in the PATROL Console according to the user-defined naming
- groups all dynamic objects under a dynamic container to facilitate the management of all the dynamic objects. The **Hide/Show Template KM Command** enables you to display or not a dynamic container content.

Dynamic objects can be analyzed with **String Search**, **Numeric Value Extraction** and **Value Map**.

**To create a dynamic object:**

1. In the PATROL Console, right-click one of the following monitoring features:
   - Command Line Analysis
   - Database Query Analysis
   - File Monitoring and Analysis
   - Multi-Parameter Formula
   - Nagios Plugins
   - PSL Commands Analysis
   - SNMP Polling
   - Text Pre-Processing
   - WBEM Query Analysis
   - Web Request Analysis
   - WMI Query Analysis
2. Select KM Commands > New > Dynamic Object Builder...

![New Dynamic Object Builder Window](image)

- **Managed System:** PC-IGUNIX_3181

- **Dynamic Instances Behavior**
  - **Keep only lines matching the regular expression below:**
  - **Exclude lines matching the regular expression below:**

- **Do not trigger any alert** when objects are missing.

- **Delete missing objects:**
  - **as soon as they are missing**

---

3. Define the rules to apply to the output data:
   - **Keep only lines matching the regular expression below:** Specify the regular expression to be searched for by the Dynamic Object Builder. Each matching line is displayed in a separated Dynamic Item automatically added in the PATROL Console under the Dynamic Container icon. Non-matching lines are discarded. If nothing is specified, all lines are kept.
   - **Exclude lines matching the regular expression below:** Specify the regular expression to be searched for by the Dynamic Object Builder. Each matching line will be ignored.
   - Define if and how Monitoring Studio must trigger an alert when the Dynamic Item, containing the extracted line, is missing from one collect to another:
     - **Do not trigger any alert:** Select this option if you do not wish Monitoring Studio to trigger any alert when objects are missing
     - **Trigger an INFORMATION:** Select this option if you wish Monitoring Studio to trigger an Information alert when objects are missing
     - **Trigger a WARNING:** Select this option if you wish Monitoring Studio to trigger a Warning alert when objects are missing
     - **Trigger an ALARM:** Select this option if you wish Monitoring Studio to trigger an Alarm alert when objects are missing

   If you have chosen to trigger an alert, a **Status** parameter will automatically be activated and displayed under the Dynamic Item in the PATROL Console to indicate that the Dynamic Item is missing. Click the **Status** parameter to open a graph pane displaying the status history of the Dynamic Item.

   - **Delete missing objects:** Set this option according to how you want Monitoring Studio to delete missing objects in the console. The available options are:
     - **Never:** Monitoring Studio never deletes missing objects in the console. They will always
be present.

- **As soon as they are missing:** Monitoring Studio deletes missing objects from the console as soon as their absence is discovered. If an alarm is configured, Monitoring Studio will trigger the alarm before deleting the object.
- **When they are missing... in a row:** Monitoring Studio deletes missing objects after they have been missing a specific number of times in a row.

4. Click **Next**.
5. Configure the **Dynamic Instances Monitor** settings templates:
   - **Internal ID Template:** PATROL internal identifier of the template.
   - **Display Name Template:** Label that will be displayed in the PATROL Console for this template. Indicate the column number in which the name of the object will be found by Monitoring Studio (e.g.: %1 for the first column).

   ![Dynamic Object Builder](image)

   **Dynamic Instances Monitor Settings Templates**

<table>
<thead>
<tr>
<th>Internal ID Template:</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{{Column Number}}</td>
</tr>
</tbody>
</table>

   **Display Name Template:**

   Dynamic Item: %{{Column Number}}

   Enter %{{Column Number}} to customize the object ID or Display Name. Column separator(s) can be selected at the next step.

   **If several lines with the same ID are found:**
   - **Keep the first line only:** the first matching line found will be reported, all other matching lines with the same ID subsequently found will be ignored.
   - **Keep the last line only:** the last matching line found will be reported, all other matching lines with the same ID previously found will be ignored.
   - **Group these lines under the same instance:** all matching lines with the same ID will be reported.
   - **Report lines with same ID in Group's CollectionErrorCount:** Select this option to have the CollectionErrorCount parameter of the Group reflect possible alerts triggered when the Dynamic Object instance reports multiple lines with the same ID. The CollectionErrorCount parameter of the Group reports on the collection errors of any Monitor associated to the group, for which this option is available and selected, providing a global view of the collection errors for the whole Group.
6. Define the way the **Dynamic Object Builder** detects columns of output data:

- Select or de-select applicable separators. For custom separators, type the character(s) one after the other in the **Others** field.

- Indicate how consecutive separators will be considered:
  - **Consecutive separators must be treated as a single one** (useful for data separated by comma, such as in a .csv file): Typically, it indicates that consecutive separators must be treated as a single separator.
  - **Consecutive separators mean empty columns** (useful for data separated by comma, such as in a .csv file): Each separator is treated as an individual column separator and the column is considered empty.

- Indicate how quotes will be considered:
  - **Do not interpret quotes**: All applicable separators will be considered as a column separator even when enclosed in quotes.
  - **Consider text in "double quotes" as a single column**: Text enclosed in double quotes will be considered as a single column. Any separator found within double quotes will not be considered as a column separator.
  - **Consider text in 'single quotes' as a single column**: Text enclosed in single quotes will be considered as a single column. Any separator found within single quotes will not be considered as a column separator.

7. Click **Next**.
8. **Configure the Monitor settings**.
9. Click **Finish**. The corresponding instance Dynamic Instance <parent instance name> is created in the PATROL interface.
Dynamic Object Templates

Dynamic Object Templates

String Search, Numeric Value Extractions and Value Map Monitors can help you analyze the Dynamic output of other Monitors (a command, a web page, the result of a SQL query, or WBEM query, etc.). In order to apply this feature to all the Dynamic monitor outputs in a single operation, the Dynamic Object Builder creates Templates instances of the feature. While the standard String Search, Numeric Value Extractions and Value Map functionality applies to a single return output, templates apply to all the Dynamic monitor outputs grouped in the attached Dynamic Container.

To create Dynamic Object Templates

1. In the PATROL Console, right-click the Dynamic Object for which you want to create a template and select KM Commands > New > String Search Template/Numeric Value Extraction Template or Value Map Template...
2. The wizard corresponding to the option you have selected is displayed. The procedure is the same as the one performed for the standard version of these Monitors. Simply follow the steps of the wizard and provide the required information.
3. Once you have clicked Finish, the String Search Template/Numeric Value Extraction Template or Value Map Template is created in the PATROL Console: String Search Template/Numeric Value Extraction Template or Value Map Template <Display Name> (Template).

Hiding/Showing Dynamic Templates and Dynamic Object Containers

In order to simplify the visual organization of dynamic instances and dynamic templates, you can choose to show or hide these instances in the PATROL console by using the Dynamic Object > KM Commands > Hide/Show Template feature.
4. Processing a Monitor Result Output

4.1 Processing a Monitor Result Output

**Detailed View**

Dynamic instances and dynamic templates are shown.
All dynamic instances for the selected dynamic object are visually attached to the **Dynamic Object Container** associated to the **Dynamic Object Monitor instance**. All existing string search, numeric value extraction dynamic templates or value map templates are visible in the Console.

**Simplified View**

Dynamic instances and dynamic templates are hidden.
All dynamic instances for the selected dynamic object are visually attached to the **Dynamic Object Monitor instance**. Any existing string search, numeric value extraction dynamic template or value map template is also hidden.

4.3 Analyzing the Monitor Output

**Searching for a Specific String**

The **String Search Monitor** allows you to run fast and powerful searches for strings on the Monitors that you previously configured (flat, or log files, output of a Web request, or a database query, OID content, etc).

Please note that the **String Search Monitor** works a bit differently on "running sources" (Log files and never-ending command lines) than on flat sources (flat files, command lines, Web requests, etc.):

- On "running sources" (Log files and never ending command lines):
  - the strings are searched only in new lines since the last polling. For a string search in a running source, two graphs are built: **Number of matches since the last acknowledgment** and **Number of matches per minute since the last polling**.
  - you can specify auto-acknowledging strings that will automatically reset the graph to the "number of matches".
- On "flat sources" (any other source):
  - the strings are searched in the entire source every time (the whole file, the whole standard output, the whole HTTP response, the whole dataset). For a String Search in a flat
source, one graph is built: **Number of matches at the current polling**.
- you cannot specify auto-acknowledging strings since the parameter is recalculated from "0" at each polling.
- you can specify where information should be searched in the file (n lines, pre-filter, etc.).

## To search for a specific string

1. If the file to be parsed is a multi-line, XML, JSON, or HTML content, **pre-process its content**.
2. In the PATROL Console, right-click the Monitor icon (file, Web request, etc.) and select **KM Commands > New > String Search**...

   ![String Search](image)

   **Searching for a Specific String - Search Definition Page**

3. Specify the information to search for:
   - **Search for lines that**:
     - **Contain/do not contain**: You can enter up to two strings (regular expressions) to look for, and decide whether or not those strings should be contained in the line.
     It is also possible to specify if the two strings should be found together (AND), or if only one of the two strings is sufficient (OR).
     - **Select where to search**: For each entered string, you can specify where in the line to search for the string:
       - **Anywhere in the line** (default)
       - **At the following character offset**: if you choose to search for the string from a character offset in the line, you must specify the offset in this field. Character offset is nothing but the character number. For example, to search for a string that starts from the seventh character in the line, you enter the digit 7 as the character offset.
       - **In the following column number**: Enter the column number
       - **Skip blank lines**: Select this option to have Monitoring Studio ignore blank lines. This option is particularly useful when searching for lines that do not contain a specific string, as blank lines would match this search criteria.
- (Log File Monitoring) **Report Matching Lines:**
- **since last acknowledge (incremental):** Select this option to count the lines matching the String Search since the last time the MatchingLineCount parameter was reset or the PATROL Agent started.
- **in the current collect only:** Select this option to count the lines matching the String Search during the current collect.

a) For String Searches on a flat source (flat file, command line, Web request, etc.), the following dialog box is displayed:

![String Search Dialog Box]

- Select which lines of the source should be scanned.
- **Search for Strings in all lines:** The string(s) will be searched for in all the lines of the specified source.
- **Search for Strings only in the following line numbers:** Enter the list of line numbers you wish to scan separated by ‘;’. Lines are specified as follows:
  - x, y: line x and line y
  - x-y: all lines from x to y inclusive
  - x: Only line x
  - x-: all lines from x to the end of the file inclusive

**Example**

Here are some examples of formats that can be used when specifying line numbers:
"4;6;8", or "3-", or ".-5", or "1;3-5;7-9"

- Click **Next**.
b) For String Searches on a running source (log file and never-ending command lines) the following dialog box is displayed:

- Set the automatic acknowledgment of alerts:
  - **Acknowledge alert(s) if the string below is found**: Check the box to acknowledge the alert.
    - Specify the string
    - Select where to search: specify the location of the string
  - **Acknowledge alert(s) after**: Check this box and then specify the time in seconds after which the alerts will be acknowledged. Default is 120 minutes.

- **When Acknowledging**: Specify the action you wish Monitoring Studio to perform when acknowledging an alert:
  - Select the **Reset the MatchingLineCount parameter to zero** option to have Monitoring Studio automatically reset the counter of the `MatchingLineCount` parameter to zero.
  - Select the **Decrease the MatchingLineCount parameter by one** option to have Monitoring Studio automatically decrease by one the value of the counter of the `MatchingLineCount` parameter. Use this option if you need the solution to acknowledge each event counter and get a close follow-up on the log activity.
4. If the option **In the following column number** was previously selected, you will have to specify the column separator to be considered to identify the relevant column:

- Select or de-select applicable separators. For custom separators, type the character(s) one after the other in the **Others** field.
- Indicate how **consecutive separators** will be considered:
  - **Consecutive separators must be treated as a single one** (useful for data separated by blank spaces): Typically, it indicates that consecutive separators must be treated as a single separator.
  - **Consecutive separators mean empty columns** (useful for data separated by comma, such as in .csv file): Each separator is treated as an individual column separator and the column is considered empty.
- Indicate how **quotes** will be considered:
  - **Do not interpret quotes**: All applicable separators will be considered as a column separator even when enclosed in quotes.
  - **Consider text in "double quotes" as a single column**: Text enclosed in double quotes will be considered as a single column. Any separator found within double quotes will not be considered as a column separator.
  - **Consider text in 'single quotes' as a single column**: Text enclosed in single quotes will be considered as a single column. Any separator found within single quotes will not be considered as a column separator.
- Click **Next**.
5. **Configure the Monitor settings.**

6. Click **Finish**. The corresponding String Search instance (String Search: <Display Name>) is created in the PATROL Console. The collected parameters for String Search Monitors are listed in the **SEN_MS_STRING** chapter.

### Extracting Numeric Values

Problems with a technology (application, server, device, etc.) are not always as simple as a sentence explaining that an "error has occurred." Sometimes, a technology reports its health by providing critical numbers, like a queue length, a processing time, a utilization percentage, etc. These numbers may be reported by the technology through its files, in the output of commands, in a database, in a Web page, etc. Monitoring Studio extracts these values and reports them as graphs in the PATROL Console. This feature is called **Numeric Value Extraction** as its purpose is to extract numeric values from a text input (the output of a command, a Web page, the result of a SQL query, or WBEM query etc.).

All you need to do is indicate how to find the numeric value(s) within the Monitor. The basic mechanism is:

1. Specify a Monitor
2. Specify with a regular expression the location of the numeric value within the searched lines
3. Indicate the numeric value’s position in these lines i.e. before/after the string; column number etc.
4. Indicate which numbers are to be considered if several lines contain the searched numeric value: first value, last value, calculate average; highest value; lowest value.

The numeric value searching function works a bit differently on "running sources" (Log files and never-ending command lines) than on flat sources (flat files, command lines, Web requests, etc.):

- **On "running sources"** (Log files and never-ending command lines); the numeric values are searched only in new lines since the last polling.
- **On "flat sources"** (flat files, command lines, Web requests, database queries), the numeric values are searched in the entire source (the whole file, the whole standard output, the whole HTTP response, the whole data-set).

If several values are found, it is possible to select which value should be kept: the last value found, the average of all values, the minimum or maximum or a total of all values. A graph is then built with this value.
To extract a numeric value

1. If the file to be parsed is a multi-line, XML, or HTML content, pre-process its content.
2. In the PATROL Console, right-click the Monitor icon (file, Web request, etc.) and select KM Commands > New > Numeric Value Extraction...

3. Specify which lines are to be searched for the numeric value. The information to enter subsequently depends on this first choice. There are three modes to choose from:
   - **All lines**: All lines of the text will be scanned for the Numeric value search. There is no need to provide further information.
   - **The Line numbers below**: Only specific line numbers will be scanned. If this mode is selected, the line numbers must be specified in the text field below. Line numbers are specified as follows:
     - x, y: line x and line y
     - x-y: all lines from x to y inclusive
     - x: only line x
     - x-: all lines from x to the end of the file inclusive

   **Example**
   Here are some examples of formats that can be used when specifying line numbers: "4;6;8", or "3-", or "-5", or "1;3-5;7-9"

   If the Skip blank lines option is selected, empty lines will be ignored in the line-count.
   - **Lines matching the regular expression below**: The lines scanned will be the lines matching the regular expression specified in the field.
4. Click **Next**.

5. Indicate the position of the numeric values to be found:
   - **The numeric values are located**: There are four different modes to choose from and the information to enter depends on this choice.
     - **After the string specified below**: If the numeric value is located after the string to be entered in the text field.
     - **Before the string specified below**: If the numeric value is located before the string to be entered in the text field.
     - **At the character offset specified below**: The value should be at a specific character offset in the line. Enter the offset number in the box. Monitoring Studio will look for the numeric value at that exact character offset in the line. If no numeric data is found, no value will be collected.
     - **In the column number specified below**: The previously selected lines contain several columns identified by a separator character. Enter the column number that should contain the value.
6. Click **Next**.

7. If the option **in the column number specified below** was previously selected, you will have to indicate how columns are separated:

   - **Select or de-select applicable separators.** For custom separators, type the character(s) one after the other in the **Others** field.

   - **Indicate how consecutive separators will be considered:**
     - **Consecutive separators must be treated as a single one** (useful for data separated by blank spaces): Typically, it indicates that consecutive separators must be treated as a single separator.
     - **Consecutive separators mean empty columns**: Each separator is treated as an individual column separator and the column is considered empty. The above two options are especially useful for data separated by blanks.

   - **Indicate how quotes will be considered:**
     - **Do not interpret quotes**: All applicable separators will be considered as a column separator even when enclosed in quotes.
     - **Consider text in "double quotes" as a single column**: Numeric values enclosed in double quotes will be considered as a single column. Any separator found within double quotes will not be considered as a column separator.
     - **Consider text in 'single quotes' as a single column**: Numeric values enclosed in single quotes will be considered as a single column. Any separator found within single quotes will not be considered as a column separator.

   - **Click **Next**.**
8. Specify the numeric values to be used and their expected format:

- From the **Expected format** drop-down list, select how the numeric values are formatted. This option allows you to extract numeric values that use blank, comma, or points as decimal and thousand separators. Monitoring Studio will look for a number in the location specified previously. When a character that does not match the expected format is found, the parsing will stop. So if the expected format is 1,000 and Monitoring Studio finds 1 000, the returned value will be 1.
- Check the box **Support negative values** if the numeric values that you are looking for can be negative. The '-' sign will be considered as an acceptable character when looking for the number.
- Check the **Convert units** option to convert the extracted numeric value from kilobytes, megabytes, gigabytes, terabytes to bytes or from days, hours, minutes to seconds. (For example: 2.1 KB will be converted to 2150 bytes and "1 Hr" will be converted to 3600 sec).

**Indicate which value should be used to set the parameter value and build the graph in the PATROL Console if several values are found in the searched text. The possible values are:**

- **Keep the first**: Only the first value found will be kept.
- **Keep the last (Default)**: Only the latest value found will be kept.
- **Calculate the average**: The values found will be averaged and only the result will be kept.
- **Select the lowest value**: Only the lowest value found will be kept.
- **Select the greatest**: Only the greatest value found will be kept.
- **Calculate the sum**: The values found will be summed and only the result will be kept.

- Click **Next**.
9. Specify the post-processing action to be performed on the monitored value:

- **Raw Value**: The values are reported as collected. No post-processing is performed.
- **Delta**: The value reported corresponds to the difference between values collected during two consecutive polling.
- **Delta per second/minute/hour**: The value reported corresponds to the result of the division of the Delta by the number of seconds/minutes/hour elapsed between the collection times.
- Select the option **Discard negative Delta values** if you do not want negative values to be reported on the graph.
- Use the **Rescaling** option to configure Monitoring Studio to rescale the value that is being extracted in order to have a more readable graph in the PATROL Console. The available rescaling options are:
  - **No rescaling**
  - **Divide the value by** a value that then gives you the reading in terms best suited to you
  - **Multiply the extracted value by** a constant factor

This can be useful if you extract numeric values in bytes but prefer to show a graph in megabytes. In such a case, you would divide the values by 1048576 (1024*1024).

- Click **Next**.

10. **Configure the Monitor settings**.
11. Click **Finish**. The corresponding Numeric Value Extraction instance (Numeric Value Extraction: `<Display Name>`) is created in the PATROL Console. The collected parameters for Numeric Value Extraction Monitors are listed in the **SEN_MS_NUMBER** chapter.

**Warning**: If Monitoring Studio does not find a number at the specified location, no numeric value will be extracted and the PATROL graph will not be refreshed (space characters are ignored).
Mapping Values

When monitored technologies report problems or errors, they do not always provide a clear portrait of what the issue consists in or about its severity. IT administrators may be provided with long files or command outputs from which they are expected to extract valuable information and establish a link between this information and a clear status about the nature and the level of gravity of the problem. Unfortunately, few technologies report problems the same way, which makes their task difficult and time consuming. Monitoring Studio provides an easy way to map the standard PATROL status to any value from a monitored technology output.

The Value Map Monitor allows you to extract values from result outputs generated by other Monitors and map up to 25 of these values to user-defined status that match your specific needs. Based on these statuses, it becomes easy to apply thresholds and have Monitoring Studio trigger PATROL standard alerts directly in your PATROL environment.

The Value Map Monitor only requires a few steps to set the search requirements:

1. Specify the source
2. Delimit the value location area (several options are available to cover a large range of possibility)
3. Provide the value to status mapping information

Monitoring Studio reports the Value Map Monitor instance output as graphs in the PATROL Console.
To map values to user-defined status

1. In the PATROL Console, right-click a Monitor instance icon (file, command line, Web query, etc.) and select **KM Commands > New > Value Map**...

![Value Map Interface](image)

- **Value Map — Source Selection**

2. Specify the line(s) from which you wish to extract values:
   - **all lines**: All lines of the source file will be scanned. There is no need to provide further information.
   - **the line numbers below**: Only specific line numbers from the source file will be scanned. If this mode is selected, the line numbers must be specified in the text field below. Line numbers are specified as follows:
     - **x, y**: line x and line y
     - **x-y**: all lines from x to y inclusively
     - **x**: only line x
     - **x-**: all lines from x to the end of the source file inclusively

   **Example**
   Here are some examples of formats that can be used when specifying line numbers: "4;6:8", or "3-", or ".5", or "1:3-5;7-9"

   - **Lines matching the regular expression below**: The lines scanned from the source file will be the lines matching the **regular expression** specified in the field.
   - If the **Skip blank lines** option is selected, empty lines will be ignored during the search process.
3. Click **Next** to continue.
4. Indicate the position of the values to be found:
   - **The value is located**: There are four different modes to choose from and the information to enter depends on this choice.

   - **after the string specified below**: If the value specified at the next step is located after the specified string. The search stops when the end of field character (or end of line) is found.
   - **before the string specified below**: If the value specified at the next step is located before the specified string. The search stops when the end of field character (or end of line) is found.
   - **at the character offset specified below**: If the value specified at the next step is located at the specified offset character. The parsing stops when the end of field character (or end of line) is found.
   - **in the column number specified below**: The previously selected lines contain several columns identified by a separator character. Enter the column number that should contain the value.
5. Click Next to continue.

6. If you have previously selected the after/below the string specified below or the at the character offset specified below option, you need to specify the character that marks the end of field (line) and stops the search for the value.

- Select or de-select applicable character and/or enter custom character(s) in the Others field.
- Indicate how quotes will be considered:
  - **Do not interpret quotes**: All applicable separators will be considered as marking the end of field and the parsing will stop even when enclosed in quotes.
  - **Consider text in "double quotes" as a single column**: Any separator found within double quotes will not be considered as an end of field character. The parsing will continue until an end of field character is found outside the double quotes, or until the end of the line is reached.
  - **Consider text in 'single quotes' as a single column**: Any separator found within single quotes will not be considered as an end of field character. The parsing will continue until an end of field character is found outside the single quotes, or until the end of the line is reached.
7. If you have previously selected the option **in the column number specified below**, you need to specify how columns are separated:

- Select or de-select applicable separators. For custom separators, type the character(s) one after the other in the **Others** field.

- Indicate how consecutive separators will be considered:
  - **Consecutive separators must be treated as a single one** (useful for data separated by blank spaces): Consecutive separators are treated as a single separator.
  - **Consecutive separators mean empty columns**: Each separator is treated as an individual column separator and the column is considered empty.

- Indicate how **quotes** will be considered:
  - **Do not interpret quotes**: All applicable separators will be considered as a column separator even when enclosed in quotes.
  - **Consider text in "double quotes" as a single column**: Values enclosed in double quotes will be considered as a single column. Any separator found within double quotes will not be considered as a column separator.
  - **Consider text in `single quotes` as a single column**: Values enclosed in single quotes will be considered as a single column. Any separator found within single quotes will not be considered as a column separator.
8. Click **Next** to continue.

9. Define **Status Value Map**.

Monitoring Studio will look for a match between the extracted value and the mapping rules status. The **Status** parameter of the **Value Map** Monitor instance will be set with the matching status. The mapping is applied from the highest alert level to the lowest (Failed, Suspicious, OK) to avoid conflicts if the same value is listed multiple times.

Enter the value(s) that you wish to map to a specific status:

- **These values will be mapped to '0' (OK)**: the retrieved value(s) will be mapped to a **Status** value of '0', indicating an **OK** status.
- **These values will be mapped to '1' (Suspicious)**: the retrieved value(s) will be mapped to a **Status** value of '1', indicating a **Suspicious** status.
- **These values will be mapped to '2' (Failed)**: the retrieved value(s) will be mapped to a **Status** value of '2', indicating a **Failed** status.
- **Any other value will be mapped to**: Select an option corresponding to the status you wish to apply to any other value.

⚠️ A maximum of 25 values can be listed in this wizard. They should be separated by semi-colon (;).
10. Click **Next** to continue.
11. (Optional) Provide additional information about the extracted value. This information will be reported by the *StatusInformation* parameter of the **Value Map** Monitor instance *(SEN_MS_VALUEMAP)*.

![Value Map — Status Information Map](image)

- **Value Map — Status Information Value Map**

  - (Optional) Click the **Status Information Value Map**:

![Value Map — Status Value Mapping](image)

- **Value**: Provide additional information that will be associated to each previously mapped status through the *Status Information* parameter. A maximum of 25 values will be displayed in this wizard.
- **Other values**: Provide the information that will be mapped to any extracted values that
cannot be mapped to provided values.

- Click **Accept** to save your changes.
- **Value to Retain**: Select which value should be used to set the Status parameter (according to the mapping previously defined) and build the graph in the PATROL Console if several values are found in the source file. The possible values are:
  - **the first value found**: Only the first value found will be kept and mapped.
  - **the last value found**: Only the last value found will be kept and mapped.
  - **the least critical value**: Monitoring Studio will compare all values found and will only map the least critical one.
  - **the most critical value (Default)**: Monitoring Studio will compare all values found and will only map the most critical one.

12. Click **Accept** to save your changes and then **Next** to continue.
13. **Configure the Monitor settings**.
14. Click **Finish**. The corresponding Value Map instance (Value Map: &lt;Display Name&gt;) is created in the PATROL Console. The collected parameters for **Value Map Monitor** are listed in the **SEN_MS_VALUEMAP** chapter.
Monitoring with the KM
Monitoring Studio KM for PATROL, makes the monitoring of any technology (application, server, device, etc.) for which there is no out-of-the-box monitoring solution very simple. Once Monitoring Studio is configured, the technology is displayed in your BMC framework. This technology is then monitored just like any other standard component of the system (hardware, OS, middleware, etc.).

**Monitoring Templates**

Monitoring Studio KM for PATROL provides pre-built templates that allow you to monitor your business-critical technologies. Templates deliver deep visibility and help you identify complex performance issues without having to install third-parties tools or invest in time-consuming setups and configurations.

**Available Templates**

Click a link to download a template and start monitoring immediately:

- Huawei V3 Storage Devices
- EMC Isilon Storage Devices
- PostgreSQL
- SharePoint
- EMC SMI-S Provider
- IronPort systems
- Microsoft Lync Server 2013
- MySQL Server 5.5
- Microsoft Lync Server 2010

**Monitoring Use Cases**

This section provides some examples of operations that can be performed with Monitoring Studio KM for PATROL such as:

- Executing a Basic Nagios Plugin
- XML LOG File Parsing
Executing a Basic Nagios Plugin

This use case explains how to download a standard Nagios Plugin from the Internet, import it to Monitoring Studio v.9.4.00 and execute it from your PATROL Classic console.

2. Download the plugin on your local machine and then transfer it to a PATROL Agent Server through the FTP client running on Monitoring Studio v.9.4.00 (Host = ICINGA1 in this example).
3. Provide the execution right to execute the plugin locally on the server with the `chmod 755` command: `chmod 755 check_x224`.

![Image of terminal output showing the execution of the plugin]

4. Execute the plugin, to verify that it is running properly: `./check_x224 -H 10.0.15.89 -p 3389 -w 5 -c 10`.

   ![Monitoring Studio v.9.4.000 will not be able to execute the plugin if it does not run successfully on the server where it is downloaded and deployed.

   Monitoring Studio v.9.4.000 will not be able to execute the plugin if it does not run successfully on the server where it is downloaded and deployed.
Once the plugin runs successfully, you can configure Monitoring Studio 9.4.00.
5. Create a **Group** on the ICINGA1 host.
6. Create a Nagios Plugin Monitor by right-clicking the **Host > KM Commands > New > Monitor...**

7. Select **Nagios Plugin** from the drop-down list and click **Next**.
8. Provide the path to the Nagios Plugin: /tmp in this example.

9. Provide the arguments used previously to test the plugin: `check_x224 -H 10.0.15.89 -p 3389 -w 5 -c 10`.
10. Click **Next** to continue.
11. Keep the default settings for the **Nagios Performance Data**.

12. Click **Next** to continue.
13. Keep the default settings for the **Missing Performance Data Detection**.

14. Click **Next** to continue.
15. Provide a **Display Name** for the Nagios Plugin Monitor: *Remote Desktop Check*, in this example.
16. Click **Finish**.
17. The Nagios Plugin Monitor is created and displayed in the PATROL Console.

Refer to the Executing Nagios Plugins chapter for detailed information about the Nagios Plugins Monitor settings and to the SEN_MS_NAGIOSPLUGIN chapter for a list and a description of the parameters collected by the Monitor.
Parsing an XML Log File

Most technologies use log files to trace their operations and notify operators when failures occur. These log files contain crucial information about the hardware, software, or system problems that may arise in your IT environment.

In this section, we will examine one of the many existing log file formats: the XML log file. The XML log file format makes the extraction of data way easier as this data is saved in a structured format: each log entry includes a date/time, the name of the logger, and many other useful elements.

Monitoring Studio can help you parse any XML Log file to monitor its properties and get notified every time a severe issue occurs.
1. **Specifying the file to monitor**

2. **Pre-processing the content of the file (converting XML to CSV)**

3. **Searching for strings in the result of the XML-to-CSV pre-processing**

**Specifying the file to monitor**

1. In the PATROL Console, right-click the **Host** or **Monitor Group** icon and select **KM Commands > New > Monitor...**

2. Select **File** from the drop-down list and click **Next**.

3. Specify that this file is a Log file, i.e. that Monitoring Studio needs to monitor only the new lines that are being added to the file.

4. Specify the path to the file. You can use wildcards (* and ?) if the name of the file changes over time (like a time-stamped Log file). In such case, Monitoring Studio monitors the most recently updated file which matches with the specified path.

5. Select the parameters you want to monitor (in our example: **Exists, Size, LastChanged, GrowthSpeed, and GrowthPercentage**). See **SEN_MS_FILE** for parameter details.

![New File for pub-ls95](image)

**Managed System:PC-CPWIN7_31B1**

**Parameters to Monitor**

Select the parameters you want to monitor for this log file:

- **Exists** File's presence
- **Size** File's size in KBytes
- **LastChanged** Time elapsed since the last modification
- **GrowthSpeed** File’s growth speed in KBytes per minute
- **GrowthPercentage** File’s growth percentage

![Selecting the Parameters to Monitor](image)

6. **Configure the Monitor settings** if you wish to change the label, the PATROL ID, or the thresholds for some parameters of the instance. In our example, we changed the label to **MySQL Server**.

You have successfully setup the monitoring of an XML log file. The corresponding “MySQL Server” log file icon has now been created and is displayed in the PATROL console.
Now to parse this file, you need to pre-process the XML text (in order to later be able to run String Searches or perform Numeric Value Extraction on the result) and thus be notified when failures occur.

### Pre-processing the content of the file (converting XML to CSV)

Since you are dealing with an XML output, the content needs to be processed to extract individual objects and their properties.

1. In the PATROL Console, right-click the **Log File: MySQL Server** icon and select **KM Commands > New > Text Pre-Processing...**
2. Select the **Convert XML to CSV (Comma-Separated Values)** option and click Next.
In this example, the records in this XML Log file are provided in the following format:

```
<rec>
  <vm>su37sr72</vm>
  <ts>2003-09-22 11:47:35.511 CEST</ts>
  <level>ERROR</level>
  <class></class>
  <method></method>
  <ctx>
    <pid>A141607</pid>
    <appid>frontnet</appid>
    <cname>User_3_0.getDefaultUserRole</cname>
    <reqid>2</reqid>
    <sesid>1uEPHTdRG2mM6GCfhv1EkwcBrCi68ffGlzIGEtGHWFM5Hc71wE7!-1625978434!-1455528670!</sesid>
    <thrid>ExecuteThread: '68' for queue: 'default'-f7c8b25c01</thrid>
    <cthid>ExecuteThread: '68' for queue: 'default'-f7c8b1696c</cthid>
  </ctx>
  <msg>
    <!CDATA[FNNotAuthorizedException;FEA002002;No authorization to execute service operation]]>
  </msg>
</exc>

<ts>2003-09-22 11:47:35.509 CEST</ts>
<sev>ERROR</sev>
<exc>
  <ts>2003-09-22 11:47:35.509 CEST</ts>
  <sev>ERROR</sev>
  <ctx>
    <pid>A141607</pid>
    <appid>frontnet</appid>
    <cname>User_3_0.getDefaultUserRole</cname>
    <reqid>2</reqid>
    <sesid>1uEPHTdRG2mM6GCfhv1EkwcBrCi68ffGlzIGEtGHWFM5Hc71wE7!-1625978434!-1455528670!</sesid>
    <thrid>ExecuteThread: '68' for queue: 'default'-f7c8b25c01</thrid>
    <cthid>ExecuteThread: '68' for queue: 'default'-f7c8b1696c</cthid>
  </ctx>
  <stack>
    <!CDATA[com.csg.pb.frontnet.exec_arch.calx.FNNotAuthorizedException: No authorization to execute service operation]]>
    
at com.csg.pb.frontnet.services.user_3_0.bean.UserBean_3_0.getDefaultUserRole(UserBean_3_0.java:345)
    
at com.csg.pb.frontnet.services.user_3_0.bean.UserBean_3_0_3c05dc_EOImpl.getDefaultUserRole(UserBean_3_0_3c05dc_EOImpl.java:145)
    
at com.csg.pb.frontnet.services.user_3_0.bean.UserBean_3_0_3c05dc_EOImpl_WLSkel.invoke(Unknown Source)
    
at weblogic.rmi.internal.BasicServerRef.invoke(BasicServerRef.java:360)
    
at weblogic.rmi.cluster.ReplicaAwareServerRef.invoke(ReplicaAwareServerRef.java:93)
    
at weblogic.rmi.internal.BasicServerRef.handleRequest(BasicServerRef.java:329)
    
at weblogic.rmi.internal.BasicServerRef.dispatch(BasicServerRef.java:178)
    
at weblogic.rmi.internal.ServerRequest.sendOneWayRaw(ServerRequest.java:92)
    
at weblogic.rmi.internal.ServerRequest.sendReceive(ServerRequest.java:112)
    
at weblogic.rmi.cluster.ReplicaAwareRemoteRef.invoke(ReplicaAwareRemoteRef.java:263)
    
at weblogic.rmi.cluster.ReplicaAwareRemoteRef.invoke(ReplicaAwareRemoteRef.java:230)
    
at weblogic.rmi.internal.ProxyStub.invoke(ProxyStub.java:35)
    
tProxy1401.getDefaultUserRole(Unknown Source)
    
at com.csg.pb.frontnet.services.user_3_0.bean.UserCA_3_0.getDefaultUserRole(UserCA_3_0.java:244)
    
at com.csg.pb.frontnet.apps.common.brokers.FnUserBroker.getDefaultPortalUserSettings(FnUserBroker.java:56)
```
The XML tag identifying each new record is `<REC>`. Let’s say that you would like to retrieve the `<TS>` value, the `<LEVEL>` value, the `<CNAME>` value under `<CTX>` and the `<MSG>` value as they are likely to provide additional information when a problem occurs.

Therefore, you need to specify that `REC` is the XML tag for a new record and that you want to include the value for the following properties and sub-tags:TS LEVEL CTX.CNAME MSG. Please
note the syntax "CTX.CNAME" which means the value of CNAME under the CTX tag.

Then, you need to specify a label and an ID for the text pre-processing object that will be created under the Log File icon (xml2Csv, for example).

As a result, a new icon representing the XML-to-CSV pre-processing (xml2Csv) is created and displayed in the Log File tree view of the PATROL Console:

Accessing the TransformResult File
This object has a single **TransformResult** text parameter as a result of the XML to CSV pre-processing:

```xml
2003-09-22 11:47:35.511 CEST;ERROR;User_3_0.getDefaultUserRole;
FNNotAuthorizedException;FEA002002;No authorization to execute service operation]]
>;  
2003-09-22 11:52:05.984 CEST;ERROR;User_3_0.getDefaultUserRole;
FNNotAuthorizedException;FEA002002;No authorization to execute service operation]]
>;  
2003-09-22 12:06:18.272 CEST;ERROR;User_3_0.getDefaultUserRole;
FNNotAuthorizedException;FEA002002;No authorization to execute service operation]]
>;  
2003-09-22 12:09:53.920 CEST;ERROR;User_3_0.getDefaultUserRole;
FNNotAuthorizedException;FEA002002;No authorization to execute service operation]]
>;  
2003-09-22 12:10:39.557 CEST;ERROR;KycBeneficialOwnerProfiles;
FNDBDataAccessFailureException;RDS001003;Code not Found - TableName: Landcode_1_RefTableObject, BusinessUnit: 0012, Language: 891, Code: 001]]
>;  
2003-09-22 12:10:39.566 CEST;ERROR;KycBeneficialOwnerProfiles;
FNDBDataAccessFailureException;RDS001003;Code not Found - TableName: Landcode_1_RefTableObject, BusinessUnit: 0012, Language: 891, Code: 001]]
>;  
2003-09-22 12:10:56.637 CEST;ERROR;CIFS_Customer_1.getCustomer;
FNDBDataAccessFailureException;RDS001002;Code not Found - TableName: Service_Status_InfoRefTableObject, BusinessUnit: 0000, Code: CIFS_Customer_1_0]]
>;  
2003-09-22 12:10:56.643 CEST;SEVERE;CIFS_Customer_1.getCustomer;
FNServiceNotAvailableException;FEA000001;Service not available - Service FNServiceState.getState]]
>;  
2003-09-22 12:10:56.945 CEST;ERROR;BPST_UserProfile_3.getUsers;
FNDBDataAccessFailureException;RDS001002;Code not Found - TableName: Service_Status_InfoRefTableObject, BusinessUnit: 0000, Code: BPST_UserProfile_3_0]]
>;  
2003-09-22 12:10:56.950 CEST;SEVERE;BPST_UserProfile_3.getUsers;
FNServiceNotAvailableException;FEA000001;Service not available - Service FNServiceState.getState]]
>;  
2003-09-22 12:21:30.004 CEST;ERROR;User_3_0.getDefaultUserRole;
FNNotAuthorizedException;FEA002002;No authorization to execute service operation]]
>;  
```

**Searching for strings in the result of the XML-to-CSV pre-processing**

From this transformed output, you can now look for strings corresponding to a severe alert level. To do so, you need to create a specific String Search on the 'SEVERE' string. Every time this word is found in your LOG file, Monitoring Studio will detect it and report it through the **MatchingLineCount** parameter of the String Search monitor, this will let you know that a severe issue occurred and needs to be addressed.

1. Right-click the **Text-Pre-Processing: xml2CSV** icon > **KM Commands > New > String Search**...
2. Search for lines that contain the string "SEVERE" in the second column, corresponding to XML records whose `<LEVEL>` is "SEVERE".
   - From the first pull-down list, select **contain** and type **SEVERE**
   - From the second pull-down list, select **in the following column number** and type **2** to search in the second column (purple color above)
Performing a String Search Command on a Converted File

3. Click **Next**
4. Set the automatic acknowledgment of alerts and click **Next**.
5. Uncheck all the separators and check the semicolon (character previously specified as separator). No other option needs to be modified in our example.
6. Click **Next**.
7. **Configure the Monitor settings**.
8. Click **Finish**

You can create as many String Searches as you want on a file and on a text pre-processing object, and you can create several different text-processing objects on the same file object.

As a result, you get the following String search object: SEVERE under the XML-to-CSV pre-processing object:

Accessing the String Search Results
Monitoring Studio is now configured to parse your XML log file in order to detect potential severe issues. Go further in the monitoring of your XML log file by configuring **Specific Alert Actions**.
Administrative Tasks
Acknowledging Alerts

After acknowledging an alert on a parameter, the parameter shows up in the PATROL Console as "offline" from the moment the alert is acknowledged and until the next data collection when the parameter status will be set again.

⚠️ Offline parameters do not trigger notifications.

Monitoring Studio KM for PATROL allows you to acknowledge alerts through the following KM Commands:

- Acknowledge Alerts
- Acknowledge all and Reset
- Reset CollectionErrorCount
Acknowledge Alerts

Alert acknowledgment is applicable to all Monitors except for String Search, SNMP Trap and Windows Event instances for which a specific command (Acknowledge all and Reset) can be used to acknowledge all alerts and reset the count parameters.

To acknowledge all alerts for a specific Monitor

1. Right-click the object icon > KM Commands > Acknowledge Alerts...

2. Monitoring Studio displays the name of the parameters for which an alert can be acknowledged. Click a parameter to select it and click OK. The selected parameter(s) will be deactivated and then reactivated. An acknowledging event will automatically be triggered.
Acknowledge all and Reset

Some technologies are unable to inform a 'back to normal situation' when performing the monitoring with String Searches, SNMP Traps, and Windows Events. You may at times want to manually acknowledge the alerts and reset the MatchingLineCount, MatchingTrapCount, or MatchingEventCount parameters to zero.

To manually acknowledge alerts, right-click the object icon > KM commands > Acknowledge all and Reset.
Reset CollectionErrorCount

The CollectionErrorcount parameter reports all data collection-related errors that occurred during the collection process. Because this parameter is cumulative, you might want to acknowledge it manually.

The Reset CollectionErrorCount menu command allows you to reset the CollectionErrorCount parameter of a Group or the Monitoring Studio instance to zero.

To perform the Reset CollectionErrorCount menu command, right-click the object icon > KM commands > Reset CollectionErrorCount.

⚠️ By default, the CollectionErrorCount parameter will be reset after 15 minutes (Group instance) or after 135 minutes (Monitoring Studio instance) if no new errors are found. These default settings can be respectively customized with the configuration variables /SENTRY/STUDIO/<groupID>/collectionErrorCountAutoAcknowledgeTime and /SENTRY/STUDIO/collectionErrorCountAutoAcknowledgeTime.
Configuring Java Settings

The Java Settings wizard enables you to define which Java instance you wish Monitoring Studio to use. You can either use the automatic detection, select a pre-detected java path or enter manually the path leading to the Java executable directory.

⚠ Monitoring Studio requires Java 1.8 or higher to perform most of its operations. As a result, the product will not start if a compatible Java Runtime Environment is not found (An error will be automatically reported by the CollectionErrorCount parameter of the Monitoring Studio instance).

By default, the product tries to locate a compatible Java Runtime Environment (JRE) on the system where the PATROL Agent runs, and uses it to run the Java Collection Hub, which is a core piece of the product, allowing to perform operations that wouldn’t normally be possible on a PATROL Agent (SSH, SSL, SNMPv3, etc.).

To access the Java Settings wizard, right-click the main Monitoring Studio icon > KM commands > KM Settings > Java Settings...

1. Use the Path to Java Runtime Environment (JRE) to specify the path of the JRE you want Monitoring Studio to use:
   - **Automatically detected at runtime**: Monitoring Studio will automatically search and use a compatible JRE at the initial discovery that occurs when the PATROL Agent and Monitoring Studio are started.
   - **List of detected JREs**: Monitoring Studio displays a list the JREs available on your system. Select the JRE you wish to use.
   - **Other**: Select this option if you wish to manually enter the Java executable directory path in the field provided below.
(Optional) If you have selected the **Other** option, you can select the **Do NOT verify the compatibility of the specified JRE** option, to prevent Monitoring Studio from verifying the compatibility of the specified JRE. Use this option only if you know that the provided JRE is compatible even if the compatibility check fails.

2. (Optional) Click the **More Information** button to display the status of the JRE compatibility check and the JRE path currently used by Monitoring Studio.

3. (Optional) Configure the **Advanced Settings**:

![Advanced Settings](Attached Image)

- **Credentials**
  - **Use the following credentials to launch the JVM process**:
    - **Username**: 
    - **Password**: 
  - **Note**: Leave empty to use the PATROL Agent's default account

- **Java Collection Hub Heap Memory**
  - **Minimum (MB):** 128 (default: 128 MB)
  - **Maximum (MB):** 512 (default: 512 MB)

- **Additional Java Collection Hub JVM Options**
  - **Command line options**: 

![Accept Cancel Buttons](Attached Image)
4. **Credentials**: Provide the specific credentials that Monitoring Studio will use to start the Java Virtual Machine (JVM). By default, the java process is launched with the same credentials as the PATROL Agent. You can provide specific settings if the PATROL Agent's default account does not have sufficient privileges to perform the operations required by the Java Collection Hub.

5. **Java Collection Hub Heap Memory**: Set the minimum and maximum size (in MB) of the Java Hub Heap Memory according to your environment requirements. (Default minimum Java heap size = 128 MB and default maximum Java heap size = 512 MB).

6. **Additional Java Collection Hub JVM Options**: Enter arbitrary arguments to the java -jar ... command line that Monitoring Studio uses to launch the Java Collection Hub in the **Command line options** field. This can be particularly useful for debugging.

7. Click **Accept** to save your settings. Monitoring Studio will automatically verify the relevance of the new settings and will warn you if a problem is detected. Changing the settings, requires for Monitoring Studio to stop the Java Collection Hub for about 10 seconds and then to restart the Java Collection Hub. In the meantime, operations that leverage the Java Collection Hub will fail and Monitoring Studio will report an error in the corresponding **Annotation** of the PATROL graph.

8. Click **Finish** to save your changes.
Configuring the SMTP Server

An SMTP server is required to receive alerts by email. If you wish to configure e-mail alert actions, you need to specify the SMTP server to be used.

To configure the SMTP server:

1. In the PATROL Console, right-click the Monitoring Studio icon and select KM Commands > KM Settings > SMTP Settings.

2. Enter a fully qualified SMTP server hostname and click OK.
Configuring the Proxy Settings

Monitoring Studio requires that you configure the settings of your proxy server, if you use one, to enable Web requests to servers located on the Internet. These settings can then be shared or bypassed when configuring the Web Request Monitor.

To configure the proxy server

1. In the PATROL Console, right-click the Monitoring Studio icon and select KM Commands > KM Settings > Proxy Settings.

2. Provide the following proxy server information:
   - **Proxy hostname**: Enter the hostname of the proxy server used to connect to web sites.
   - **Port number**: Enter the port to be used to access the proxy server. Default: 3128.

3. Enter the **Username** and **Password** to use for proxy authentication and click **OK**.
Configuring Thresholds

Alert thresholds are dynamically set by Monitoring Studio KM for PATROL. However, it is possible to change the threshold mechanism as well as the alert settings, as described in the following sub-sections:

- **Specifying the Threshold Mechanism Mode**
- **Setting or Modifying Alert Thresholds**
Specifying the Threshold Mechanism Mode

By default, Monitoring Studio KM for PATROL automatically determines which mechanism (Tuning or Event Management) is best suited to the managed system when it first runs. This threshold mechanism selection can however be modified later on through the Thresholds Mechanism Selection KM Command:

1. Right-click the main Monitoring Studio icon > KM Commands > KM Settings > Thresholds Mechanism Selection...

2. Select one of the following options:
   - **Automatic (Recommended)**: Monitoring Studio automatically detects the way thresholds are managed by the PATROL Agent and uses the most relevant mode (Tuning or Event Management).
   - **Tuning**: Monitoring Studio manages its thresholds through the standard internal PATROL mechanism (Override parameters). Thresholds are stored in the PATROL Agent configuration under the /__tuning__ tree.
   - **Event Management**: Monitoring Studio manages its thresholds through the Event Management mechanism. Thresholds are stored in the PATROL Agent configuration under the /AS tree. This option requires that you set up the PATROL for Event Management KM on your PATROL Agent. PATROL for Event Management has to be enabled and preloaded.

⚠️ In order to avoid side effects and unpredictable behavior, if you change the thresholds management option, Monitoring Studio will automatically migrate the thresholds set through the previous method to the new method (only for its monitored objects).
Setting or Modifying Alert Thresholds

Monitoring Studio enables you to set alert thresholds for each parameter through a command menu. To set or modify alert thresholds, right-click any Monitor icon > KM commands > Thresholds...

The Threshold Management wizard can also be prompted immediately following the setup of a new monitoring object. The last panel of the wizard offers you the option of either accepting default thresholds, modifying them right away, or setting them later.

A drop-down list displays the parameters available for the instance.

An asterisk symbol "*" displayed next to the parameter name signifies that the parameter already has thresholds. You can then choose to either keep the same thresholds or just view them or modify them... as per the requirement.

The Reference Guide lists all the classes with details on the menu commands and the parameters available for each class.

Threshold Type and Alert actions

Monitoring Studio has two "types" of thresholds: Simple (2 alerts maximum) and Advanced (3 alerts maximum). Depending on the user’s selection, both types of thresholds can raise the following alerts:
- Do not trigger any alert
- Trigger an INFORMATION
- Trigger a WARNING
- Trigger an ALARM

In addition, you can set Alert Actions to be triggered when an alert is raised on a parameter. Please see the section: **Alert Actions capabilities** for more details on the same.

**Simple Thresholds**

This type of threshold is easy and very user-friendly: the threshold values are expressed in terms of "greater than/lower than". All that you need to do is to enter actual "limits" as per the specific requirement, and the frequency of occurrence. This type of threshold has a maximum of two alert types.

Certain parameters have "pre-defined" values. Basically these parameters are the ones with boolean values, so the choice you can make can only be one or the other – depending on the parameter itself.
Advanced thresholds

**Threshold Manager Wizard – Parameter Selection**

Select a parameter and check the **Use advanced thresholds management** box to access the advanced threshold options. Click **Next**.

The advanced thresholds mode allows you to set up to a maximum of 3 alerts.
**Threshold Manager Wizard – Thresholds Setting**

Managed System: PC-IGVINZ_3481

Set thresholds for "GrowthPercentage"

**Border Alert:**
- If value is NOT between:
- Do not trigger any alert
- Value between: $x$ and $y$
- Occurs: $t$ times in a row.

**Alarm 1 Alert:**
- If value is between:
- Do not trigger any alert
- Value between: $x$ and $y$
- Occurs: $t$ times in a row.

**Alarm 2 Alert:**
- If value is between:
- Do not trigger any alert
- Value between: $x$ and $y$
- Occurs: $t$ times in a row.

The unit for this parameter is: Percentage per minute (%/min)

This type of threshold setting is a little more complex as it requires the user to enter specific ranges of values as thresholds. This setting is like that of PATROL thresholds, where you have the possibility of raising three alert types:

- **Border Alert:** An alert will be triggered if the value polled is not between "x" and "x" and occurs "x" times in a row.
- **Alert 1:** An alert will be triggered if the value is between "x" and "x" and occurs "x" times in a row.
- **Alert 2:** An alert will be triggered if the value is between "x" and "x" and occurs "x" times in a row.

**Note applicable to both Simple and Advanced Thresholds:** In general, Alert Actions are triggered only when a parameter breaches its thresholds. However, for certain parameters, such as MatchingTrapCount, MatchingLineCount, and MatchingEventCount, it is possible to trigger Alert Actions not just when thresholds are breached, but also each time an SNMP trap is received or a matching line/event is found. The option Execute the Alert Actions appears in the thresholds wizard for the parameters to which this feature is applicable under the section for Alarms.

**Example**

Let's say you are setting thresholds on a String Search. In this wizard you will find the option Execute the Alert Action as String Search has the parameter MatchingLineCount. If you select Execute the Alert Actions every time a matching line is found on the MatchingLineCount parameter, Monitoring Studio will run the Alert Actions every time a line matching the String Search is found.
Copy, Cut, and Paste

Once you have configured one or several monitoring tools, you may need to copy or move them under a new container. You may for example need to duplicate a String Search that you have setup to one Command Line execution to another one. This can be done easily with the Copy, Cut, and Paste feature of Monitoring Studio.

To access the Copy, Cut, and Paste features, right-click the object icon > KM commands > and select either Copy, Cut, or Paste, depending on the action that you wish to perform.

⚠️ Some objects cannot be pasted<other objects. For example, you cannot paste a String Search object under a Process monitoring object. Basically, you can paste everything everywhere that you would have been able to do with the regular wizards.

⚠️ The Paste menu command also pastes the children of the copied/cut object.
Deploying your Configuration

Monitoring Studio allows you to backup your entire Monitoring Studio configuration or export a single Group configuration to then deploy it on all the servers where the technology to be monitored is installed. The deployment consists in:

1. **Backing up an entire Monitoring Studio configuration** or **exporting a single Group configuration**
2. **Importing a configuration on the relevant machines**
Backing Up an Entire Monitoring Studio Configuration

The **Backup Configuration** feature provides a simple way to save the entire Monitoring Studio configuration by backing it up to a file for later reuse. The configuration saved in a configuration file (.cfg) can then be loaded onto a different PATROL Agent through the PATROL interface. The main benefits are:

- It eliminates the trouble of manually configuring Monitoring Studio on another PATROL Agent all over again
- It ensures both configurations are identical

**To back up the entire Monitoring Studio configuration**

1. Right-click the main **Monitoring Studio** icon > **KM commands** > **Configuration** > **Backup Configuration**.
2. (Optional) Check **Include Monitoring Studio global settings** if you want the global settings to be included in the .cfg file.

3. Click **Next**.

4. Select the global variables to be included. Please note that global variables will only be displayed if they have previously been configured.

5. Click **Next**.
6. Enter the path and name of the file on the PATROL Agent system to backup the configuration. By default, the file will be saved on the PATROL Agent. It will be written to SEN_MS_<hostname>-<port>.cfg under %PATROL_HOME%\config on Windows and $PATROL_HOME/config on UNIX systems.

7. (Optional) **Overwrite the existing file:** Select this option if the file already exists and you wish to replace it.

8. Click **Next** and wait while Monitoring Studio backs up the configuration.

9. A confirmation message is displayed once the configuration has been successfully backed up. Click **Finish**.

### Exporting a Group Configuration

The **Export Configuration** feature provides a simple way to save the information relevant to a single Group. The Group configuration saved in a configuration file (.cfg) can then be loaded onto a different PATROL Agent through the PATROL interface. The main benefits are:

- It eliminates the trouble of manually configuring your Monitoring Studio Group on another PATROL Agent all over again
- It ensures both configurations are identical

When exporting your Group configuration, several options are offered (providing unique host descriptions, clearing host information, or clearing credentials) to tailor it to the monitored system on which it will be imported. The Group configuration saved in a configuration file (.cfg) can then simply be **imported and loaded onto a different PATROL Agent** through the PATROL
To export a Group configuration

1. Right-click a **Group** icon > **KM commands** > **Export Configuration...**

   ![Export Configuration Interface](image)

   **Exporting a Group Configuration - Clear Host Information and Credentials**

   2. (Optional) **Clear host information**: Select this option if you do not want the exported Group configuration to contain host information (hostname and display name). You will then have the possibility to enter the host information when importing the configuration on another PATROL Agent. You can click the **Hosts** button to view the complete list of hosts to be cleared:

   ![Hosts Dialog](image)

   **Exporting a Group Configuration - Clear Host Information**
3. **(Optional) Clear credentials:** Select this option if you do not want the exported Group configuration to contain credentials (username and password). You will then have the possibility to enter the credentials when importing the configuration on another PATROL Agent. You can click the **Credentials** button to view the complete list of credentials to be cleared:

![Credentials](image)

Exporting a Group Configuration - Clear Credentials

4. Click **Next**.
5. (Optional) If you have previously chosen to clear the host information, provide a unique description to easily identify each host during the import process and click **Next**.

6. (Optional) Select the Group Constants that you wish to clear in the exported configuration file and click **Next**. Note that the Group Constants will remain in the exported file, only their values will be cleared.
7. Enter the path and name of the file that will contain the Group configuration. By default, the file will be saved on the PATROL Agent. It will be written to SEN_MS_<hostname>-<port>.cfg under %PATROL_HOME%\config on Windows and $PATROL_HOME/config on UNIX systems.

8. (Optional) **Overwrite the existing file**: Select this option if the file already exists and you wish to replace it.

9. Click **Next** and wait while Monitoring Studio exports the Group configuration.

10. A confirmation message is displayed once the Group configuration has been successfully exported. Click **Finish**.
Importing a Configuration

This feature allows you to configure Monitoring Studio by importing a configuration already in use on another PATROL Agent:

- It eliminates the trouble of manually configuring Monitoring Studio on another PATROL Agent all over again
- It ensures both configurations are identical

Basically, you can import from another PATROL Agent:

- an entire Monitoring Studio configuration that has been backed-up which contains all the Groups, Hosts, and Monitors created under the Monitoring Studio icon (💡) or,
- a single Group configuration that has been exported which contains all the Hosts and Monitors created under the chosen Group (💡).

All information found in the configuration file (.cfg) will be imported and merged into the current installation.

⚠️ You may import a PATROL Agent configuration (".cfg") with any type of thresholds. If necessary, Monitoring Studio will convert the thresholds to the type of thresholds currently used at the next discovery (within an hour).

To import an existing configuration

1. Right-click the Monitoring Studio icon > KM commands > Configuration > Import Configuration.
2. Enter the path to the folder containing the configuration file(s) and click **Next**. By default, all .cfg files found under `%PATROL_HOME%\config` on Windows and `$PATROL_HOME/config` on UNIX systems will be listed.
3. Select the configuration file to import. It can either be an entire Monitoring Studio configuration or a single Group configuration.
4. Click **Next** and wait while Monitoring Studio processes your configuration file.

5. *(Optional)* When importing an entire Monitoring Studio configuration, select which global variables you wish to import and click **Next**. Please note that global variables will only be displayed if they have previously been exported.

⚠️ **Importing global settings will overwrite the current global settings of your existing objects.**
6. Provide a value for each Group Constant to be imported and click **Next**.

- **Warning**: If a Group Constant is marked as required, its corresponding value cannot be left blank.

- **Tip**: Updating the Group Constant(s) remains possible after the import process by right-clicking a Group > KM commands > Group Constants...

![Importing Configuration - Group Constants Import](managed_system_toland_3181)

This Monitoring Studio template has 8 host(s) that require host specific information. The following steps will ask for their details.

![Importing Configuration - Host Information](hosts)
7. (Optional) If you have cleared the host information during the export process, Monitoring Studio indicates the number of hosts found in the configuration file that require specific information before being imported. You can click the Hosts button to view the complete list.
8. Click Next.

9. Provide specific information for each imported host (a counter indicates the number of hosts to identify):
   - **Hostname/IP address/FQDN**: Enter the Hostname, IP address or Fully Qualified Domain Name of the host on which the technology you wish to monitor is running.
   - **(Optional) Display Name**: Enter the name that will be displayed in the console for this host.
   - **(Optional) Description**: Enter a unique description or edit the existing description if you have already provided one during the export process.
   - **Create a separate device in TrueSight**: Check this box if you want the remote monitored host to appear as a separate device in BMC TrueSight Operations Management.
10. Click **Next**.

11. (Optional) If you have chosen to clear credentials during the export process, provide system credentials and SNMP information for each imported host. A counter indicates the number of system credentials to provide:
   - **Monitors using these credentials**: Click this button to display the list of Monitors using system credentials.
   - Enter the **Username** and **Password** to connect to the targeted host. These credentials will be used for all system related monitoring tasks to gather data.
   - **(Optional - for system credentials only) SNMP version used by <Managed System>**: Select the SNMP version to be used when monitoring via SNMP and/or when performing an SNMP availability check. If no SNMP agent is running on the host, select None.

12. Click **Next**.

*The following steps which consist in configuring SNMP settings (v1, v2c, or v3) are optional and only available when an SNMP version is selected.*
13. If you have previously selected SNMP version 1, indicate:
   - the **Community** to be used.
   - the **Port** number (default: 161).

13. If you have previously selected SNMP version 2c, indicate:
   - the **Community** to be used.
- the **Port** number (default: 161).
- the number of seconds Monitoring Studio will wait for an SNMP response (Default: 120 seconds). This timeout covers all the sub-queries that compose the 'parent' query.

13. If you have previously selected SNMP version 3, indicate:
   - The **Username** to be used to perform the SNMP query.
   - The **Authentication protocol** to be used to authenticate the SNMP v3 messages. Possible values are: *None*, MD5, SHA.
   - The **Authentication password** to be used to authenticate the SNMP v3 messages.
   - The **Privacy protocol** to be used to encrypt SNMP v3 messages. Possible values are: *None*, AES, DES.
   - The **Privacy password** associated with the privacy protocol.
   - The **Context name** accessible to the SNMP entity.
   - The **Port** number (default: 161).
   - The number of seconds Monitoring Studio will wait for an SNMP response (Default timeout: 120 seconds).
14. Click **Next**.

![Import Configuration](image)

**Overwrite/Rename Existing Objects**

Some objects defined in `SEM_MS_HyGroup.cfg` already exist.

- **Overwrite**
- **Rename by adding this prefix:**

15. This step is only displayed when some objects (Groups, Hosts, or Monitors) defined in the imported configuration already exist in the PATROL Agent configuration. In this case, you can view the complete list of conflicting objects by clicking the **Conflicting Objects** button. You then need to select one of these two options:
  - **Overwrite**: if you want to overwrite existing objects.
  - **Rename by adding this prefix**: if you do not want to overwrite existing objects. You can enter a prefix to be assigned to these copied objects. The prefix will be applied to the ID of all the Groups, Hosts and Monitors found in the configuration. Example: you enter "NEW_" in the prefix box and you have a Group called "Lync". The new ID will be "NEW_Lync" thus avoiding overwriting the existing "Lync" configuration.
16. Click **Next**.

17. Before importing the configuration, the number of objects, groups, and hosts to be imported are listed. You can choose to **pause the monitoring of the imported objects** to have time to review them. This will allow you to make sure the configuration is correct before starting the monitoring of the imported objects.

   ![Import Summary](image_url)

   **Note that you can resume the monitoring of the imported objects at anytime by right-clicking on the instances > KM Commands > Resume Monitoring.**
18. Click **Next** to start the import process.

![Import Configuration](image)

**Importing Configuration - Completed Import Confirmation**

505 configuration variables have been imported to the PATROL Agent's configuration.

Now performing a full discovery...

19. A message confirming the import and listing the number of imported configuration variables appears. Click **Finish**.

Once the discovery process is completed, all monitors related to the imported configuration will be displayed on your console.
Deleting a Monitored Object

%sPRODUCT SHORT NAME% allows you to delete:

- any monitored object by right-clicking Object icon > KM commands > Delete. This object and all its children will be removed and no longer monitored.
- all objects and children under a Group, Host, or Monitor Group instance by right-clicking it and selecting KM commands > Delete All.
Editing a Monitored Object

To edit a Host, Group or Monitor:

1. Right-click the instance of the monitored object > **KM Commands** > **Edit**...
2. Modify the information related to the selected object, just as you wish. Click **Next**.

3. The final step of the wizard displays the **Monitor Settings** of the monitored object. The wizard:
   - Displays a reminder of the object's **Internal ID**.
   - Lets you edit the object's **Display name**.

   *Please note that you can also edit the Display Name using the Rename KM command.*

4. Click **Finish** to validate your changes.
Forcing the Classic Mode

Monitoring Studio KM for PATROL can either be configured from BMC PATROL Console (Classic Mode) or BMC TrueSight Operations Management. When the KM is installed on a PATROL Agent, which is managed by Central Monitoring Administration (CMA), all the KM configuration menus are disabled in the PATROL Console. These configuration menus will display an error when selected, asking to use CMA instead. To configure Monitoring Studio from a PATROL Console, you need to force the KM to run in Classic Mode.

This will disconnect Monitoring Studio from receiving configuration from BMC ProactiveNet Central Monitoring Administration. Any monitoring that was configured in CMA and used by the PATROL Agent will be removed and replaced by Monitors created from the PATROL Console. However, this will not delete the policies created in CMA from the configuration.

⚠️ Any configuration for Monitoring Studio received from BMC ProactiveNet Central Monitoring Administration will be ignored.

To force the KM to run in Classic Mode:

1. In the PATROL Console, right-click the Monitoring Studio icon > KM commands > KM Settings > Force Classic Configuration Mode...
2. Check **Force the KM to run in Classic mode** and click **OK**.

Monitoring Studio will then start running in Classic Mode, enabling you to use the KM Configuration menus.
Modifying a Monitor Display Name and/or ID

Once you have configured one or several Monitors, you may need to rename/modify their display names or IDs. The Rename feature allows you to:

- Enter a new Monitor display name
- Modify the Monitor internal identifier (ID)
To modify the Monitor display name
1. Right-click the object icon > KM commands > Rename...
2. In the Monitor display name text field, enter a new display name.
3. Click OK to validate your changes.

To modify the Monitor internal identifier (ID)
1. Right-click the object icon > KM commands > Rename...
2. In the Monitor internal identifier (ID) text field, enter a new ID. If the Monitors attach...
Modifying a Group Display Name and/or ID

Once you have added Groups, you may need to rename/modify their display names or IDs. The Rename feature allows you to:

- Enter a new Group display name
- Modify the Group internal identifier (ID)
To modify the Group display name
1. Right-click the object icon > KM commands > Rename...
2. In the Group display name text field, enter a new display name.
3. Click OK to validate your changes.

To modify the Group internal identifier (ID)
1. Right-click the object icon > KM commands > Rename...
2. In the Group internal identifier (ID) text field, enter a new short ID.
3. Click OK to validate your changes. The following pop-up appears:

   ![Modification of a Group Name and/or ID - Warning]

4. If you are certain that the new ID will not cause any problem, click Yes.
   The PATROL Console will display the new label and will update the ID of the Group as well as all its dependent items.
Modifying Group Constants

Group Constants are defined at the Group level and can be reused in the various objects underneath. They facilitate the monitoring across various systems of a technology whose properties may change from one system to another. If you have previously specified Group constants when creating a Group, you will be able to modify them later on.

To modify Group Constants
1. Right-click a Group icon and select KM Commands > Group Constants...

2. For each of the predefined Group Constants, enter a new value.
3. Click OK to validate your changes.

The new Group Constants will now be taken into consideration by Monitoring Studio.

*Modifying Group Constants is also possible by right-clicking a Group > KM Commands > Edit.*
Temporarily Suspend the monitoring of an Object

In order to perform certain tasks, such as maintenance for example, it is often useful to pause the monitoring of an object or a group of objects. To pause the monitoring of a specific object or group of objects, right-click the object (or object container) icon > KM Commands > Pause Monitoring.

When in paused state, the object/group of objects is displayed with an OFFLINE status in the PATROL Console. Additionally, the mention '(Suspended)' may appear beside each related parameter.
To resume the monitoring of the object/group of objects, click Resume Monitoring in the menu.
Refreshing Parameters

To manually refresh parameters, without waiting for an automatic polling cycle, right-click the object icon > KM Commands > Refresh Parameters. This will reset the time to wait for the next polling interval. This will not affect "Once a day" or "Once a week" type collects. They will be collected at the scheduled day and time.
Specifying Alert Actions

Alert Actions enable you to choose/configure the execution of certain actions when a monitored technology goes into a state of alert. With the Alert Actions functionality, it is possible to do one of the following:

- Customize notifications
- Specify recovery actions

Global vs Specific Alert Actions

Alert Actions can be set at two different levels:

- **Global Alert Actions** are set at the Group level for the overall technology. They specify the way in which the notification of a problem is done for all instances under this Group. This means that any parameter within this Group that triggers an alert will perform the Alert Actions set at the global level, in addition to any specific alert action that might have been set.

- **Specific Alert Actions** are set at the Host level and Monitor level for individual parameters and generally indicate the actions to perform in order to recover from a particular problem (e.g.: when this parameter goes into alarm - execute this recovery script). Therefore, specific Alert Actions need to be set for each parameter as the need may be.

If a specific Alert Action is set on a parameter belonging to a Group already set to trigger Global Alert Actions, both settings will be taken into account for that particular parameter instance.
Set Global Alert Actions

1. In the PATROL Console, right-click the **Group icon** and select **KM Commands > Alert Actions...**

2. Select the **Alert Action(s)** that you want to trigger each time the parameters enter a state of alert and click **Next**.

   - **Trigger an event**
   - **Annotate the graph**
   - **Execute a command locally**
   - **Execute a PSL statement (advanced)**
   - **Display a pop-up on the consoles**
   - **Log to a file**
   - **Send a basic SNMP trap**
   - **Send a custom SNMP trap (advanced)**
   - **Send an e-mail**

3. **Enter the text for the event:**

   ```
   ${SEN_ALERT_DEFAULTCONTENT}
   ```

   **Note:** You can use macro variables in the fields above. See the documentation for details about supported macros.
3. Configure the selected Alert Action and click Next.

⚠️ The %SEN_ALERT_DEFAULTCONTENT alert macro is the default alert content suitable for events and annotations. It resolves to the following macro syntax: %SEN_GROUP_NAME %SEN_HOST_NAME %SEN_PARAMETER_STATUS on %SEN_PARAMETER_NAME of %SEN_OBJECT_LABEL.

⚠️ Please note that macros are case sensitive and should then always be written in uppercase.

4. Global Alert Actions have been set. Click Finish.

⚠️ You can set additional Alert Actions on any/all of the monitored parameters by using the "Alert Actions" menu from the monitor.
Set Specific Alert Actions

1. In the PATROL Console, right-click any Host or Monitor instance and select **KM Commands > Alert Actions**...
2. Select the parameter for which you wish to set specific Alert Actions and click Next.

⚠ You can see the list of previously set Global Alert Actions by clicking the Group Alert Actions button.

3. The panel lists all available Alert Actions. Make your selection and click Next.
4. Configure the selected **Alert Action** and click **Next**.

![Specifying Specific Alert Actions - Alert Execution Confirmation Page](image)

4. Click **Finish**. When the selected parameter goes into alert, it will trigger the specific Alert Action as configured.

⚠️ You can set Global Alert Actions on all parameters by using the "Alert Actions" menu from the Group icon.
Alert Actions Capabilities

The Alert Actions feature offers a large choice of actions in order to notify you of a problem with the technology or to recover from a particular problem. Monitoring Studio can be configured to run one or several types of actions when an alert is triggered by PATROL. You can use Alert Action macros to construct Alert Actions.

⚠️ Please note that macros are case sensitive and should then always be written in uppercase.

The following types of Alert Actions can be performed by Monitoring Studio:

**Trigger an event**

If you select the **Trigger an Event** action, you need to enter the string that will be displayed with the event.

A PATROL Event can be viewed from:

- Standard PATROL Consoles (Classic Console, PATROL Central)
- PATROL Enterprise Manager
- BMC Impact Manager
- BMC ProactiveNet Performance Management/BMC TrueSight Operations Management
- Other third-party products that interface with PATROL.
Annotate the graph

If you select the **Annotate the graph** action, you need to enter the string that will be displayed within the annotation point.

To view an annotation, when a threshold is breached, open the graph associated to the parameter in alert and double-click on the black annotation mark on top of the first data point.
Execute a command

At the Group Level

If you select the **Execute a command** action, you need to complete the following panel:

- Enter a command line to be executed on the PATROL Agent node.
- (Optional) Enter valid credentials to run the command on the PATROL Agent node. Leave blank to use the PATROL Agent's default account.

The command line can be a program utility or a script call with arguments.

⚠️ The command must be non-interactive (no window, no user input)

At the Host or Monitor Level

If you select the **Execute a command** action, you need to complete the following panel:
Alert Actions: Execute a Command (Host or Monitor Level)

- **Credentials**: Select from the drop-down list the type of credentials that you want to use to execute a command:
  - *Agent's Default Account*: Uses the PATROL Agent credentials for localhost monitoring.
  - *System Credentials (default)*: Uses the credentials previously specified when creating the Host.
  - *Add new credentials...*: Let’s you set new credentials for this specific monitoring.
- Enter a command line to be executed on the target host.

Note: You can use macro variables in the field(s) above. See the documentation for details about supported macros.
Execute a PSL statement

If you select the **Execute a PSL statement (advanced)** action, you need to enter the PSL statement to be executed by the PATROL Agent on the localhost. Although only a single line is permitted, it can have several PSL instructions.

⚠️ The PSL command is recommended for advanced users of PATROL.
Display a pop-up on the consoles

If you select the **Display a pop-up on the consoles** action, you need to enter the message that will be displayed in the pop-up as well as in the title of the pop-up window.

![Alert Actions dialog](image)

**Console Pop-up Configuration**

- **Enter the pop-up window title:**
  
  ```
  ${SEN_PARAMETER_STATUS}: ${SEN_GROUP_LABEL}
  ```

- **Enter the message for the pop-up displayed on the consoles:**
  
  ```
  ${SEN_ALERT_DEFAULTCONTENT}
  ```

**Note:** You can use macro variables in the field(s) above. See the documentation for details about supported macros.
Log to a file

If you select the **Log to a file** action, you need to enter the log file path on the localhost and the content of the line.

![Image of Alert Actions configuration](image)

Enter the log file path:

```plaintext
C:\PROGRAMS\SOF\Patrol\log\%{SEN_GROUP_ID}.log
```

Enter the line to be written to the log file:

```plaintext`
%{SEN_ALERT_DEFAULTLOGMESSAGE}
```

Note: You can use macro variables in the field(s) above. See the documentation for details about supported macros.
Send a basic SNMP trap

If you select the **Send a basic SNMP trap** action, you need to complete the following panel:

- IP address or hostname of the SNMP trap destination (**Host** field)
- SNMP port and community string
- Text that will be sent with the SNMP trap
Send a custom SNMP trap

If you select the **Send a custom SNMP trap (advanced)** action, you need to complete the following panel:

- IP address or hostname of the SNMP trap destination (**Host** field)
- SNMP port and community string
- All the characteristics of the trap: Enterprise OID, trap number and up to 4 varbinds.
Send an e-mail

An SMTP server is required to receive alerts by email and should be configured first by right-clicking the Monitoring Studio icon > KM commands > KM Settings > SMTP Settings.

If you select the **Send an e-mail** action, you need to complete the following panel:

- Enter the sender and the recipient email address in the **From** and **To** fields. To send the email to multiple recipients, use the comma (,) or the semi-column (;) to separate the recipients' email addresses (ex: RecipientAddress1;RecipientAddress2;RecipientAddress2 OR RecipientAddress1,RecipientAddress2,RecipientAddress2)
- With the help of **Alert Action macros**, type the email subject and the message you wish to send.
### Alert Action Macros

A macro is a variable whose value is replaced when an Alert Action is triggered. Macros can be used to customize the content of each Alert Action. For example: `%{VALUE}` is replaced by the actual current value of the parameter that triggered the alert.

Each macro listed in the tables below contains information about what triggered the PATROL alert. Some macros are "general" or "common" - these can be used for any object, and some are "object-specific" macros that are specific to the object, such as databases or files, etc.

- **Macro syntax supports white spaces.**
- **Macros are case sensitive and should then always be written in uppercase.**

### General Macros

The macros given in the table below can be used on any object:

<table>
<thead>
<tr>
<th>General Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>% {SEN_ALERT_DEFAULTCONTENT}</td>
<td>Default alert content suitable for events and annotations, which resolves to the following macro syntax: % {SEN_GROUP_NAME} : % {SEN_HOST_NAME} %{SEN_PARAMETER_STATUS} on % {SEN_PARAMETER_NAME} of % {SEN_OBJECT_LABEL}; or % {SEN_GROUP_NAME} : %{SEN_PARAMETER_STATUS} on % {SEN_PARAMETER_NAME} of % {SEN_OBJECT_LABEL} (When the macros is used at Group level).</td>
</tr>
<tr>
<td>% {SEN_ALERT_DEFAULTLOGMESSAGE}</td>
<td>Default alert content suitable for log files, which resolves to the following macro syntax: % {SEN_TIME} : % {SEN_GROUP_ID} %{SEN_PARAMETER_STATUS} on % {SEN_PARAMETER_NAME} of % {SEN_OBJECT_LABEL}.</td>
</tr>
<tr>
<td>% {SEN_CREDENTIALS}</td>
<td>System Credentials.</td>
</tr>
<tr>
<td>% {SEN_HOST_DOMAIN}</td>
<td>Domain of the targeted host.</td>
</tr>
<tr>
<td>% {SEN_HOST_FQDN}</td>
<td>Fully qualified domain name of the targeted host.</td>
</tr>
<tr>
<td>% {SEN_HOST_IPADDRESS}</td>
<td>IP address of the targeted host.</td>
</tr>
<tr>
<td>% {SEN_HOST_NAME}</td>
<td>Name of the targeted host.</td>
</tr>
<tr>
<td>% {SEN_HOST_SNMP_COMMUNITY}</td>
<td>SNMP community set for the SNMP Agent on the targeted host.</td>
</tr>
<tr>
<td>% {SEN_HOST_SYSTEMTYPE}</td>
<td>Operating system type of the targeted host.</td>
</tr>
<tr>
<td>% {SEN_HOSTNAME}</td>
<td>Name of the monitored host as provided in the GUI.</td>
</tr>
<tr>
<td>% {SEN_INFORMATION}</td>
<td>Provides additional information about the problem/alert.</td>
</tr>
<tr>
<td>% {SEN_INFORMATIONONELINE}</td>
<td>Provides additional information about the problem/alert in a single line.</td>
</tr>
</tbody>
</table>
### General Macros

<table>
<thead>
<tr>
<th>Macro Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_NEWLINE}</td>
<td>Inserts carriage return.</td>
</tr>
<tr>
<td>%{SEN_OBJECT_CLASS}</td>
<td>Class name of the object to which the alert action belongs.</td>
</tr>
<tr>
<td>%{SEN_OBJECT_ID}</td>
<td>PATROL ID of the object triggering the alert.</td>
</tr>
<tr>
<td>%{SEN_OBJECT_LABEL}</td>
<td>Display name of the object triggering the alert.</td>
</tr>
<tr>
<td>%{SEN_OBJECT_TYPE}</td>
<td>Type of the object triggering the alert (&quot;Process&quot;, &quot;String&quot;, etc.).</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_ALARM1MAX)</td>
<td>Alarm1 maximum range of the parameter triggering the alert.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_ALARM1MIN)</td>
<td>Alarm1 minimum range of the parameter triggering the alert.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_ALARM1NTIMES)</td>
<td>Number of consecutive times the parameter triggering the alert must have a value within the alarm1 range before the alert occurs.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_ALARM1TYPE)</td>
<td>Alarm alert type of the parameter triggering the alert (OK, WARN, ALARM).</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_ALARM2MAX)</td>
<td>Alarm2 maximum range of the parameter triggering the alert.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_ALARM2MIN)</td>
<td>Alarm2 minimum range of the parameter triggering the alert.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_ALARM2NTIMES)</td>
<td>Number of consecutive times the parameter triggering the alert must have a value within the alarm2 range before the alert occurs.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_ALARM2TYPE)</td>
<td>Alarm2 alert type of the parameter triggering the alert (OK, WARN, ALARM).</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_BORDERMAX)</td>
<td>Border maximum range of the parameter triggering the alert.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_BORDERMIN)</td>
<td>Border minimum range of the parameter triggering the alert.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_BORDERNTIMES)</td>
<td>Number of consecutive times the parameter triggering the alert must have a value outside the border range before the alert occurs.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_BORDERTYPE)</td>
<td>Border alert type of the parameter triggering the alert (OK, WARN, ALARM).</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_NAME)</td>
<td>Name of the parameter triggering the alert.</td>
</tr>
<tr>
<td>% (SEN_PARAMETER_STATUS)</td>
<td>Status of the parameter.</td>
</tr>
<tr>
<td>General Macros</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>%{SEN_PARAMETER_VALUE}</td>
<td>Value of the parameter triggering the alert.</td>
</tr>
<tr>
<td>%{SEN_PARENT_&lt;macro-name-without-SEN_&gt;}</td>
<td>Gets the parent object’s macro. The name of the parent's macro is copied in the syntax without the SEN_. Example: In the case of String Search performed on a Command Line (parent) output, to read the full output from the alert action in LastMatchingLines, we need to read parent's %{SEN_RESULT}, which can be accessed using %{SEN_PARENT_RESULT}.</td>
</tr>
<tr>
<td>%{SEN_PARENT_CLASS}</td>
<td>Class name of the parent object to which the alert action belongs.</td>
</tr>
<tr>
<td>%{SEN_PARENT_ID}</td>
<td>PATROL identifier of the object’s parent.</td>
</tr>
<tr>
<td>%{SEN_PARENT_LABEL}</td>
<td>Display name of the object’s parent.</td>
</tr>
<tr>
<td>%{SEN_PARENT_TYPE}</td>
<td>Type of the object’s parent. (&quot;File&quot;, &quot;CommandLine&quot;, etc.).</td>
</tr>
<tr>
<td>%{SEN_PASSWORD}</td>
<td>Encrypted password of the targeted host.</td>
</tr>
<tr>
<td>%{SEN_RESULT}</td>
<td>Query result received for the monitored object during data collection, when available.</td>
</tr>
<tr>
<td>% (SEN_STATUSINFORMATION)</td>
<td>Provides additional information about the Status, as reported by the StatusInformation parameter (where available).</td>
</tr>
<tr>
<td>%{SEN_TIME:&lt;date-time-format&gt;}</td>
<td>Time of the alert action with a configurable time format as described in the Format Symbols chapter. Example: %{SEN_TIME:%H%M%S} may read 094517 at run time.</td>
</tr>
<tr>
<td>%{SEN_USERNAME}</td>
<td>Username to use to connect to the targeted host.</td>
</tr>
<tr>
<td>%{/...}</td>
<td><strong>FOR ADVANCED USERS ONLY</strong> Provides an internal instance variable name to be inserted. The path is relative to the object triggering the alert. Example: %{/worstParam} will contain the name of the worst parameter on this instance, which is an application instance built-in variable (see the &quot;PATROL Script Language Reference&quot; document).</td>
</tr>
</tbody>
</table>
Object Specific Macros

The macros listed below can be used with alert actions specifically for their respective object type.

- **Command Line Macros**

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_COMMANDLINE}</td>
<td>Command line being executed and analyzed.</td>
</tr>
<tr>
<td>%{SEN_EXITSTATUSCODE}</td>
<td>Exit status returned by the system after executing the command.</td>
</tr>
</tbody>
</table>
Database Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{%SEN_DATABASENAME%}</td>
<td>Name of the database the SQL query is sent to. May be the database name for SQL Server, or the Oracle SID for Oracle.</td>
</tr>
<tr>
<td>{%SEN_DATABASETYPE%}</td>
<td>Type of the database.</td>
</tr>
<tr>
<td>{%SEN_QUERY%}</td>
<td>SQL statement sent for execution.</td>
</tr>
</tbody>
</table>

Dynamic Object Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{%SEN_RESULT%}</td>
<td>Returns the output of the dynamic object.</td>
</tr>
</tbody>
</table>

Dynamic Value Map Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>% (SEN_RETAINEDVALUE)</td>
<td>Value retained by the collect and mapped to a status.</td>
</tr>
<tr>
<td>% (SEN_MAPPEDSTATUSINFORMATION)</td>
<td>Provides additional information about the Status of the value mapping result, as reported by the StatusInformation parameter.</td>
</tr>
</tbody>
</table>

File Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{%SEN_FILENAME%}</td>
<td>Name of the monitored file as entered in the GUI.</td>
</tr>
<tr>
<td>{%SEN.MONITOREDFILE%}</td>
<td>Current file being monitored.</td>
</tr>
</tbody>
</table>

MFile System Macro

<table>
<thead>
<tr>
<th>Macro</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{%SEN_FILESYSTEM%}</td>
<td>Name of the monitored file system.</td>
</tr>
</tbody>
</table>
Folder Macros

<table>
<thead>
<tr>
<th>Macro</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_FOLDER}</td>
<td>Folder being monitored.</td>
</tr>
<tr>
<td>%{SEN_OLEDESTFILESINFOFOLDER}</td>
<td>Name of the oldest file in the folder.</td>
</tr>
</tbody>
</table>

Group Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_GROUP_CLASS}</td>
<td>Class name of the Monitor Group.</td>
</tr>
<tr>
<td>% {SEN_GROUP_COLLECTIONERRORS}</td>
<td>List of collection errors that occurred between the current collect and the previous one</td>
</tr>
<tr>
<td>%{SEN_GROUP_CONTACT}</td>
<td>Contact information in case of a Group failure.</td>
</tr>
<tr>
<td>% {SEN_GROUP_DESCRIPTION}</td>
<td>Description of the Group.</td>
</tr>
<tr>
<td>%{SEN_GROUP_ID}</td>
<td>PATROL ID of the Group triggering the alert.</td>
</tr>
<tr>
<td>%{SEN_GROUP_LABEL}</td>
<td>Display name of the Group triggering the alert action.</td>
</tr>
<tr>
<td>%{SEN_GROUP_TYPE}</td>
<td>Type of the Group triggering the alert (&quot;Group&quot;).</td>
</tr>
</tbody>
</table>

Host Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_AVAILABILITYCHECKS}</td>
<td>List of availability checks, separated by commas.</td>
</tr>
<tr>
<td>%{SEN_CREDENTIALLIST}</td>
<td>List of credentials, separated by commas.</td>
</tr>
<tr>
<td>%{SEN_SIGNATUREFILES}</td>
<td>List of signatures files, separated by commas.</td>
</tr>
<tr>
<td>%{SEN_TCP_PORT}</td>
<td>Port number used for the TCP availability check.</td>
</tr>
</tbody>
</table>

Multi-Parameter Formula Macro

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_FORMULA}</td>
<td>User-defined formula used to calculate the parameter value.</td>
</tr>
</tbody>
</table>

Process Macros
### Specifying Alert Actions

**Monitoring Studio KM for PATROL 9.4.01**

#### SNMP Polling Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_CONTENT}</td>
<td>Value of the OID being polled.</td>
</tr>
<tr>
<td>%{SEN_OID}</td>
<td>SNMP OID being polled.</td>
</tr>
</tbody>
</table>

#### SNMP Trap Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_CONTENT}</td>
<td>Content of the found trap.</td>
</tr>
<tr>
<td>%{SEN_ENTERPRISEID}</td>
<td>Enterprise ID (OID) of the SNMP traps being looked for.</td>
</tr>
<tr>
<td>%{SEN_FOUNDIP}</td>
<td>Actual originating IP address of the trap that has been received.</td>
</tr>
<tr>
<td>%{SEN_FOUNDTRAPNUMBER}</td>
<td>Actual SNMP trap number that has been received and matches the entered criteria.</td>
</tr>
<tr>
<td>%{SEN_TRAPNUMBER}</td>
<td>SNMP Trap numbers (specific numbers) being looked for.</td>
</tr>
</tbody>
</table>
### String Search Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_LASTMATCHINGLINE}</td>
<td>Last line that matches with the string search criteria.</td>
</tr>
<tr>
<td>%{SEN_LASTMATCHINGLINES}</td>
<td>Last 10 lines that match with the string search criteria.</td>
</tr>
<tr>
<td>%{SEN_STRING1}</td>
<td>First regular expression being searched for.</td>
</tr>
<tr>
<td>%{SEN_STRING2}</td>
<td>Second regular expressions being searched for.</td>
</tr>
</tbody>
</table>

### Value Map Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>% (SEN_RETAINEDVALUE)</td>
<td>Value retained by the collect and mapped to a status.</td>
</tr>
<tr>
<td>% (SEN_MAPPEDSTATUSINFORMATION)</td>
<td>Provides additional information about the status of the value mapping result, as reported by the StatusInformation parameter.</td>
</tr>
</tbody>
</table>

### WBEM Query Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_NAMESPACE}</td>
<td>Namespace of the WBEM query.</td>
</tr>
<tr>
<td>%{SEN_QUERY}</td>
<td>WBEM statement sent for execution.</td>
</tr>
</tbody>
</table>

### Web Request Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_HTTPMETHOD}</td>
<td>GET or POST depending on what was selected in the GUI.</td>
</tr>
<tr>
<td>%{SEN_URL}</td>
<td>URL being tested.</td>
</tr>
</tbody>
</table>

### WMI Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_NAMESPACE}</td>
<td>Namespace of the WMI query.</td>
</tr>
<tr>
<td>%{SEN_QUERY}</td>
<td>WMI statement sent for execution.</td>
</tr>
</tbody>
</table>
### Windows Event Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_CONTENT}</td>
<td>Message content of the last matching event.</td>
</tr>
<tr>
<td>%{SEN_EVENTID}</td>
<td>ID of the Windows events being searched for.</td>
</tr>
<tr>
<td>%{SEN_EVENTLOG}</td>
<td>Name of the Windows event log being monitored.</td>
</tr>
<tr>
<td>%{SEN_MATCHINGEVENTS}</td>
<td>List of matching events.</td>
</tr>
<tr>
<td>%{SEN_PROVIDER}</td>
<td>Windows Event source whose new entries are monitored.</td>
</tr>
<tr>
<td>%{SEN_RECORDNUMBER}</td>
<td>Last matching event record number</td>
</tr>
</tbody>
</table>

### Windows Performance Macros

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_PERFORMANCECOUNTER}</td>
<td>Windows performance counter being monitored.</td>
</tr>
<tr>
<td>%{SEN_PERFORMANCEINSTANCE}</td>
<td>Windows performance object instances being monitored.</td>
</tr>
<tr>
<td>%{SEN_PERFORMANCEOBJECT}</td>
<td>Windows performance object name being monitored.</td>
</tr>
</tbody>
</table>

### Windows Service Macro

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%{SEN_SERVICENAME}</td>
<td>Name of the service of a Windows service class.</td>
</tr>
</tbody>
</table>
Format Symbols for %{SEN_TIME:...} Macros

The following table recapitulates all of the time formats available in the %{SEN_TIME:...} and %{SEN_LASTTIME:...} macros in the Command Line execution wizard, the File monitoring and analysis wizard, and the Alert Actions wizard.

<table>
<thead>
<tr>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%%</td>
<td>This symbol allows you to use a percent sign (%) in the format of a date string</td>
</tr>
<tr>
<td>%a</td>
<td>Locale's abbreviated name of the day of week</td>
</tr>
<tr>
<td>%A</td>
<td>Locale's full name of the day of week</td>
</tr>
<tr>
<td>%b</td>
<td>Locale's abbreviated name of the month</td>
</tr>
<tr>
<td>%B</td>
<td>Locale's full name of the month</td>
</tr>
<tr>
<td>%c</td>
<td>Locale's appropriate date and time representation</td>
</tr>
<tr>
<td>%C</td>
<td>Data and time as %c</td>
</tr>
<tr>
<td>%d</td>
<td>Day of month [1,31]; single digits are preceded by 0</td>
</tr>
<tr>
<td>%D</td>
<td>Date as %m/%d/%y</td>
</tr>
<tr>
<td>%e</td>
<td>Day of month [1,31]; single digits are preceded by a space</td>
</tr>
<tr>
<td>%h</td>
<td>Locale's abbreviated name of the month</td>
</tr>
<tr>
<td>%H</td>
<td>Hour (24-hour clock) [0,23]; single digits are preceded by 0</td>
</tr>
<tr>
<td>%I</td>
<td>Hour (12-hour clock) [1,12]; single digits are preceded by 0</td>
</tr>
<tr>
<td>%j</td>
<td>Day of year [1,366]; single digits are preceded by 0</td>
</tr>
<tr>
<td>%k</td>
<td>Hour (24-hour clock) [0,23]; single digits are preceded by a space</td>
</tr>
<tr>
<td>%l</td>
<td>Hour (12-hour clock) [1,12]; single digits are preceded by a space</td>
</tr>
<tr>
<td>%m</td>
<td>Month as a decimal number [1,12]; single digits are preceded by 0</td>
</tr>
<tr>
<td>%M</td>
<td>Minute [0,59]; leading zero is permitted but not required</td>
</tr>
<tr>
<td>%n</td>
<td>Insert a new line</td>
</tr>
<tr>
<td>%p</td>
<td>Locale’s equivalent of either a.m. Or p.m.</td>
</tr>
<tr>
<td>%r</td>
<td>Appropriate time representation in 12-hour clock format with %p</td>
</tr>
<tr>
<td>%R</td>
<td>Time as %H:%M</td>
</tr>
<tr>
<td>%s</td>
<td>Seconds [0,61]</td>
</tr>
<tr>
<td>%t</td>
<td>Insert a tab</td>
</tr>
<tr>
<td>%T</td>
<td>Time as %H:%M:%S</td>
</tr>
<tr>
<td>%u</td>
<td>Day of week as a decimal number [1,7], with 1 representing Monday</td>
</tr>
<tr>
<td>%U</td>
<td>Week of the year as a decimal number [0,53], with Sunday as the first day of week 1</td>
</tr>
<tr>
<td>%V</td>
<td>Week of the year as a decimal number [01,53], with Monday as the first day of the week. If the week containing 1 January has four or more days in the new year, then it is considered week 1; otherwise, it is week 53 of the previous year, and the next week, is, week 1.</td>
</tr>
<tr>
<td>Format</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>%w</td>
<td>Day of week as a decimal number [0,6], with 0 representing Sunday</td>
</tr>
<tr>
<td>%W</td>
<td>Week of the year as a decimal number [0,53], with Monday as the first day of week 1</td>
</tr>
<tr>
<td>%x</td>
<td>Locale's appropriate date representation</td>
</tr>
<tr>
<td>%X</td>
<td>Locale's appropriate time representation</td>
</tr>
<tr>
<td>%y</td>
<td>Year within century [0,99]</td>
</tr>
<tr>
<td>%Y</td>
<td>Year, including the century (for example 1993)</td>
</tr>
<tr>
<td>%z</td>
<td>Abbreviated or full name of time zone, or no bytes if no information of the time zone exists</td>
</tr>
<tr>
<td>%Z</td>
<td>Locale's alternative appropriate date and time representation</td>
</tr>
<tr>
<td>%c</td>
<td>Name of the base year (period) in the locale's alternative representation</td>
</tr>
<tr>
<td>%c</td>
<td>Locale's alternative date representation</td>
</tr>
<tr>
<td>%e</td>
<td>Locale's alternative time representation</td>
</tr>
<tr>
<td>%E</td>
<td>Offset from %c (year only) in the locale's alternative representation</td>
</tr>
<tr>
<td>%E</td>
<td>Alternative representation of the year in full</td>
</tr>
<tr>
<td>%d</td>
<td>Day of the month using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%e</td>
<td>Same as %d</td>
</tr>
<tr>
<td>%H</td>
<td>Hour (24-hour clock) using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%I</td>
<td>Hour (12-hour clock) using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%m</td>
<td>Month using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%M</td>
<td>Minutes using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%S</td>
<td>Seconds using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%U</td>
<td>Week of the year (Sunday as the first day of the week) using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%W</td>
<td>Day of week (Sunday=0) using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%W</td>
<td>Week of the year (Monday as the first day of the week) using the locale's alternative numeric symbols</td>
</tr>
<tr>
<td>%y</td>
<td>Year (offset from %C) in the locale's alternative representation and using the locale's alternative numeric symbols</td>
</tr>
</tbody>
</table>
Alert Actions Example

In this example, we monitor a log file and look for the string "error" in the lines. Every time an
"error" is found, we want to trigger a standard PATROL event containing the name of the
parameter, the name of the log file, the content of the line that triggered the alert and the name
of the technology.

Before setting the alert action, you will first have to make sure that the thresholds for the
MatchingLineCount Parameter is set to. To do so:
1. Right-click the **String Search** icon and select **Thresholds**...

2. Select the **MatchingLineCount** parameter and click **Next**.
3. Verify that an alarm is triggered if the value is greater than 1 and execute the Alert Actions "every time a matching line is found”.

4. Click Set thresholds.

**To Set the Alert Action**

1. Right-click the String Search icon and select the Alert Actions... command from the menu to run the Alert Actions wizard:

![Alert Actions wizard](image)

3. Select the MatchingLineCount parameter and click Next.

4. Select the type of action to be executed when the MatchingLineCount parameter goes into alert state:
   - Check the Trigger an event box and click Next.
   - Enter the message of the event: Name of the Group ; Name of the log file ; The error message (i.e. the line in the log file that triggered the alert)
   - Enter the text below in the box called Enter the text to be sent with the event:

   ```
   %SEN_GROUP_LABEL: error found in %SEN_PARENT_LABEL.%SEN_NEWLINE
   Error message: %SEN_LASTMATCHINGLINE
   %SEN_GROUP_LABEL contains the Group display name in the PATROL Console.
   %SEN_PARENT_LABEL contains the log file display name (it is the parent of the String search in the PATROL Console).
   %SEN_NEWLINE will create a new line in the message.
   %SEN_LASTMATCHINGLINE contains the entire line in the log file that triggered the alert.
   ```
Setting the Alert Parameters

The text sent with the event will look like this:

```
Group: Lync: error found in Log File: C:\Program Files\Lync\Lync_Error.log
Error message: server connection error - main server is down.
```

5. Click the **Next** button to get to the final panel of the Alert Actions wizard that summarizes the Alert Actions set for the **MatchingLineCount** parameter.

In this example you just set one Alert Action, **Trigger an event**, but it would have been possible to set several Alert Actions, such as a command line that runs a recovery action for the monitored application or writes an annotation to the graph built by the **MatchingLineCount** parameter with the content of the matching line.
Setting Credentials

For security purpose, some technologies may require additional privileges to allow access to their data. In this case, Monitoring Studio enables users to:

- **Set new Specific Credentials**
- **Modify existing Specific Credentials**
- **Delete existing Specific Credentials**

⚠️ System Credentials cannot be edited or deleted using the below menu commands. Use the "Edit" menu command from the Host icon to edit System Credentials.

To add new credentials for local or Windows hosts

1. Right-click the **Host** icon > **KM Commands > Credentials > New Credentials**...

![New Credentials](image)

1. **Current Credentials**: Click this button to display the list of previously configured credentials.
2. Provide the following information:
   - **Username**: Enter the username on the target host.
   - **Password**: Enter the corresponding password for the user. Leave this field empty if, for security reasons, you want Monitoring Studio to prompt for the password every time you add or edit a Monitor using specific credentials.
   - **(Optional) Associated OpenSSH Private Key File Path**: This option is only available for monitoring remote hosts running UNIX, Linux or other types of operating system that support SSH authentication key file.
   - **(Optional) Label**: Enter the label for the new credentials.
If no label is provided, the username will be used as the label and set by the credentials wizard.

4. Click OK. The new credentials are successfully added and ready to use.

To add new credentials for remote UNIX/Linux hosts

2. Right-click the Host icon > KM Commands > Credentials > New Credentials...

3. **Current Credentials**: Click this button to display the list of previously configured credentials.

4. Provide the following information:
   - **Username**: Enter the username on the target host.
   - **Password**: Enter the corresponding password for the user. Leave this field empty if, for security reasons, you want Monitoring Studio to prompt for the password every time you add or edit a Monitor using specific credentials.
   - (Optional) **Label**: Enter the label for the new credentials.

   If no label is provided, the username will be used as the label and set by the credentials wizard.

6. (Optional) **Associated OpenSSH Private Key File Path**: When monitoring remote hosts running UNIX, Linux or other types of operating systems that support SSH authentication key file, you may need to provide an OpenSSH private key file to establish a secured connection with the remote host. Enter the path of the OpenSSH private key file you wish to use to establish a connection with the remote host and enter the optional PassPhrase in the Password field.

   The Private Key File should exists on the PATROL Agent node.
7. Click **OK**. The new credentials are successfully added and ready to use.

**To edit credentials**

1. Right-click the Host icon > KM Commands > Credentials > Edit Credentials...

2. Select the **Credentials** to edit from the drop-down list and click **Next**.
The screen provided below as an example may vary according to the target host system type.

3. Monitors using these credentials: Click this button to see which Monitor(s) is/are currently using these Credentials.

4. Modify the following information:
   - **Username**: Edit the username on the target host.
   - **Password**: Edit the corresponding password for the user. Leave this field empty if, for security reasons, you want Monitoring Studio to prompt for the password every time you add or edit a Monitor using specific credentials.
   - (Optional) **Label**: Edit the label for the new credentials.
   - (Depending on the target host) **Associated OpenSSH Private Key File Path**: When monitoring remote hosts running UNIX, Linux or other types of operating systems that support SSH authentication key file, you may need to provide an OpenSSH private key file to establish a secured connection with the remote host. Enter the path of the OpenSSH private key file you wish to use to establish a connection with the remote host and enter the optional PassPhrase in the Password field.

⚠️ The Private Key File should exists on the PATROL Agent node.
5. Click **Finish** to validate your changes.

**To delete credentials**

1. Right-click the **Host icon > KM Commands > Credentials > Delete Credentials...**

![Delete Credentials](image)

2. Select the Credentials to delete from the drop-down list and click **Next**.

   - **Warning**: Credentials that are currently being used by any Monitor cannot be deleted.
3. Click **Finish** to confirm and permanently delete the selected credentials.
Setting the Discovery Interval

By default, Monitoring Studio performs a discovery every hour to create, modify, and update monitored objects, if necessary. However, you can easily change the default discovery interval to meet your specific needs.

1. Right-click the Monitoring Studio icon > KM commands > KM Settings > Discovery Interval...

2. Use the spin button to customize the discovery interval from once every 5 minutes to once every 24 hours.

3. Click OK to save your settings.

You can also force a discovery manually by right-clicking on the Monitoring Studio icon > KM Commands > Trigger a KM Discovery.
Setting the Polling Interval

A polling interval defines how often new data is collected. A new collect can be performed from once every second, to once in a week. Polling intervals can be set for objects created by Monitoring Studio that collect data (files, processes, command lines, SNMP polling, etc.). By default, the polling interval is set to 2 minutes on all objects, which can be modified at any time.

⚠️ The option to set polling intervals is not available for String Searches, Numeric Values, Text Pre-Processing, Monitor Groups, and SNMP Trap instances, since either they do not have collectors, or as in the case of SNMP Traps – have collectors that react to events.

To set the polling interval:

1. In the PATROL Console, right-click the object icon > KM commands > Polling Interval...
2. Configure the polling interval options:
   - **Collect every: x hours x minutes x seconds**: Set the polling interval in hours, minutes, and seconds.
   - **Collect once a day at: x hours (24 hours) x minutes x seconds**: Here the values selected indicate the time of day. Example: 14 hours 30 minutes 0 seconds would mean that the polling is done only once a day at 2:30pm (14:30 hrs)
   - **Collect once a week on <weekday> at: x hour x minutes x seconds**: Here the values selected indicate the time of the selected weekday. Example: Monday at 14 hours 30 minutes 0 seconds would mean that the polling is done only once a week on Mondays at 2:30pm (1430 hrs)
3. (For File Systems/Processes/Windows Events only) **Monitoring Studio** relies on a cache mechanism to share the information among the Monitors in order to use as little resources as possible on the target host and over the network. The cache will be refreshed if one of the Monitors needs to collect data (polling interval reached) and the cache is older than the selected minimum cache refresh. Configure the cache refreshing frequency:
   - **Minimum File System/Process/Windows Event cache refresh interval**: Use the spin button to set the minimum number of seconds Monitoring Studio must wait before refreshing the cache. Default is 15 seconds.
Setting the Polling Interval

4. Click **OK** to validate your changes.
Visualizing Running Processes

The **Process Viewer** tool displays all the processes that are currently running on a host as well as their characteristics to help you monitor them in the most efficient way.

To access the **Process Viewer** tool:

1. Right-click the desired **Host** icon > **KM commands** > **Tools** > **Process Viewer**...

2. Click:
   - **Update** to refresh the process list
   - **Close** to close the **Process Viewer** tool.

   - The **Process Viewer** tool uses the system credentials provided during Host configuration.
Visualizing SNMP Traps

For localhost monitoring, the Real-time SNMP Trap Listener tool allows you to visualize all of the SNMP traps and their characteristics received by the PATROL Agent and Monitoring Studio. This tool is particularly helpful when you want to setup a SNMP Trap listening object in Monitoring Studio but you ignore the characteristics of the traps you want to detect.

⚠️ Only SNMP version 1 is supported for SNMP trap listening. SNMP v2c and v3 traps are not supported.

To access the SNMP Trap Listener tool

1. Right-click the main Monitoring Studio icon > KM commands > Tools > Real-time SNMP Trap Listener.
   - The incoming SNMP traps are shown in this window as soon as they arrive.
   - You can view their main characteristics (originating IP address, community, Enterprise OID, and trap specific number) as well as their attached varbinds (variable bindings). This will help you specify the search criteria in the SNMP Trap Listening wizard.
   - The newest (or latest arrived) trap is shown first in the list.
2. Pause if you wish to stop the reception of traps and have more time to analyze the characteristics of the previously arrived traps.
3. Resume to resume the listening.
4. Close to quit the tool window.

⚠️ Due to an SNMP protocol limitation, it is not possible to have more than one program on one computer listening to SNMP traps (handling the UDP/162 port). If another program is listening to SNMP traps, Monitoring Studio is not able to listen for SNMP traps and an error message is shown. For the same reason, it is not possible to use this tool and listen for SNMP traps from a Monitoring Studio object in the PATROL Console. It may just be one or the other.
Visualizing a Host Windows Event Log

The Windows Event Log Reader tool displays all Windows Events registered in a specific event log during the last selected period. It helps to identify the provider, event ID and event level for configuring Windows Event Log Monitors.

To access the Windows Event Log Reader tool
1. Right-click the desired Host icon > KM commands > Tools > Windows Event Log Reader...

2. Select the Event log and the time span you wish to view, and click the Update button to refresh the window. This may take a few seconds to retrieve the events from the host. They are displayed in the order they are registered.

3. Click:
   - **Update** to refresh the information displayed.
   - **Close** to quit the Windows Event log Reader tool.

   The Windows Event Log Reader tool uses the system credentials provided during Host configuration.
Visualizing a Monitored File Content

Administrators can use the File Viewer tool to check the content of a file that is being monitored.

To view the content of a monitored file
1. Right-click the Host icon > KM commands > Tools > File Viewer... or the File icon > KM commands > View File Content
2. Specify the visualization options:
   - **File path**: Path of the file that you wish to view
   - **Show me**: The first/last: x KB: Indicate the size and part of the file that you wish to view
   - **Only the lines matching with**: (optional regular expression) Enter a regular expression and only the lines matching this regular expression are displayed in the File content field.

3. Click **Update**. The file content is now displayed.
4. Click **Close**.

If the tool is accessed from File object, the File Viewer tool uses the credentials assigned to the File Monitor; if this tool is accessed from the Host icon, the System Credentials configured for the Host are used.
Visualizing Properties

Administrators can use the Properties KM command to display a Configuration Report for the monitored object of their choice at a given instant. This tool can be used to verify the configuration of a monitored object and all its children objects.

To visualize properties

1. In the PATROL Console, right-click the **object** icon > **KM commands** > **Properties** or the Monitoring Studio icon > **KM Commands** > **Configuration** > **Show Configuration**...

A new task window opens and displays the Monitoring Studio Configuration Report.

---

**Monitoring Studio Configuration Report**

- **Date:** 05/18/15 - 15:29:25
- **Host:** 1
- **Object ID:** MyGroup@172.16.16.46
- **Object Display Name:** Host 1
- **Object Type:** Host
- **Hostname:** 172.16.16.46
- **FQDN:** 172.16.16.46
- **IP:** 172.16.16.46
- **Domain:** internal.sentrysoftware.net
- **System type:** Windows
- **Credentials List:** System Credentials
- **SNMP Community:** Sentry
- **Availability Checks:** Ping
- **Disable Monitors if Unreachable:** yes
- **Polling Interval:** Every 2 minute(s)

---

**Parameters Setup**

- **Status:** activated
- **BORDER:** No Threshold
- **ALARM:** Trigger a WARNING if value is between 1 and 1 (i.e. equal to 1); 1 time(s) in a row
- **ALARM:** Trigger an ALARM if value is between 2 and 2 (i.e. equal to 2); 1 time(s) in a row

---

A new task window opens and displays the Monitoring Studio Configuration Report.
Visualizing SNMP Agent Variables

The **SNMP Browser** tool allows you to easily visualize the variables (OID and values) available in an SNMP agent. This tool is very useful to identify which OID you need to poll in the SNMP Polling wizard.

> It is recommended to turn off the debug mode when using this KM Command.

**To set the SNMP Browser tool**

1. Right-click a **Host** icon > **KM commands > Tools > SNMP Browser**.
2. **Start SNMP walking from this root OID**: Specify the root OID from which you want to perform a SNMP walk.
3. Click the **Update** button. If the information entered is correct, the result of the SNMP walk is shown a few seconds later. If not, an error message is displayed with additional information about the failure. You can use the same settings when you setup a new SNMP Polling object in Monitoring Studio.

> Performing an SNMP walk on a remote SNMP agent that has a large amount of variables may take a long time.

4. Click **Close** to quit the SNMP Browser tool.
Troubleshooting
This section provides some troubleshooting guidelines for Monitoring Studio. It describes how the KM debugging can be enabled to diagnose any issues and then states the most frequently asked questions based on issues encountered by customers.

**Enabling the Debug Mode**

If you encounter an issue, and want to report it to Sentry Software, you will be asked to enable the **Debug Mode** and provide the debug output to the Sentry Software support team. The debug output file will allow you to:

- have debug information about the discovery process since debug information about the discovery process may be lost by the PATROL Console during the PATROL Agent startup.
- trace the activity of Monitoring Studio for a few minutes since information may also be lost by the PATROL Console if its buffer is full.

**To enable the debug mode**

1. Right-click the main Monitoring Studio icon > KM Commands > KM Settings > Debug...

![Enabling Debug Mode](image)
2. Check the **Enable Debug Mode** option.
3. Indicate the date and time at which the system must stop logging debug information. The required format is: YYYY/mm/dd HH:MM:SS, based on a 24 hour-day.
4. Click **OK**. The debug files will automatically be saved in the %PATROL_HOME%\log or $PATROL_HOME/log folder.

**The following debug files are generated:**

- `SEN_MS_debug_km_<PatrolAgent_Port>_<YYYY-mm-dd-HH-MM>.log`, with the debug output of the KM (example: `SEN_MS_debug_km_3181_2016-02-12-17-25.log`)
- `SEN_MS_CollectionHub_debug_psl_<PatrolAgent_Port>_<YYYY-mm-dd-HH-MM>.log`, with the PSL debug output of the Collection Hub (example: `SEN_MS_CollectionHub_debug_psl_3181_2016-01-27-09-27.log`)
- `SEN_MS_CollectionHub_debug_java_<PatrolAgent_Port>_<YYYY-mm-dd-HH-MM>.log`, with the Java debug output of the Collection Hub (example: `SEN_MS_CollectionHub_debug_java_3181_2016-01-27-09-27.log`
Frequently Asked Questions and Problems

How Do I Know Which Version of Monitoring Studio I Am Running?

To find out which version of Monitoring Studio KM for PATROL you are running, right-click the main Monitoring Studio icon > KM Commands > About... The Monitoring Studio version and release date are displayed in the dialog box that pops up.

To know if your version of Monitoring Studio is up-to-date, please check the BMC Software Web site at www.bmc.com or the Sentry Software Web site at www.sentrysoftware.com for the latest information and product versions. If the "Unlimited" option has been selected.

Although SNMP Trap Listening Seems to Work, No Trap Matches My Criteria

Use the built-in SNMP Trap Listener tool of Monitoring Studio to visualize the incoming traps and their characteristics. Verify the originating IP address and the community used to generate the trap.

Check that the SNMP agent that generates the traps is properly configured to send them to the computer, where Monitoring Studio and the PATROL Agent are running.

Monitoring Studio Is Unable to Listen to SNMP Traps

Because of SNMP protocol limitations, it is not possible to have several programs listening to SNMP traps on the same computer (at least on the same network interface). Only one program can listen to the UDP/162 port for incoming SNMP traps.

If you have another program that is listening to SNMP traps, Monitoring Studio will not be able to listen to incoming SNMP traps.

Please stop any other program that listens to SNMP traps (typically the SNMP Traps Windows service).
Can I Monitor the Processor Time Usage Made by a Windows Service?

You cannot directly monitor the processor time usage of a Windows service.

1. First identify the process (its characteristics) that corresponds to this process
2. Then, simply add the monitoring of this process with the Process monitoring wizard

⚠️ It can be difficult to identify the process associated to a given Windows service because one single program can handle/host several Windows services and one program (e.g. svchost.exe)

Monitoring Studio Reads My Log File Entirely

If you erroneously selected the flat file option when you actually wanted to monitor a log file, you will have to redo the whole configuration: create a new file monitoring and choose the log file option, and then copy and paste the String and Numeric Value searches from your previous file monitoring to the new one.

It is very important to properly configure a file monitoring by first indicating the type of the file being monitored: flat or log.

- A flat file is updated entirely and so must be read entirely, at each poll.
- In log files, new lines are appended at the end of the file. Only these new lines need to be analyzed.

I'm Unable to Poll an SNMP Agent (Getting a Warning)

Please check the IP address or, if you use a hostname, that the hostname can be resolved.

Check the community and ensure that the SNMP agent that is running on the target computer uses SNMP v1 and matches what has been set in Monitoring Studio.

Use a MIB Browser utility to identify the OID that matches your needs.

You can also use the built-in SNMP Browser tool of Monitoring Studio to check your settings: Right-click on a Host icon > KM commands > Tools > SNMP Browser.
I'm Unable to See the Monitoring Studio Icons in PATROL Central

In order to have the Monitoring Studio icons in PATROL Central, you need to install Monitoring Studio on the Agents (managed systems) as well as on the Console Server.

Please read the Monitoring Studio Installation Guide for more information about how to setup Monitoring Studio.

Infinite Loop Reported in the PATROL Agent Log

In some cases, the PATROL Agent may report a possible infinite loop in its log file (*.errs) or in the System Output Window in the PATROL Console, as in the following:

**Tue May 4 17:12:28 2004 PatrolAgent-W-EUSER: PSL script 'fileColl', 'File: myLogFile.LOG', 'myApp#myLogFileLOG' may be in an infinite loop - executed 500016 instructions**

While this message may be worrying, it is only due to an outdated loop mechanism of the PATROL Agent, as default settings often do not suit the computing power of recent machines.

Monitoring Studio has internal mechanisms, which prevent it from consuming too much processor time (for example, it will not parse more than 8 megabytes of a file at one time, it will not process more than 1,000 files in one folder, etc.).

The loop detection settings of the PATROL Agent can be changed and Sentry Software recommends that users set these upper thresholds. This can be achieved by modifying the following configuration variables with WPCONFIG or xpconfig:

- `/AgentSetup/AgentTuning/pslInstructionMax = 1,000,000`
- `/AgentSetup/AgentTuning/pslInstructionPeriod = 1`
- This basically disables the loop detection mechanism of the PATROL Agent

**Disabling the loop detection mechanism of the PATROL Agent does have a drawback: you will not be able to detect infinite loops in other KMs anymore. Sentry Software recommends that you use this tip only when necessary and that you set the thresholds back to the default if your PATROL Agent does not behave properly.**
Is It Possible to Monitor the CPU/Memory Usage of a Process Tree?

Monitoring Studio is able to measure the processor time usage and memory usage of a group of processes like a tree (a main process, its children, etc.).

Monitoring Studio can also count the number of children of the process that match the specified criteria. To do so: Select the ChildCount parameter in the Process monitoring wizard.

Monitoring Studio Fails to Authenticate to My Web Server

Monitoring Studio supports only HTTP authentication, which is different from an HTML-based login form.

To recognize HTTP authentication from HTML-based login forms, use a Web browser to connect to the URL you want to request. If a popup dialog appears, asking for username and password, it’s HTTP authentication.

Most Internet Web sites do NOT use HTTP authentication but an HTML-based login form (like your favorite social network).

In case of HTTP authentication, Monitoring Studio will use the specified credentials to access the URL. Most authentication schemes are supported but the "Negotiate" and "NTLM" schemes will require Java 1.8 (which ships with Monitoring Studio).

In case of HTML-based authentication, Monitoring Studio may be able to access the URL, provided that:

- You POST to the proper login URL
- You include the username and password in request body
- You specify in a variable (either in the login URL or in the request body) the URL you wish to be redirected to once the server has validated your credentials.

To find the proper login URL and variable names, you will need to carefully study the HTML source of the login page.
Monitoring Studio Does Not Follow an HTTP Redirection (Web)

Some HTTP redirections are not implemented as true HTTP redirection (HTTP status codes 300, 301, 302, etc.) but are instead HTML pages with specific meta-tags, or even HTML pages with Javascript code performing the redirection from the browser.

Monitoring Studio only supports true HTTP redirection and does not support other types of redirection that require advanced browser features.

Monitoring Studio Is Not Able to Post a Web Form to My Web Server

Verify that the URL that you entered is not the URL of the page that displays the form to fill in but the URL indicated in the `<FORM ACTION="<url>"` tag in the HTML source of this page.

Verify that the names of the variables to post are correct. They are identified by the `<INPUT NAME="<variableName>"` and `<SELECT NAME="<variableName>"` tags in the HTML source of the page that shows the form to fill in.

The `<INPUT TYPE="SUBMIT" NAME="...">` variable may be required for the form to be properly processed by the Web server.
Introduction

This chapter contains tables describing the parameters used in the KM, grouped by Application Classes, and provides a brief description of each parameter and its default settings. It also gives you some additional information about HTTP authentication, Internal Identifiers, Performance Counters, Processes, Regular expressions and WMI.
Application Classes and Parameters

This section lists the application classes of Monitoring Studio KM for PATROL. It gives you details on the icons that represent the class, Infobox, parameters, and the menu commands available for each application class.

If the KM is properly loaded, the following application classes should be loaded on the monitored system and console:

<table>
<thead>
<tr>
<th>Application Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEN_MS_COMMANDLINE</td>
<td>Executes, monitors, and parses command lines and scripts</td>
</tr>
<tr>
<td>SEN_MS_DBQUERY</td>
<td>Executes and monitors database queries</td>
</tr>
<tr>
<td>SEN_MS_DYNAMIC</td>
<td>Creates Dynamic Items</td>
</tr>
<tr>
<td>SEN_MS_DYNAMIC_CONT</td>
<td>Creates a dynamic object container</td>
</tr>
<tr>
<td>SEN_MS_DYNAMICDISCOVERY</td>
<td>Creates a dynamic object builder</td>
</tr>
<tr>
<td>SEN_MS_DYNAMICNUMBER</td>
<td>Performs dynamic numeric value extractions</td>
</tr>
<tr>
<td>SEN_MS_DYNAMICSTRING</td>
<td>Executes dynamic string searches</td>
</tr>
<tr>
<td>SEN_MS_DYNAMICVALUEMAP</td>
<td>Extracts values from result outputs of other Monitors and map these values to user-defined status that match your specific needs</td>
</tr>
<tr>
<td>SEN_MS_FILE</td>
<td>Monitors files and file content</td>
</tr>
<tr>
<td>SEN_MS_FILESYSTEM</td>
<td>Monitors file systems</td>
</tr>
<tr>
<td>SEN_MS_FOLDER</td>
<td>Monitors directories</td>
</tr>
<tr>
<td>SEN_MS_FORMULA</td>
<td>Monitors values derived from other parameters</td>
</tr>
<tr>
<td>SEN_MS_GROUP</td>
<td>Groups all the hosts</td>
</tr>
<tr>
<td>SEN_MS_HOST</td>
<td>Monitors hosts and allows adding more monitors</td>
</tr>
<tr>
<td>SEN_MS_MAIN</td>
<td>Main application class where groups are created from</td>
</tr>
<tr>
<td>SEN_MS_MONITORGROUP</td>
<td>Monitor Group. Groups monitors within a host</td>
</tr>
<tr>
<td>SEN_MS_NAGIOSPERF</td>
<td>Monitors Nagios plugins performance metrics</td>
</tr>
<tr>
<td>SEN_MS_NAGIOSPLUGIN</td>
<td>Monitors Nagios plugins status and execution time</td>
</tr>
<tr>
<td>SEN_MS_NUMBER</td>
<td>Extracts numeric values</td>
</tr>
<tr>
<td>SEN_MS_PROCESS</td>
<td>Monitors processes</td>
</tr>
<tr>
<td>SEN_MS_PSLCOMMAND</td>
<td>Executes, monitors, and parses PSL commands</td>
</tr>
<tr>
<td>SEN_MS_SNMPPOLLING</td>
<td>Polls SNMP devices</td>
</tr>
<tr>
<td>SEN_MS_SNMPTRAP</td>
<td>Listens for SNMP traps</td>
</tr>
<tr>
<td>SEN_MS_STRING</td>
<td>Executes string searches</td>
</tr>
<tr>
<td>SEN_MS_TRANSFORM</td>
<td>Transforms complex (multi-line, HTML, XML) text to enable string/numeric value searches</td>
</tr>
<tr>
<td>Application Class</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SEN_MS_VALUEMAP</td>
<td>Extracts values from result outputs of other Monitors and map these values to user-defined status that match your specific needs.</td>
</tr>
<tr>
<td>SEN_MS_WBEMQUERY</td>
<td>Performs WBEM queries</td>
</tr>
<tr>
<td>SEN_MS_WEBREQUEST</td>
<td>Executes Web requests</td>
</tr>
<tr>
<td>SEN_MS_WINEVENT</td>
<td>Monitors Windows event logs</td>
</tr>
<tr>
<td>SEN_MS_WINPERF</td>
<td>Monitors Windows performance counters</td>
</tr>
<tr>
<td>SEN_MS_WINSERVICE</td>
<td>Monitors Windows services</td>
</tr>
<tr>
<td>SEN_MS_WMIQUERY</td>
<td>Performs WMI queries</td>
</tr>
</tbody>
</table>
## SEN_MS_COMMANDLINE

### Icon

![Image](image.png)

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExecutionTime*</td>
<td>Time taken by the command to run.</td>
<td>Seconds</td>
<td>Warnig ≥ 30</td>
<td>Resp onse Time</td>
</tr>
<tr>
<td>ExitCode</td>
<td>Exit code returned by the executed command.</td>
<td>None</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>ExitStatus</td>
<td>Status of the command execution. Can depend on the user-defined exit code.</td>
<td>[0 = Successful ; 1 = Failed]</td>
<td>Alarm = 1</td>
<td>Availability</td>
</tr>
<tr>
<td>Result</td>
<td>Displays the return output of the command.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td>Status*</td>
<td>Status of the execution.</td>
<td>[0 = OK ; 1 = Suspicious ; 2 = Failed]</td>
<td>Warnin g = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
</tbody>
</table>

For detailed information about Baselining and KPI, see Managing Baselines and Key Performance Indicators.

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Command Line instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Command Line instance.</td>
</tr>
<tr>
<td>Command Line</td>
<td>Command line provided to the OS for execution.</td>
</tr>
</tbody>
</table>
### Application Classes and Parameters

**Monitoring Studio KM for PATROL 9.4.01**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Line Type</td>
<td>Indicates whether the command line is performed locally or remotely.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Maximum time allowed for the Command Line execution.</td>
</tr>
<tr>
<td>Username</td>
<td>Username configured for the Command Line execution.</td>
</tr>
</tbody>
</table>

### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search instance for the Command Line.</td>
</tr>
<tr>
<td>New &gt; Text Pre-Processing</td>
<td>Creates a new Text Pre-Processing instance for the Command Line.</td>
</tr>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Command Line instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval for this Command Line execution.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the Command Line execution.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Command Line execution.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Command Line instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Command Line instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Command Line instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Command Line instance and all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename...</td>
<td>Allows renaming the label (Display Name) of the Command Line instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts...</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the Command Line instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the Command Line instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Re-executes the command line and refreshes all parameters. All related Monitor instances and parameters will be refreshed as well.</td>
</tr>
</tbody>
</table>
**SEN_MS_DBQUERY**

**Icon**

![Icon](image)

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Execution Time</strong></td>
<td>Time taken by the query to run. Value set by dbQueryColl every 2 minutes.</td>
<td>Seconds</td>
<td>Warning ≥ 15</td>
<td>Response Time</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Alarm ≥ 60</td>
<td></td>
</tr>
<tr>
<td><strong>Result</strong></td>
<td>Displays the output of the database query execution. Value set by dbQueryColl every 2 minutes.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td><strong>Status</strong></td>
<td>Indicates whether or not the query was successfully executed. Value set by dbQueryColl every 2 minutes.</td>
<td></td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
<td>Availability</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

For detailed information about Baselining and KPI, see *Managing Baselines and Key Performance Indicators*.

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ID</strong></td>
<td>PATROL internal identifier of the Database Query instance.</td>
</tr>
<tr>
<td><strong>Parent ID</strong></td>
<td>PATROL internal identifier of the parent Database Query instance.</td>
</tr>
<tr>
<td><strong>Database Type</strong></td>
<td>Displays the database type.</td>
</tr>
<tr>
<td><strong>Connection Type</strong></td>
<td>Displays the connection type.</td>
</tr>
<tr>
<td><strong>Database Name</strong></td>
<td>Displays the database name.</td>
</tr>
<tr>
<td><strong>Connect as</strong></td>
<td>Login used to connect to the database server.</td>
</tr>
<tr>
<td><strong>SQL Query</strong></td>
<td>Displays the SQL query that will be executed.</td>
</tr>
</tbody>
</table>

**Menu Commands**

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search instance for the Database Query instance.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>New &gt; Numeric Value Extraction</td>
<td>Creates a new Numeric Value Extraction for the Database Query instance.</td>
</tr>
<tr>
<td>New &gt; Text Pre-Processing</td>
<td>Creates a new Text Pre-Processing instance for the Database Query instance.</td>
</tr>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Database Query instance</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval for the monitoring of the Database Query instance.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the Database Query instance monitoring.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Database Query instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Database Query instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Database Query instance..</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Database Query instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Database Query instance and all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the the Database Query instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of this Database Query instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the the monitoring of this Database Query instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Refreshes all instance parameters of the SEN_MS_DBQUERY application class.</td>
</tr>
</tbody>
</table>
SEN_MS_DYNAMIC

Icon

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExtractedLine</td>
<td>Displays the output of the Dynamic Object instance.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td>Status*</td>
<td>Indicates whether the Dynamic Object instance is missing or not. Default polling interval: 1 minute.</td>
<td>{0 = Instance is present; 1 = Instance is missing}</td>
<td>Alarm = 1</td>
<td>Availability</td>
</tr>
</tbody>
</table>

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Dynamic Object instance.</td>
</tr>
</tbody>
</table>

Menu Commands

None.
SEN_MS_DYNAMIC_CONT

Icon

Parameters
None.

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of Dynamic Object container instance.</td>
</tr>
</tbody>
</table>

Menu Commands
None.
SEN_MS_DYNAMICDISCOVERY

Icon

Parameters
None

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the object.</td>
</tr>
<tr>
<td>Type</td>
<td>Dynamic Discovery type.</td>
</tr>
</tbody>
</table>

Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Allows editing the settings of the Dynamic Object Builder instance.</td>
</tr>
<tr>
<td>Alert actions</td>
<td>Adds specific Alert Actions to the Dynamic Object Builder instance.</td>
</tr>
<tr>
<td>Remove Missing Items</td>
<td>Removes missing items.</td>
</tr>
<tr>
<td>Show/Hide Templates</td>
<td>Displays or hides the Dynamic Object Container and any Dynamic Template associated to the Dynamic Object Builder instance.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Dynamic Object Builder instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Dynamic Object Builder instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Dynamic Object Builder instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Dynamic Object Builder instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Dynamic Object Builder instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the Dynamic Object Builder instance.</td>
</tr>
</tbody>
</table>
**SEN_MS_DYNAMICNUMBER**

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delta</td>
<td>Displays the difference between values collected during two consecutive pollings.</td>
<td>Delta</td>
<td>None</td>
<td>Delta</td>
</tr>
<tr>
<td>DeltaPerSecond</td>
<td>Displays the value corresponding to &quot;Delta&quot; divided by the elapsed time in seconds between the collection times.</td>
<td>Delta/s</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>Value*</td>
<td>Displays the value of the searched Numeric Value (no value will be given if no number is found). Value set by the collector of the parent’s object.</td>
<td>Value</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>ValueFound</td>
<td>States if a numeric value has been found.</td>
<td>{0 = Value found ; 1 = Value not found}</td>
<td>Alarm = 1</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the object.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent object.</td>
</tr>
<tr>
<td>Type</td>
<td>Dynamic Number type.</td>
</tr>
<tr>
<td>Line Mode</td>
<td>Method used to select lines to search for the numeric value.</td>
</tr>
<tr>
<td>Column Mode</td>
<td>Method chosen to search and extract the numeric value in the line.</td>
</tr>
<tr>
<td>Parameter Type</td>
<td>Type of the parameter.</td>
</tr>
<tr>
<td>Line Numbers</td>
<td>Line numbers in which the numeric value will be extracted from.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Regular expression used to select the lines where the number will be searched for.</td>
</tr>
<tr>
<td>Skip Blank Lines</td>
<td>Indicates whether or not blank lines are ignored when searching for the numeric value.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>After/Before</td>
<td>Searches for the numeric value either after or before the specified string.</td>
</tr>
<tr>
<td>Specified String</td>
<td>Searches for the numeric value before or after this specified string.</td>
</tr>
<tr>
<td>Character Offset</td>
<td>Character offset where the numeric value is searched for.</td>
</tr>
<tr>
<td>Field Number</td>
<td>Field number where the numeric value is extracted from.</td>
</tr>
<tr>
<td>Field Separators</td>
<td>Characters that separate the fields in a text line.</td>
</tr>
<tr>
<td>Unique Separator</td>
<td>Considers consecutive separators as a single separator.</td>
</tr>
</tbody>
</table>
**SEN_MS_DYNAMICSTRING**

**Icon**

![Icon](image)

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>MatchingLineCount*</td>
<td>Indicates if a string matching the provided criteria has been found. Value set by the collector of the parent object</td>
<td></td>
<td>Alarm = 1</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the object.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent object.</td>
</tr>
<tr>
<td>Type</td>
<td>Dynamic String type.</td>
</tr>
<tr>
<td>Lines</td>
<td>Lines that are searched.</td>
</tr>
<tr>
<td>Run Alert Actions</td>
<td>When Alert Actions have to be executed.</td>
</tr>
</tbody>
</table>
# SEN_MS_DYNAMICVALUEMAP

## Icon

![Icon](image)

## Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Status of the value mapping result.</td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
<td>1 = Warning 2 = Alert</td>
<td>Availability</td>
</tr>
<tr>
<td>Status Information</td>
<td>Additional information about the value mapping.</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

## Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Dynamic Value Map instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Dynamic Value Map instance.</td>
</tr>
<tr>
<td>Line Mode</td>
<td>Indicates the option selected to define the line(s) from which the searched values are extracted.</td>
</tr>
<tr>
<td>Line Numbers</td>
<td>Indicates the line number(s) from which the value to map is extracted.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Indicates the regular expression that needs to match the lines from which the value to map is extracted.</td>
</tr>
<tr>
<td>Skip Blank Lines</td>
<td>Indicates whether or not empty lines are ignored when searching for a value.</td>
</tr>
<tr>
<td>After/Before</td>
<td>Indicates if the value to map is located after or before the provided string.</td>
</tr>
<tr>
<td>Specified String</td>
<td>Indicates the string after or before which the value to map is extracted.</td>
</tr>
<tr>
<td>Character Offset</td>
<td>Indicates the character offset specifying the position of the value to map.</td>
</tr>
<tr>
<td>Field Number</td>
<td>Indicates the number of the field where the value to map is located.</td>
</tr>
<tr>
<td>Field Separators</td>
<td>Indicates the column or end of field separator specified by the user.</td>
</tr>
<tr>
<td>Unique Separator</td>
<td>Indicates whether or not multiple consecutive separators must be treated as a single one.</td>
</tr>
<tr>
<td>Values Mapped to OK</td>
<td>Indicates the value(s) mapped to the OK status.</td>
</tr>
<tr>
<td>Values Mapped to Suspicious</td>
<td>Indicates the value(s) mapped to the Suspicious status.</td>
</tr>
<tr>
<td>Values Mapped to Failed</td>
<td>Indicates the value(s) mapped to the Failed status.</td>
</tr>
</tbody>
</table>
### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edit</strong></td>
<td>Allows editing the settings for the Dynamic Value Map Monitor instance.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the Dynamic Value Map instance monitoring.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the Dynamic Value Map instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the Dynamic Value Map instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the Dynamic Value Map instance.</td>
</tr>
<tr>
<td><strong>Paste</strong></td>
<td>Pastes a previously copied or cut Dynamic Value Map instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the Dynamic Value Map instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the Dynamic Value Map instance.</td>
</tr>
<tr>
<td><strong>Acknowledge Alerts</strong></td>
<td>Acknowledges all alerts for Dynamic Value Map instance.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the Dynamic Value Map instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the Dynamic Value Map instance as well as all its related Monitor instances and parameters.</td>
</tr>
</tbody>
</table>
### SEN_MS_FILE

**Icon**

![File Icon](image)

#### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exists*</td>
<td>Indicates whether the file exists or not. Value set by fileColl.</td>
<td>{0 = File exists; 1 = File does not exist}</td>
<td>Alarm = 1</td>
<td>Availability</td>
</tr>
<tr>
<td>GrowthPercentage</td>
<td>File growth percentage. Value set by fileColl.</td>
<td>Percentage per minute (%/min)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>GrowthSpeed</td>
<td>File growth speed. Value set by fileColl.</td>
<td>Kilobytes per minute (KB/min)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>LastChanged</td>
<td>Elapsed time since the file was modified. Value set by fileColl.</td>
<td>Minutes</td>
<td>Warning ≥ 7 200 Alarm ≥ 14 400</td>
<td>Statistics</td>
</tr>
<tr>
<td>Size*</td>
<td>File size. Value set by fileColl.</td>
<td>Kilobytes (KB)</td>
<td>Alarm = 0</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

#### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the File instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the File instance's parent.</td>
</tr>
<tr>
<td>File</td>
<td>Configured file path.</td>
</tr>
<tr>
<td>Monitored File</td>
<td>Currently monitored file path.</td>
</tr>
<tr>
<td>File Type</td>
<td>File type (flat/log) as configured in the wizard.</td>
</tr>
</tbody>
</table>

#### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New String Search</td>
<td>Creates a new String Search instance for the File instance.</td>
</tr>
</tbody>
</table>
### Application Classes and Parameters

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>New &gt; Numeric Value Extraction</strong></td>
<td>Creates a new Numeric Value Extraction for the File instance.</td>
</tr>
<tr>
<td><strong>New &gt; Text Pre-Processing</strong></td>
<td>Creates a new Text Pre-Processing instance for the File instance.</td>
</tr>
<tr>
<td><strong>New &gt; Dynamic Object Builder</strong></td>
<td>Creates a new Dynamic Object Builder for the File instance.</td>
</tr>
<tr>
<td><strong>Edit</strong></td>
<td>Allows editing the settings for the File Monitor instance.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Polling Interval</strong></td>
<td>Allows setting the polling interval for the monitoring of the File instance.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the File instance monitoring.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the File instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the File instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the File instance.</td>
</tr>
<tr>
<td><strong>Paste</strong></td>
<td>Pastes a previously copied or cut File instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the File instance and all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the File instance.</td>
</tr>
<tr>
<td><strong>View File Content</strong></td>
<td>Displays the File content.</td>
</tr>
<tr>
<td><strong>Restart Scan</strong></td>
<td>Restart the scanning of the file from the beginning at the next polling cycle, when performing String Searches or Numeric Value Extractions.</td>
</tr>
<tr>
<td><strong>Acknowledge Alerts</strong></td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the File instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the File instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Refreshes all instance parameters of the SEN_MS_FILE application class.</td>
</tr>
</tbody>
</table>
## SEN_MS_FILESYSTEM

### Icon

[Image of icon]

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>AvailableCapacity*</td>
<td>Total capacity not consumed in the file system. Value set by fileSystemColl.</td>
<td>Megabytes (MB)</td>
<td>Alarm ≤ 10</td>
<td>Statistics</td>
</tr>
<tr>
<td>AvailableCapacityPercentage*</td>
<td>Percentage of capacity not consumed in the file system. Value set by fileSystemColl.</td>
<td>Percent (%)</td>
<td>Warning ≤ 10, Alarm ≤ 1</td>
<td>Statistics</td>
</tr>
<tr>
<td>AvailableInodesPercentage</td>
<td>Percentage of available inodes on UNIX and Linux platforms. Value set by fileSystemColl.</td>
<td>Percent (%)</td>
<td>Warning ≤ 10, Alarm ≤ 1</td>
<td>Statistics</td>
</tr>
<tr>
<td>ConsumedCapacityGrowthPercentage</td>
<td>Percentage of the capacity that is actually consumed per hour in the file system. Value set by fileSystemColl.</td>
<td>Percentage per hour (%/h)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>ConsumedCapacityGrowthSpeed</td>
<td>Speed at which the capacity is actually consumed in the file system. Value set by fileSystemColl.</td>
<td>Megabytes per hour (MB/h)</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the object.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent object.</td>
</tr>
<tr>
<td>File System</td>
<td>Drive or path of the monitored file system.</td>
</tr>
</tbody>
</table>

### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the File System instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
</tbody>
</table>
**Menu Command** | **Description**  
--- | ---  
**Polling Interval** | Allows setting the polling interval for the monitoring of the File System instance.  
**Alert Actions** | Adds specific Alert Actions to this File System instance.  
**Properties** | Displays all available information for this File System instance.  
**Cut** | Cuts the File System instance.  
**Copy** | Copies the File System instance.  
**Paste** | Pastes a previously copied or cut File System instance.  
**Delete** | Deletes the File System instance and all its related Monitor instances and parameters.  
**Rename** | Allows renaming the label (Display Name) of the File System instance.  
**Acknowledge Alerts** | Deactivates and reactivates the parameters to acknowledge the alert.  
**Pause Monitoring** | Pauses the monitoring of the File System instance as well as all its related Monitor instances and parameters.  
**Resume Monitoring** | Resumes the monitoring of the File System instance as well as all its related Monitor instances and parameters.  
**Refresh Parameters** | Refreshes all instance parameters of the SEN_MS_FILESystEM application class.
## SEN_MS_FOLDER

### Icon
![Folder Icon](image)

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeletedFileRate</td>
<td>Displays the number of deleted files per minute. Value set by folderColl.</td>
<td>Files/min</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>Exists*</td>
<td>Indicates whether the folder exists or not since the last collect. Value set by folderColl.</td>
<td>{0 = Folder exists ; 1 = Folder does not exist}</td>
<td>Alarm = 1</td>
<td>Availability</td>
</tr>
<tr>
<td>FileCount*</td>
<td>Displays the current number of files in a folder (includes sub-folders, if any, when the option is activated). Value set by folderColl.</td>
<td>File(s)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>GrowthPercentage</td>
<td>Displays the percentage of the folder size growth per minute since the last polling. Value set by folderColl.</td>
<td>Percentag e per minute (%)/min</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>GrowthSpeed</td>
<td>Displays the folder size growth per minute since the last polling. Value set by folderColl.</td>
<td>Kilobytes per minute (KB/min)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>LastModifiedFileElapsedTime</td>
<td>Displays the elapsed time since the last modification of any file in the folder. Value set by folderColl.</td>
<td>Minutes</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>LongestTimeFileRemainsInFolder</td>
<td>Displays the longest time an existing file has been placed in the folder. Value set by folderColl.</td>
<td>Minutes</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>ModifiedFileRate</td>
<td>Displays the rate of modified files per minute. Value set by folderColl.</td>
<td>Files/min</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>
### Application Classes and Parameters

#### Monitoring Studio KM for PATROL 9.4.01

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>NewFileRate</td>
<td>Displays the rate of new files per minute. Value set by folderColl.</td>
<td>Files/min</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>OldestModifiedFileElapsedTime</td>
<td>Displays the elapsed time since the oldest modification of any file in the folder (or sub-folder, when the option is activated). Value set by folderColl.</td>
<td>Minutes</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>Note: User-defined settings could impact the way Monitoring Studio manages alerts and values for this parameter (see Monitoring Folders for more information).</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Size*</td>
<td>Displays the total size of all the files in the folder (include sub-folders if any) in MB. Value set by folderColl.</td>
<td>Megabytes (MB)</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

#### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Folder.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Folder.</td>
</tr>
<tr>
<td>Folder</td>
<td>Path to the monitored folder.</td>
</tr>
<tr>
<td>Include subfolders</td>
<td>Displays whether subfolders are monitored or not.</td>
</tr>
</tbody>
</table>

#### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Folder instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling interval</td>
<td>Allows setting the polling interval for the monitoring of the Folder instance.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the Folder instance.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Folder instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Folder instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Folder instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Folder instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Folder instance.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the Folder instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates andreactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the Folder instance as well as all its related Monitor instances and parameters</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the Folder instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Refreshes all instance parameters of the SEN_MS_FOLDER application class.</td>
</tr>
</tbody>
</table>
SEN_MS_FORMULA

Icon

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Result</td>
<td>Displays the (alphanumeric) return output of the formula. Value set by formulaColl.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td>Value*</td>
<td>Derived (numeric) value from the formula based on the input parameters. Value set by formulaColl.</td>
<td>Value</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the object.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent object.</td>
</tr>
<tr>
<td>Formula</td>
<td>Formula applied to the parameters.</td>
</tr>
<tr>
<td>Variable [A to Z]</td>
<td>Parameter associated to the variable.</td>
</tr>
</tbody>
</table>

Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search instance for the Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>New &gt; Numeric Value Extraction</td>
<td>Creates a new Numeric Value Extraction instance for the Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>New &gt; Text Pre-Processing</td>
<td>Creates a new Text Pre-Processing instance for the Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>New &gt; Dynamic Object Builder</td>
<td>Creates a new Dynamic Object Builder instance for the Multi-Parameter Formula.</td>
</tr>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Polling Intervals</td>
<td>Allows setting the polling interval for the monitoring of the Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the Multi-Parameter Formula instance monitoring.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Multi-Parameter Formula instance and all its dependent Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the Multi-Parameter Formula instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the Multi-Parameter Formula instance as well as all its dependent Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of this Multi-Parameter Formula instance as well as all its dependent Monitor instances and parameters.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Refreshes all instance parameters of the SEN_MS_FORMULA application class.</td>
</tr>
</tbody>
</table>
**SEN_MS_GROUP**

**Icon**

![Icon](image)

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CollectionErrorCount*</td>
<td>Number of collection problems that occurred on the Host and the related Monitors attached to its group. This parameter is cumulative, new errors increase the value of the parameter. The CollectionErrorCount value will be reset after a given time if no new errors are found. The default timeout is 15 minutes but can be configured with the: <code>/SENTRY/STUDIO/&lt;groupId&gt;/collectionErrorCountAutoAcknowledgeTime</code> variable. Value set by collectionErrorColl.</td>
<td>Errors</td>
<td>Alarm ≥ 1</td>
<td>Collection Status</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the object.</td>
</tr>
<tr>
<td>Contact</td>
<td>Name or contact information of the person in charge of the Group.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the Group.</td>
</tr>
<tr>
<td>Constant 1</td>
<td>Name of the first Group Constant.</td>
</tr>
<tr>
<td>Constant 2</td>
<td>Name of the second Group Constant.</td>
</tr>
<tr>
<td>Constant 3</td>
<td>Name of the third Group Constant.</td>
</tr>
<tr>
<td>Constant 4</td>
<td>Name of the fourth Group Constant.</td>
</tr>
<tr>
<td>Constant 5</td>
<td>Name of the fifth Group Constant.</td>
</tr>
</tbody>
</table>
## Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>New &gt; Host</strong></td>
<td>Adds a Host to the Group.</td>
</tr>
<tr>
<td><strong>Edit</strong></td>
<td>Edits the Group monitoring settings.</td>
</tr>
<tr>
<td><strong>Group Constants</strong></td>
<td>Modifies the Group Constants.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Polling Interval</strong></td>
<td>Allows setting the polling interval for the monitoring of the Group instance.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the Group instance monitoring and to its related Monitor instances.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the Group instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the Group instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the Group instance.</td>
</tr>
<tr>
<td><strong>Paste</strong></td>
<td>Pastes a previously copied or cut Group instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the Group instance and all its related Monitor instances.</td>
</tr>
<tr>
<td><strong>Delete all</strong></td>
<td>Deletes all objects related to the Group instance.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the Group instance.</td>
</tr>
<tr>
<td><strong>Reset CollectionErrorCount</strong></td>
<td>Resets the CollectionErrorCount parameter.</td>
</tr>
<tr>
<td><strong>Export Configuration</strong></td>
<td>Exports the Group instance configuration.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the Group instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the Group instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Refreshes all instance parameters of the SEN_MS_GROUP application class.</td>
</tr>
</tbody>
</table>
**SEN_MS_HOST**

**Icon**

![Icon Image](image)

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status*</td>
<td>Status of the Host availability. Value set by the availabilityCheckColl.</td>
<td>{ 0 = OK; 1 = Signature Files Not Present; 2 = Unreachable }</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Host: &lt;groupID&gt;@&lt;hostname&gt;.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent instance.</td>
</tr>
<tr>
<td>Hostname</td>
<td>Hostname or IP address of the Host.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the Host.</td>
</tr>
<tr>
<td>BMC TrueSight Device</td>
<td>FQDN and Token ID of the Host (separated by '/').</td>
</tr>
<tr>
<td>Credentials</td>
<td>Lists the type of credentials provided.</td>
</tr>
<tr>
<td>Signature File 1</td>
<td>First signature file.</td>
</tr>
<tr>
<td>Signature File 2</td>
<td>Second signature file.</td>
</tr>
<tr>
<td>Signature File 3</td>
<td>Third signature file.</td>
</tr>
<tr>
<td>Signature File 4</td>
<td>Fourth signature file.</td>
</tr>
</tbody>
</table>

**Menu Commands**

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; Monitor</td>
<td>Adds a new Monitor.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>New &gt; Monitor Group</td>
<td>Adds a new Monitor Group.</td>
</tr>
<tr>
<td>Edit</td>
<td>Edits the Host settings.</td>
</tr>
<tr>
<td>Credentials &gt; New Credentials</td>
<td>Creates new credentials.</td>
</tr>
<tr>
<td>Credentials &gt; Edit Credentials</td>
<td>Edits existing credentials.</td>
</tr>
<tr>
<td>Credentials &gt; Delete Credentials</td>
<td>Deletes existing credentials.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds Alert Actions that will be used for the Host as well as its related Monitor instances.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Host.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Host instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies Host instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Host instance.</td>
</tr>
<tr>
<td>Clone</td>
<td>Replicates the monitoring settings from a Host to another.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Host instance and all its related instances.</td>
</tr>
<tr>
<td>Delete all</td>
<td>Deletes all instances related to the Host.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the Host as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the Host as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Tools &gt; File Viewer</td>
<td>Starts the file viewer tool.</td>
</tr>
<tr>
<td>Tools &gt; Process Viewer</td>
<td>Starts the process viewer tool.</td>
</tr>
<tr>
<td>Tools &gt; SNMP Browser</td>
<td>Starts the SNMP browser tool.</td>
</tr>
<tr>
<td>Tools &gt; Windows Event Log Reader</td>
<td>Starts the Windows Event log reader tool (On Windows Agents only).</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Refreshes all instance parameters of the SEN_MS_HOST application class.</td>
</tr>
</tbody>
</table>
## SEN_MS_MAIN

### Icon

![Icon](image)

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CollectionErrorCount*</td>
<td>Number of errors that prevent Monitoring Studio from operating properly. This parameter is cumulative, new errors increase the value of the parameter. The CollectionErrorCount value will be reset after a given time if no new errors are found. The default timeout is 135 minutes but can be configured with the following variable: <code>/SENTRY/STUDIO/collectionErrorCountAutoAcknowledgeTime</code>. Value set by discoveryColl.</td>
<td>Errors</td>
<td>Alarm ≥ 1</td>
<td>Statistics</td>
</tr>
<tr>
<td>DebugStatus</td>
<td>Indicates whether the debug mode is enabled or not. Value set by studioColl.</td>
<td>(0 = Off; 1 = On)</td>
<td>None</td>
<td>Availability</td>
</tr>
<tr>
<td>DiscoveryStatus</td>
<td>Indicates if the global discovery is currently running. Value set by discoveryColl.</td>
<td>(0 = Not Running; 1 = Running)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>DiscoveryTime</td>
<td>Time taken to execute the global discovery. Value set by discoveryColl.</td>
<td>Seconds</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>HostCount</td>
<td>Total number of Hosts managed by Monitoring Studio. Note: Identical hosts are counted as if they were unique. Value set by studioColl.</td>
<td>Hosts</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>MonitorCount*</td>
<td>Total number of Monitors managed by Monitoring Studio. Value set by studioColl.</td>
<td>Monitors</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.
Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>Name of the product.</td>
</tr>
<tr>
<td>Version</td>
<td>Current version of the product.</td>
</tr>
<tr>
<td>Release Date</td>
<td>Release date of the current version of the product.</td>
</tr>
<tr>
<td>Copyright</td>
<td>Copyright information.</td>
</tr>
<tr>
<td>Website</td>
<td>Website address.</td>
</tr>
<tr>
<td>Support</td>
<td>Support contact information.</td>
</tr>
</tbody>
</table>

Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Group</td>
<td>Adds a new Group.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Trigger a KM Discovery</td>
<td>Force a full KM discovery.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes the previously copied or cut Monitor/instance under the Monitoring Studio main icon.</td>
</tr>
<tr>
<td>Tools &gt; Real-time SNMP Trap listener</td>
<td>Starts the real-time SNMP trap listener.</td>
</tr>
<tr>
<td>KM Settings &gt; Discovery Interval</td>
<td>Customizes the discovery interval. By default, discovery is performed every hour.</td>
</tr>
<tr>
<td>KM Settings &gt; Threshholds Mechanism Selection</td>
<td>Customizes the thresholds mechanism used by Monitoring Studio.</td>
</tr>
<tr>
<td>KM Settings &gt; Java Settings</td>
<td>Changes the path of the JRE used by Monitoring Studio. By default, it is automatically discovered.</td>
</tr>
<tr>
<td>KM Settings &gt; SMTP Settings</td>
<td>Configures the SMTP server to be used to receive e-mail alert actions.</td>
</tr>
<tr>
<td>KM Settings &gt; Proxy Settings</td>
<td>Configures the proxy server to be used to execute Web requests.</td>
</tr>
<tr>
<td>KM Settings &gt; Force Classic Configuration Mode</td>
<td>Forces the KM to run in &quot;Classic Mode&quot;. All configuration will be performed from a PATROL Console and any policy sent by TrueSight OM will be ignored.</td>
</tr>
<tr>
<td>KM Settings &gt; Debug</td>
<td>Enables or disables the KM debug options.</td>
</tr>
<tr>
<td>Configuration &gt; Show Configuration</td>
<td>Generates a report containing the entire Monitoring Studio configuration.</td>
</tr>
<tr>
<td>Configuration &gt; Backup Configuration</td>
<td>Backs up the entire Monitoring Studio configuration.</td>
</tr>
<tr>
<td>Configuration &gt; Import Configuration</td>
<td>Imports a PATROL Agent configuration.</td>
</tr>
<tr>
<td><strong>Configuration &gt; Import Nagios Configuration</strong></td>
<td>Imports an existing Nagios configuration into Monitoring Studio.</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Reset CollectionErrorCount</strong></td>
<td>Resets the <code>CollectionErrorCount</code> parameter.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Refreshes all instance parameters of the SEN_MS_MAIN application class.</td>
</tr>
<tr>
<td><strong>About...</strong></td>
<td>Shows the version, build number, and other general information about Monitoring Studio.</td>
</tr>
</tbody>
</table>
SEN_MS_MONITORGROUP

Icon

Parameters

None

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the object.</td>
</tr>
</tbody>
</table>

Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; Monitor</td>
<td>Adds a new Monitor.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Monitor Group instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Monitor Group instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Monitor Group instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Monitor Group instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Monitor Group instance and all its dependent instance.</td>
</tr>
<tr>
<td>Delete all</td>
<td>Deletes all instances related to the Monitor Group instance.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the Monitor Group instance.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the Monitor Group instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the Monitor Group instance as well as all its related Monitor instances and parameters.</td>
</tr>
</tbody>
</table>
## SEN_MS_NAGIOSPERF

### Icon

![Nagios Icon](image)

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delta</td>
<td>Displays the difference between values collected during two consecutive pollings. Value set by nagiosPluginColl.</td>
<td>Delta</td>
<td>None</td>
<td>Delta</td>
</tr>
<tr>
<td>DeltaPerSecond</td>
<td>Displays the value corresponding to &quot;Delta&quot; divided by the elapsed time in seconds between the collection times. Value set by nagiosPluginColl.</td>
<td>Delta/seconds (delta/s)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>Percentage*</td>
<td>Displays the percentage of the Value against the maximum, if a maximum value is found in the performance data. Value set by nagiosPluginColl.</td>
<td>Percent (%)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>Present</td>
<td>Monitors whether the performance object is present in the performance data received. Value set by nagiosPluginColl.</td>
<td>{0 = Found; 1 = Not Found}</td>
<td>Alarm = 1</td>
<td>Availability</td>
</tr>
<tr>
<td>Value*</td>
<td>Value interpreted from the performance data. Value set by nagiosPluginColl.</td>
<td>Value</td>
<td>Set dynamically</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Nagios Performance Data instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Nagios Performance Data instance.</td>
</tr>
<tr>
<td>Raw Value</td>
<td>Value reported as collected.</td>
</tr>
<tr>
<td>Converted Value</td>
<td>Value processed according to the selected rescaling option.</td>
</tr>
</tbody>
</table>

### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the Nagios Performance Data instance.</td>
</tr>
<tr>
<td><strong>Acknowledge Alerts...</strong></td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
</tbody>
</table>
## SEN_MS_NAGIOSPLUGIN

### Icon

![Nagios Plugin Icon](image)

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExecutionTime*</td>
<td>Time taken to execute the Nagios plugin. Value set by nagiosPluginColl.</td>
<td>Seconds</td>
<td>Warning ≥ 30</td>
<td>Response Time</td>
</tr>
<tr>
<td>Result</td>
<td>Displays the return output of the Nagios plugin execution. Value set by</td>
<td>--</td>
<td>--</td>
<td></td>
</tr>
<tr>
<td></td>
<td>nagiosPluginColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Status*</td>
<td>Exit code returned by the Nagios plugin. Value set by nagiosPluginColl.</td>
<td>{0 = OK ; 1 = Warning ; 2 = Critical}</td>
<td>1 = Warning</td>
<td>Availability</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2 = Alarm</td>
<td></td>
</tr>
</tbody>
</table>

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

For detailed information about Baselining and KPI, see [Managing Baselines and Key Performance Indicators](#).

### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Nagios plugin instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Nagios plugin instance.</td>
</tr>
<tr>
<td>Nagios Plugin Command</td>
<td>Nagios plugin command line to be executed.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Maximum time allowed for the Nagios plugin execution.</td>
</tr>
<tr>
<td>Username</td>
<td>Username configured for the Nagios plugin execution.</td>
</tr>
</tbody>
</table>

### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search instance for the Nagios plugin execution.</td>
</tr>
<tr>
<td>New &gt; Numeric Value Extraction</td>
<td>Creates a new Numeric Value Extraction instance for the Nagios plugin execution.</td>
</tr>
<tr>
<td>New &gt; Text Pre-Processing</td>
<td>Creates a new Text Pre-Processing instance for the Nagios plugin execution.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>New &gt; Dynamic Object Builder</td>
<td>Creates a new Dynamic Object Builder for the Nagios plugin execution.</td>
</tr>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Nagios plugin instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval for the Nagios plugin execution.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the Nagios plugin execution.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the command line execution.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Nagios plugin instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Nagios plugin instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Nagios Plugin instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Nagios plugin instance and all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename...</td>
<td>Allows renaming the label (Display Name) of the Nagios plugin instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts...</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Resumes the monitoring of the Nagios plugin instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the Nagios plugin instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Re-executes the Nagios plugin and refreshes all parameters. All related Monitor instances and parameters will be refreshed as well.</td>
</tr>
</tbody>
</table>
**SEN_MS_NUMBER**

**Icon**

![Icon](image)

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value*</td>
<td>Value returned by the Numeric Value extraction (no value will be reported if no number is found). Value set by the collector of the parent's object.</td>
<td>Value</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>ValueFound</td>
<td>States if a numeric value has been found.</td>
<td>{0 = Value found ; 1 = Value not found}</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Numeric Value extraction instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Numeric Value extraction instance.</td>
</tr>
<tr>
<td>Parent Type</td>
<td>Type of the parent of the Numeric Value Extraction instance (File, Command Line, etc.).</td>
</tr>
<tr>
<td>Line Mode</td>
<td>Method used to select lines to search for the numeric value.</td>
</tr>
<tr>
<td>Column Mode</td>
<td>Method chosen to search and extract the numeric value in the line.</td>
</tr>
<tr>
<td>Parameter Type</td>
<td>Type of the parameter.</td>
</tr>
<tr>
<td>Line Numbers</td>
<td>Line numbers in which the numeric value will be extracted from.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Regular expression used to select the lines where the numeric value will be searched.</td>
</tr>
<tr>
<td>Skip Blank Lines</td>
<td>Indicates whether or not blank lines are skipped when searching for the numeric value.</td>
</tr>
<tr>
<td>After/Before</td>
<td>Searches for the numeric value either after or before the specified string.</td>
</tr>
<tr>
<td>Specified String</td>
<td>Searches for the numeric value before or after this specified string.</td>
</tr>
<tr>
<td>Character Offset</td>
<td>Character offset in which the numeric value is searched.</td>
</tr>
<tr>
<td>Field Number</td>
<td>Number of the field in which the numeric value will be searched.</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>Field Separators</td>
<td>Characters that separate the fields in a text line.</td>
</tr>
<tr>
<td>Unique Separator</td>
<td>Considers consecutive separators as a unique separator.</td>
</tr>
</tbody>
</table>

### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edit</strong></td>
<td>Allows editing the settings for the Numeric Value Extraction Monitor instance.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the Numeric Value Extraction instance.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for this Numeric Value Extraction instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the Numeric Value Extraction instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the Numeric Value Extraction instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the Numeric Value Extraction object.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the Numeric Value Extraction instance.</td>
</tr>
<tr>
<td><strong>Acknowledge Alerts</strong></td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the Numeric Value Extraction instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the Numeric Value Extraction instance as well as all its related Monitor instances and parameters.</td>
</tr>
</tbody>
</table>
## SEN_MS_PROCESS

### Icon

![Image](image.png)

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildCount</td>
<td>Displays the number of child processes that match the user-defined criteria.</td>
<td>Processes</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td></td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Count*</td>
<td>Displays the number of processes that match the user-defined criteria.</td>
<td>Processes</td>
<td>Alarm = 0</td>
<td>Statistics</td>
</tr>
<tr>
<td></td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HandleCount</td>
<td>Displays the number of handles opened by the matching process(es).</td>
<td>Handles</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>(Windows only)</td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MatchingProcesses</td>
<td>Displays a list of all processes counted in the Count parameter according to the user-defined criteria.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PageFaultsPerSec</td>
<td>Displays the number of page faults per second caused by the matching process(es).</td>
<td>Faults/s</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>(Windows only)</td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PageFileBytes</td>
<td>Displays the page file used by the matching process(es).</td>
<td>Megabytes (MB)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>(Windows only)</td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PrivateBytes</td>
<td>Displays the amount of memory that has been allocated to the process and that cannot be shared with others.</td>
<td>Megabytes (MB)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>(Windows only)</td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ProcessorTime*</td>
<td>Displays the percentage of processor time used by the matching process(es).</td>
<td>Percentage (%)</td>
<td>Warning ≥ 100</td>
<td>Statistics</td>
</tr>
<tr>
<td></td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Status*</td>
<td>Status of the process monitoring execution.</td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
<tr>
<td>(UNIX/Linux only)</td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ThreadCount</td>
<td>Displays the number of threads of the matching process(es).</td>
<td>Threads</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>(Windows only)</td>
<td>Value set by processColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Application Classes and Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>VirtualBytes</td>
<td>Displays the amount of virtual used by the matching process(es). Value set by processColl.</td>
<td>Megabytes (MB)</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>WorkingSet (Windows only)</td>
<td>Displays the working set size of the matching process(es). Value set by processColl.</td>
<td>Megabytes (MB)</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

For detailed information about Baselining and KPI, see *Managing Baselines and Key Performance Indicators*.

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Process instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the Process instance's parent.</td>
</tr>
<tr>
<td>Process IDs</td>
<td>Monitored process IDs (PIDs).</td>
</tr>
<tr>
<td>Child Process IDs</td>
<td>Detected child process IDs (PIDs).</td>
</tr>
</tbody>
</table>

### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Process Monitor instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval for the monitoring of the Process instance.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the Process instance monitoring.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for this Process monitoring.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Process instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Process instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Process instance and all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the Process instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the Process instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the Process instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Refreshes all instance parameters of the SEN_MS_PROCESS class.</td>
</tr>
</tbody>
</table>
**SENG_MS_PSLCOMMAND**

**Icon**

![Icon](image)

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Condition(s)</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExecutionTime*</td>
<td>Time taken by the PSL Command to be executed. Value set by pslCommandColl.</td>
<td>Seconds</td>
<td>Warning ≥ 30</td>
<td>Response Time</td>
</tr>
<tr>
<td>Result</td>
<td>Displays the return output of the PSL command. Value set by pslCommandColl.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td>Status*</td>
<td>Status of the execution. Value set by pslCommandColl.</td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the PSL Command instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the PSL Command instance's parent.</td>
</tr>
<tr>
<td>PSL Command</td>
<td>PSL command executed by the PATROL Agent.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Maximum time in seconds allowed for the PSL command execution.</td>
</tr>
</tbody>
</table>

**Menu Commands**

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search from the PSL Command instance.</td>
</tr>
<tr>
<td>New &gt; Numeric Value Extraction</td>
<td>Creates a new Numeric Value Extraction from the result of the PSL Command instance.</td>
</tr>
<tr>
<td>New &gt; Text Pre-Processing</td>
<td>Creates a new Text Pre-Processing instance from the result of the PSL Command instance.</td>
</tr>
<tr>
<td>New &gt; Dynamic Object Builder</td>
<td>Creates a new Dynamic Object Builder from the result of the PSL Command instance.</td>
</tr>
</tbody>
</table>
### Menu Command

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edit</strong></td>
<td>Allows editing the settings for the PSL Command Monitor instance.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Polling Interval</strong></td>
<td>Allows setting the polling interval for the monitoring of the PSL Command instance.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the PSL Command instance monitoring.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the PSL Command instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the PSL Command instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the PSL Command instance.</td>
</tr>
<tr>
<td><strong>Paste</strong></td>
<td>Pastes a previously copied or cut PSL Command instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the PSL Command instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Rename...</strong></td>
<td>Allows renaming the label (Display Name) of the PSL Command instance.</td>
</tr>
<tr>
<td><strong>Acknowledge Alerts...</strong></td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the PSL Command instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the PSL Command instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Re-executes the PSL command and refreshes all parameters. All related Monitor instances and parameters will be refreshed as well.</td>
</tr>
</tbody>
</table>
**SEN_MS_SNMPPolling**

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content</td>
<td>Content of the SNMP polling returned output. Value set by snmpPollingColl. Note: Only applicable to SNMP of String type.</td>
<td>--</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td>ExecutionTime*</td>
<td>Time taken by the SNMP polling to be executed. Value set by snmpPollingColl.</td>
<td>Seconds</td>
<td>Warning ≤ 30</td>
<td>Response Time</td>
</tr>
<tr>
<td>Status</td>
<td>Status of the SNMP polling. Value set by snmpPollingColl.</td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
<tr>
<td>Value*</td>
<td>Value of the SNMP polling returned output. Value set by snmpPollingColl. Note: Only applicable to SNMPs of integer type.</td>
<td>Value</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the SNMP Polling instance.</td>
</tr>
<tr>
<td>Hostname</td>
<td>Hostname where the SNMP agent is running.</td>
</tr>
<tr>
<td>SNMP Version</td>
<td>SNMP version used by the monitored device. Possible values: 1, 2c, or 3.</td>
</tr>
<tr>
<td>Port</td>
<td>Port used to perform SNMP queries.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Number of seconds Monitoring Studio will wait for the completion of the SNMP polling (default: 120 seconds). This timeout must be long enough to complete the polling of an entire SNMP table.</td>
</tr>
<tr>
<td>Community</td>
<td>(SNMP v1 or 2c) Community used to perform SNMP queries.</td>
</tr>
<tr>
<td>Username</td>
<td>(SNMP v3) Username used to perform SNMP queries.</td>
</tr>
<tr>
<td>Authentication Protocol</td>
<td>(SNMP v3) Protocol used to authenticate the SNMP v3 messages. Possible values: None, MD5, or SHA.</td>
</tr>
<tr>
<td>Privacy Protocol</td>
<td>(SNMP v3) Privacy protocol used to encrypt/decrypt SNMP v3 messages.</td>
</tr>
</tbody>
</table>
## Application Classes and Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context Name</td>
<td>Context name accessible to the SNMP entity.</td>
</tr>
<tr>
<td>Polling Type</td>
<td>Polling type.</td>
</tr>
<tr>
<td>OID</td>
<td>OID that is polled.</td>
</tr>
<tr>
<td>OID type</td>
<td>Type of the selected OID value (number or string).</td>
</tr>
</tbody>
</table>

## Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search instance for the SNMP Polling instance.</td>
</tr>
<tr>
<td>New &gt; Numeric Value</td>
<td>Creates a new Numeric Value Extraction for the SNMP Polling Instance.</td>
</tr>
<tr>
<td>New &gt; Dynamic Object Builder</td>
<td>Creates a new Dynamic Object Builder the SNMP Polling instance.</td>
</tr>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the SNMP Polling Monitor instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval for the monitoring of the SNMP Polling instance.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the SNMP Polling instance monitoring.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the SNMP Polling instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the SNMP Polling instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Cuts the SNMP Polling instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut SNMP Polling instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the SNMP Polling instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the SNMP Polling instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the SNMP Polling as well as all its dependent objects.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the SNMP Polling as well as all its dependent objects.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Re-polls the OID and refresh all parameters. All related Monitor instances and parameters will be refreshed as well.</td>
</tr>
</tbody>
</table>
SEN_MS_SNMPTRAP

Icon

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>MatchingTrapCount*</td>
<td>Number of SNMP traps matching the search. Value set by snmpTrapColl.</td>
<td>Traps</td>
<td>Alarm = 1</td>
<td>Statistics</td>
</tr>
<tr>
<td>MatchingTrapRate*</td>
<td>Number of SNMP traps per minute matching the search. Value set by snmpTrapColl.</td>
<td>Traps/min</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the SNMP traps instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the SNMP traps instance.</td>
</tr>
<tr>
<td>Community</td>
<td>Community used to listen to traps.</td>
</tr>
<tr>
<td>Enterprise ID</td>
<td>Enterprise ID of the SNMP agent that raises the traps.</td>
</tr>
<tr>
<td>Trap number</td>
<td>Number of the searched trap.</td>
</tr>
<tr>
<td>OID 1</td>
<td>OID of the first varBind of the searched traps.</td>
</tr>
<tr>
<td>String 1</td>
<td>Searched string in the first varBind.</td>
</tr>
<tr>
<td>OID 2</td>
<td>OID of the second varBind of the searched traps.</td>
</tr>
<tr>
<td>String 2</td>
<td>Searched string in the second varBind.</td>
</tr>
<tr>
<td>Acknowledging Trap Number</td>
<td>Trap number that will acknowledge this trap search.</td>
</tr>
<tr>
<td>Acknowledging OID 1</td>
<td>OID of the first varBind of the trap that will acknowledge this trap search.</td>
</tr>
<tr>
<td>Acknowledging String 1</td>
<td>Searched string in the first varBind of the acknowledging trap.</td>
</tr>
<tr>
<td>Acknowledging OID 2</td>
<td>OID of the second varBind of the trap that will acknowledge this trap search.</td>
</tr>
<tr>
<td>Acknowledging String 2</td>
<td>Searched string in the second varBind of the acknowledging trap.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Timeout for auto-acknowledgment.</td>
</tr>
<tr>
<td>Run Alert Actions</td>
<td>Indicates when Alert Actions should be executed.</td>
</tr>
</tbody>
</table>
## Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edit</strong></td>
<td>Allows editing the settings for the SNMP Trap Monitor instance.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the SNMP Trap instance monitoring.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the SNMP Trap instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the SNMP Trap instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the SNMP Trap instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the SNMP Trap instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the SNMP Trap instance.</td>
</tr>
<tr>
<td><strong>Acknowledge all and reset</strong></td>
<td>Acknowledges all alerts on the SNMP Trap instance and resets the &quot;MatchingTrapCount&quot; parameter to '0'.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the SNMP Trap instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the SNMP Trap instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Refreshes all parameters of the SEN_MS_SNMPTRAP application class.</td>
</tr>
</tbody>
</table>
**SEN_MS_STRING**

**Icon**

![Icon Image](image_url)

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>LastMatchingLines</td>
<td>Lines matching the string search. Value set by the collector of the parent object. Note: The number of lines displayed can be changed to a custom value by adding the variable &quot;SENTRY/STUDIO/LastMatchingLinesNumber&quot; with the proper line number to the PATROL Agent configuration. Default: last 50 matching lines.</td>
<td>n/a</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>MatchingLineCount*</td>
<td>Number of lines matching the string search. Value set by the collector of the parent object.</td>
<td>Lines</td>
<td>Alarm ≥ 1</td>
<td>Statistics</td>
</tr>
<tr>
<td>MatchingLineRate*</td>
<td>Number of lines matching the string search per minute. Value set by the collector of the parent object. The MatchingLineRate parameter is only activated for string searches in log files and in &quot;never-ending&quot; command lines.</td>
<td>Lines/min</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the String Search instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the String Search instance.</td>
</tr>
<tr>
<td>Parent Type</td>
<td>Type of the parent of the String Search instance (File, Command Line, etc.).</td>
</tr>
<tr>
<td>Search</td>
<td>Search mode.</td>
</tr>
<tr>
<td>Lines</td>
<td>Lines that are searched.</td>
</tr>
<tr>
<td>Acknowledging String</td>
<td>String that auto-acknowledges the String Search instance.</td>
</tr>
<tr>
<td>Acknowledge</td>
<td>Indicates whether the auto-acknowledgment is enabled.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Timeout for the auto-acknowledging.</td>
</tr>
<tr>
<td>Run Alert Actions</td>
<td>Indicates when Alert Actions should be executed.</td>
</tr>
</tbody>
</table>

---
## Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the String Search Monitor instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the String Search instance monitoring.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the String Search instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the String Search instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the String Search instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut String Search instance</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the String Search instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the String Search instance.</td>
</tr>
<tr>
<td>Acknowledge all and reset</td>
<td>Acknowledges all alerts for String Search instance. The <em>MatchingLineCount</em> parameter is set to zero.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the String Search instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the String Search instance as well as all its related Monitor instances and parameters.</td>
</tr>
</tbody>
</table>
SEN_MS_TRANSFORM

Icon

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>TransformResult</td>
<td>Lines matching the Text Pre-Processing.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>Parent Type</td>
<td>Type of the parent of the Text Pre-Processing instance (File, Command Line, etc.).</td>
</tr>
</tbody>
</table>

Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search from the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>New &gt; Numeric value extraction</td>
<td>Creates a new Numeric Value Extraction from the result of the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>New &gt; Dynamic Object Builder</td>
<td>Creates a new Dynamic Object Builder instance from the result of the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Text Pre-Processing Monitor instance.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Text Pre-Processing instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Text Pre-Processing instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the Text Pre-Processing instance.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the Text Pre-Processing instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the Text Pre-Processing instance as well as all its related Monitor instances and parameters.</td>
</tr>
</tbody>
</table>
## SEN_MS_VALUEMAP

### Icon
![Icon](image)

### Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Status of the value mapping result.</td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
<td>1 = Warning 2 = Alert</td>
<td>Availability</td>
</tr>
<tr>
<td>Status Information</td>
<td>Additional information about the value mapping.</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

### Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Value Map instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Value Map instance.</td>
</tr>
<tr>
<td>Parent Type</td>
<td>Type of the parent of the Value Map instance (File, Command Line, etc.).</td>
</tr>
<tr>
<td>Line Mode</td>
<td>Indicates the option selected to define the line(s) from which the searched values are extracted.</td>
</tr>
<tr>
<td>Line Numbers</td>
<td>Indicates the line number(s) from which the value to map is extracted.</td>
</tr>
<tr>
<td>Regular Expression</td>
<td>Indicates the regular expression that needs to match the lines from which the value to map is extracted.</td>
</tr>
<tr>
<td>Skip Blank Lines</td>
<td>Indicates whether or not empty lines are ignored when searching for a value.</td>
</tr>
<tr>
<td>After/Before</td>
<td>Indicates if the value to map is located after or before the provided string.</td>
</tr>
<tr>
<td>Specified String</td>
<td>Indicates the string after or before which the value to map is extracted.</td>
</tr>
<tr>
<td>Character Offset</td>
<td>Indicates the character offset specifying the position of the value to map.</td>
</tr>
<tr>
<td>Field Number</td>
<td>Indicates the number of the field where the value to map is located.</td>
</tr>
<tr>
<td>Field Separators</td>
<td>Indicates the column or end of field separator specified by the user.</td>
</tr>
<tr>
<td>Unique Separator</td>
<td>Indicates whether or not multiple consecutive separators must be treated as a single one.</td>
</tr>
<tr>
<td>Values Mapped to OK</td>
<td>Indicates the value(s) mapped to the OK status.</td>
</tr>
<tr>
<td>Values Mapped to Suspicious</td>
<td>Indicates the value(s) mapped to the Suspicious status.</td>
</tr>
<tr>
<td>Values Mapped to Failed</td>
<td>Indicates the value(s) mapped to the Failed status.</td>
</tr>
</tbody>
</table>
### Application Classes and Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown Values are Mapped to</td>
<td>Indicate to which status the unknown values are mapped (OK, Suspicious or Failed).</td>
</tr>
<tr>
<td>Retained Value</td>
<td>Indicates which value is retained when multiple values matching the defined criteria are found.</td>
</tr>
<tr>
<td>Status Information Map</td>
<td>Indicates the user-defined additional information reported by the StatusInformation parameter.</td>
</tr>
</tbody>
</table>

### Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edit</strong></td>
<td>Allows editing the settings for the Value Map Monitor instance.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the Value Map instance monitoring.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the Value Map instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the Value Map instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the Value Map instance.</td>
</tr>
<tr>
<td><strong>Paste</strong></td>
<td>Pastes a previously copied or cut Value Map instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the Value Map instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the Value Map instance.</td>
</tr>
<tr>
<td><strong>Acknowledge Alerts</strong></td>
<td>Acknowledges all alerts for Value Map instance.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the Value Map instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the Value Map instance as well as all its related Monitor instances and parameters.</td>
</tr>
</tbody>
</table>
SEN_MS_WBEMQUERY

Icon

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExecutionTime*</td>
<td>Time taken by the WBEM query to be executed.</td>
<td>Seconds</td>
<td>None</td>
<td>Response Time</td>
</tr>
<tr>
<td></td>
<td>Value set by wbemQueryColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Result</td>
<td>Displays the return output of the WBEM query.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td>Value set by wbemQueryColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Status*</td>
<td>Status of the WBEM query execution.</td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
<tr>
<td></td>
<td>Value set by wbemQueryColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the WBEM Query instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the WBEM Query instance.</td>
</tr>
<tr>
<td>Name Space</td>
<td>Displays the WBEM namespace.</td>
</tr>
<tr>
<td>WBEM Query</td>
<td>Displays the WBEM Query.</td>
</tr>
<tr>
<td>Port</td>
<td>Displays the port number used for the WBEM Query.</td>
</tr>
<tr>
<td>Protocol</td>
<td>Displays the protocol used (HTTP/HTTPS).</td>
</tr>
<tr>
<td>Username</td>
<td>Username for the WBEM Query.</td>
</tr>
</tbody>
</table>

Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search from the WBEM Query instance.</td>
</tr>
<tr>
<td>New &gt; Numeric Value Extraction</td>
<td>Creates a new Numeric Value Extraction from the result of the WBEM Query instance.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>New &gt; Text Pre-Processing</td>
<td>Creates a new Text Pre-Processing instance for the result of the WBEM Query instance.</td>
</tr>
<tr>
<td>New &gt; Dynamic Object Builder</td>
<td>Creates a new Dynamic Object Builder from the result of the WBEM Query instance.</td>
</tr>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the WBEM Query Monitor instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval for the monitoring of this WBEM query execution.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the WBEM Query execution.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the WBEM Query instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the WBEM Query instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the WBEM Query instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut WBEM Query instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the WBEM Query instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the WBEM Query instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the WBEM Query instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the WBEM Query instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Refreshes all instance parameters of the SEN_MS_WBEMQUERY application class.</td>
</tr>
</tbody>
</table>
**SEN_MS_WEBREQUEST**

**Icon**

![HTTP Icon]

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExecutionTime*</td>
<td>Time taken by the Web request to be executed.</td>
<td>Seconds</td>
<td>Warning ≥ 15 Alarm ≥ 30</td>
<td>Response Time</td>
</tr>
<tr>
<td></td>
<td>Value set by webRequestColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTTPStatus*</td>
<td>Web Request status. Value set by webRequestColl.</td>
<td>{0 = OK ; 1 = Degraded ; 2 = Failed}</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
<tr>
<td>HTTPStatusCode</td>
<td>Web response status code. Value set by webRequestColl.</td>
<td>n/a</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>Result</td>
<td>Displays the result of the Web Request execution.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td>Value set by webRequestColl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Status</td>
<td>Status of the connection to the Web server.</td>
<td>{0 = OK ; 1 = Suspicious ; 2 = Failed}</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
</tbody>
</table>

For detailed information about Baselining and KPI, see Managing Baselines and Key Performance Indicators.

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Web Request instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Web Request instance.</td>
</tr>
<tr>
<td>Used HTTP method</td>
<td>HTTP method used by the Web Request.</td>
</tr>
</tbody>
</table>

**Menu Commands**

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search from the Web Request instance.</td>
</tr>
<tr>
<td>New &gt; Numeric Value Extraction</td>
<td>Creates a new Numeric Value Extraction from the result of the Web Request instance.</td>
</tr>
</tbody>
</table>
### Menu Command

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>New &gt; Text Pre-Processing</strong></td>
<td>Creates a new Text Pre-Processing instance for the result of the Web Request instance.</td>
</tr>
<tr>
<td><strong>New &gt; Dynamic Object Builder</strong></td>
<td>Creates a new Dynamic Object Builder from the result of the Web Request instance.</td>
</tr>
<tr>
<td><strong>Edit</strong></td>
<td>Allows editing the settings for the Web Request Monitor instance.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Polling Interval</strong></td>
<td>Allows setting the polling interval for the monitoring of this Web Request instance.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the Web Request instance monitoring.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the Web Request instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the Web Request instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the Web Request instance.</td>
</tr>
<tr>
<td><strong>Paste</strong></td>
<td>Pastes a previously copied or cut Web Request instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the Web Request instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the Web Request instance.</td>
</tr>
<tr>
<td><strong>Acknowledge Alerts</strong></td>
<td>Deactivates andreactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the Web Request instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the Web Request instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Refreshes all instance parameters of the SEN_MS_WEBREQUEST application class.</td>
</tr>
</tbody>
</table>
SEN_MS_WINEVENT

Icon

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>MatchingEventCount*</td>
<td>Number of matching Windows Events. Value set by winEventCol.</td>
<td>Events</td>
<td>Alarm ≥ 1</td>
<td>Statistics</td>
</tr>
<tr>
<td>MatchingEventRate*</td>
<td>Rate at which Windows Events are found. Value set by winEventCol.</td>
<td>Events/m</td>
<td>None</td>
<td>Statistics</td>
</tr>
<tr>
<td>MatchingEvents</td>
<td>Lists all matching events, which are counted in MatchingEventCount.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Windows Events instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Windows Events instance.</td>
</tr>
<tr>
<td>Event Log</td>
<td>Indicates which event log is searched.</td>
</tr>
<tr>
<td>Provider</td>
<td>Indicates the provider selected.</td>
</tr>
<tr>
<td>Event ID</td>
<td>ID of the event to be searched.</td>
</tr>
<tr>
<td>Acknowledging Event ID</td>
<td>Event IDs that will acknowledge the detected events.</td>
</tr>
<tr>
<td>Acknowledge</td>
<td>Indicates what is to be acknowledged: one event or all events.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Time after which a matching Windows Event is acknowledged.</td>
</tr>
<tr>
<td>Run Alert Actions</td>
<td>Indicates when Alert Actions should be executed.</td>
</tr>
</tbody>
</table>

Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Windows Event Monitor instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Polling Interval</strong></td>
<td>Allows setting the polling interval for the monitoring of the Windows Event instance.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the Windows Event instance monitoring.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the Windows Event instance.</td>
</tr>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the Windows Event instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the Windows Event instance.</td>
</tr>
<tr>
<td><strong>Paste</strong></td>
<td>Pastes a previously copied or cut Windows Event instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the Windows Event instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the Windows Event instance.</td>
</tr>
<tr>
<td><strong>Acknowledge All and Reset</strong></td>
<td>Acknowledges all alerts for Windows Event instance. The MatchingLineCount parameter is set to zero.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the Windows Event instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the Windows Event instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Refreshes all instance parameters of the SEN_MS_WINEVENT application class.</td>
</tr>
</tbody>
</table>
SEN_MS_WINPERF

Icon

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value *</td>
<td>Value of the monitored Windows Performance counter. Value set by winPerfColl.</td>
<td>Depends on the counter</td>
<td>None</td>
<td>Statistics</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

Infobox

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Windows Performance instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Windows Performance instance.</td>
</tr>
<tr>
<td>Performance Object</td>
<td>Name of the Windows Performance instance that is read.</td>
</tr>
<tr>
<td>Instance Name</td>
<td>Selected Windows Performance instance.</td>
</tr>
<tr>
<td>Performance Counter</td>
<td>Name of the Windows Performance counter that is read.</td>
</tr>
</tbody>
</table>

Menu Commands

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the Windows Performance Monitor instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval for the monitoring of the Windows Performance instance.</td>
</tr>
<tr>
<td>Alert Actions</td>
<td>Adds specific Alert Actions to the Windows Performance instance monitoring.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the Windows Performance instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the Windows Performance instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the Windows Performance instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut Windows Performance instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the Windows Performance instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the PSL Command instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the Windows Performance instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the Windows Performance instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Refreshes all instance parameters of the SEN_MS_WINPERF application class.</td>
</tr>
</tbody>
</table>
**SEN_MS_WINSERVICE**

**Icon**

![Windows Service Icon](image)

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status*</td>
<td>Status of the Windows Service. Value set by winServiceColl.</td>
<td>{0 = Started; 1 = Intermediate state; 2 = Stopped}</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
<tr>
<td>ServiceDetails</td>
<td>Details about the Windows Service.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

*Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.*

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the Windows Service instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the Windows Service instance.</td>
</tr>
<tr>
<td>Service name</td>
<td>Name of the monitored Windows Service.</td>
</tr>
<tr>
<td>Startup Mode</td>
<td>Indicates the startup mode. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>• Boot: device driver started by the operating system loader</td>
</tr>
<tr>
<td></td>
<td>• System: device driver started by the operating system initialization process</td>
</tr>
<tr>
<td></td>
<td>• Auto: service to be started automatically by the service control manager during system startup</td>
</tr>
<tr>
<td></td>
<td>• Manual: these services do not start unless a user logs on and starts them</td>
</tr>
<tr>
<td></td>
<td>• Disabled: service that cannot be started until its startup mode is changed to either Auto or Manual</td>
</tr>
<tr>
<td>Path</td>
<td>Fully qualified path to the service binary file that implements the Windows Service.</td>
</tr>
</tbody>
</table>

**Menu Commands**

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edit</strong></td>
<td>Allows editing the settings for the Windows Service Monitor instance.</td>
</tr>
<tr>
<td><strong>Thresholds</strong></td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td><strong>Polling Interval</strong></td>
<td>Allows setting the polling interval for the monitoring of this Windows Service instance.</td>
</tr>
<tr>
<td><strong>Alert Actions</strong></td>
<td>Adds specific Alert Actions to the Windows Service instance monitoring.</td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td>Displays all available information for the Windows Service instance.</td>
</tr>
</tbody>
</table>
## Application Classes and Parameters

### Monitoring Studio KM for PATROL 9.4.01

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cut</strong></td>
<td>Cuts the Windows Service instance.</td>
</tr>
<tr>
<td><strong>Copy</strong></td>
<td>Copies the Windows Service instance.</td>
</tr>
<tr>
<td><strong>Delete</strong></td>
<td>Deletes the Windows Service instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Rename</strong></td>
<td>Allows renaming the label (Display Name) of the Windows Service instance.</td>
</tr>
<tr>
<td><strong>Acknowledge Alerts</strong></td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td><strong>Pause Monitoring</strong></td>
<td>Pauses the monitoring of the Windows Service instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Resume Monitoring</strong></td>
<td>Resumes the monitoring of the Windows Service instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td><strong>Refresh Parameters</strong></td>
<td>Refreshes all instance parameters of the SEN_MS_WINSERVICE application class.</td>
</tr>
</tbody>
</table>
**SEN_MS_WMIQUERY**

**Icon**

![WMI Icon]

**Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Alert Conditions</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExecutionTime*</td>
<td>Time taken by the WMI query to be executed. Value set by wmiQueryColl.</td>
<td>Seconds</td>
<td>Warning ≤ 30</td>
<td>Response Time</td>
</tr>
<tr>
<td>Result</td>
<td>Displays the result of the WMI query execution. Value set by wmiQueryColl.</td>
<td>n/a</td>
<td>None</td>
<td>--</td>
</tr>
<tr>
<td>Status*</td>
<td>Status of the WMI query execution. Value set by wmiQueryColl.</td>
<td>0 = OK ; 1 = Suspicious ; 2 = Failed</td>
<td>Warning = 1 Alarm = 2</td>
<td>Availability</td>
</tr>
</tbody>
</table>

For detailed information about Baselining and KPI, see Managing Baselines and Key Performance Indicators.

* Parameters marked with an asterisk are used by default when visualizing the corresponding monitor instance in BMC TrueSight Operations Management.

**Infobox**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>PATROL internal identifier of the WMI Query instance.</td>
</tr>
<tr>
<td>Parent ID</td>
<td>PATROL internal identifier of the parent of the WMI Query instance.</td>
</tr>
<tr>
<td>Namespace</td>
<td>Displays WMI namespace (e.g.: root\cimv2).</td>
</tr>
<tr>
<td>WQL Query</td>
<td>Displays the WQL query to be executed.</td>
</tr>
<tr>
<td>Username</td>
<td>Displays the user name.</td>
</tr>
</tbody>
</table>

**Menu Commands**

<table>
<thead>
<tr>
<th>Menu Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New &gt; String Search</td>
<td>Creates a new String Search from the WMI Query instance.</td>
</tr>
<tr>
<td>New &gt; Numeric Value Extraction</td>
<td>Creates a new Numeric Value Extraction from the result of the WMI Query instance.</td>
</tr>
<tr>
<td>Edit</td>
<td>Allows editing the settings for the WMI Query Monitor instance.</td>
</tr>
<tr>
<td>Thresholds</td>
<td>Sets or edits the thresholds.</td>
</tr>
<tr>
<td>Menu Command</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Polling Interval</td>
<td>Allows setting the polling interval for the monitoring of the WMI Query instance.</td>
</tr>
<tr>
<td>Alert actions</td>
<td>Adds specific Alert Actions to the WMI Query instance monitoring.</td>
</tr>
<tr>
<td>Properties</td>
<td>Displays all available information for the WMI Query instance.</td>
</tr>
<tr>
<td>Cut</td>
<td>Cuts the the WMI Query instance.</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the the WMI Query instance.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes a previously copied or cut WMI Query instance.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the WMI Query instance and its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Rename</td>
<td>Allows renaming the label (Display Name) of the WMI Query instance.</td>
</tr>
<tr>
<td>Acknowledge Alerts</td>
<td>Deactivates and reactivates the parameters to acknowledge the alert.</td>
</tr>
<tr>
<td>Pause Monitoring</td>
<td>Pauses the monitoring of the WMI Query instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Resume Monitoring</td>
<td>Resumes the monitoring of the WMI Query instance as well as all its related Monitor instances and parameters.</td>
</tr>
<tr>
<td>Refresh Parameters</td>
<td>Refreshes all instance parameters of the SEN_MS_WMIQUERY application class.</td>
</tr>
</tbody>
</table>
Managing Baselines and Key Performance Indicators

To detect abnormalities on the monitored environment, BMC TrueSight Operations Management calculates baselines per attribute based on values collected over a specified period of time to determine a normal operating range. When the collected values for these parameters are out of range, an alert is triggered. Some attributes are identified by default as Key Performance Indicators (identified with the 📈 icon) and automatically included in the base lining calculation.

Managing baselines

The baseline is the expected normal operating range for an attribute of a monitor. There are two baselines: Baseline High and Baseline Low. Baseline High represents the point at which 95% of the weighted average of the historical values fall below this value for the selected time period; Baseline Low represents the point at which 90% of the weighted average of historical values for the selected time period fall above this line.

Baselines are generated for KPI attributes that have an active abnormality thresholds.

Managing Key Performance Indicators

Starting from v9.5 of BPPM, attributes that have not been initially designated in the KM as Key Performance Indicators (KPIs) cannot be flagged as KPIs from BPPM/TrueSight. Although enabling baseline is possible through the Options > Administration > Intelligent Event Thresholds feature available in the Infrastructure Management Server operator console, BMC does not recommend doing it.

⚠️ For more information, refer to the BMC TrueSight Operations Management documentation available from docs.bmc.com.
Configuration Variables
Global Configuration Variables

This section lists the configuration variables used by Monitoring Studio KM for PATROL. Configuration variables are stored in the PATROL Agent configuration and can be managed through:

- PATROL Configuration Manager (PCM)
- wpconfig.exe (Windows)
- xpconfig (UNIX/Linux)

Global configuration variables are stored under the /SENTRY/STUDIO/ folder in the configuration tree.

⚠️ Restart the PATROL Agent or force a discovery for changes made to configuration variables to take effect.

The following table sums up the configuration variables used by Monitoring Studio globally, i.e. that apply to all of the monitored hosts. These configuration variables are stored under /SENTRY/STUDIO/ in the PATROL Agent’s configuration.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>addTimestampToLastMatchingLines</td>
<td>When set to '1', adds a timestamp to the matching lines when stored in the PATROL Agent namespace. Default: unset.</td>
</tr>
<tr>
<td>bufferMaxBytes</td>
<td>Maximum number of bytes stored in the PATROL namespace (memory) for one Monitor collection result. If the result is over the number of bytes set, it will be stored in a temporary file instead.</td>
</tr>
<tr>
<td>collectionErrorCountAutoAcknowledgeTime</td>
<td>Number of seconds after which the SEN_MS_MAIN/CollectionErrorCountParameter is reset if no new error is detected. Default: 8100 (2 hours and 15 minutes).</td>
</tr>
<tr>
<td>collectionHubOverrideJavaCommandLine</td>
<td>Command line used by the KM to launch the Java Collection Hub. Default: Not set This variable should only be set if instructed by Sentry Support.</td>
</tr>
<tr>
<td>completeCommandLineOnAIX</td>
<td>When set to '1', the process information is obtained using the standard (system V) “ps” command instead of the custom AIX command.</td>
</tr>
</tbody>
</table>
| defaultExitStatusCodes                | Command line default status and status code, separated by a line. This is used to set the default values in the appropriate fields of the command line wizard. Default: 1
0. |
<p>| dfCommand                             | df command executed on Linux and UNIX systems to retrieve file systems information. |</p>
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dfCommand2</td>
<td>df command executed to retrieve filesystem inode usage information for the given Linux and UNIX systems.</td>
</tr>
<tr>
<td>disableI2D</td>
<td>When set to 1, no MetaTokenID and no MetaFQDN information will be set in the PATROL namespace for any Host.</td>
</tr>
<tr>
<td>disableJRECheck</td>
<td>To disable the validation tests of the JRE used by the KM to run Java code and therefore force the KM to use a non-Sun or non-Oracle JRE.</td>
</tr>
<tr>
<td>disablePsIExecuteBug</td>
<td>When set to ‘1’, disables the workaround in the KM for a bug in the PsIExecute() PSL function. If the KM detects that the version of the PATROL Agent is affected by the PsIExecute() bug, it uses an alternate technique to create asynchronous threads with the event_trigger() function and the RemPsl standard event. The disablePsIExecuteBugWorkaround variable disables this workaround. Default: Not set.</td>
</tr>
<tr>
<td>doNotUsePsIExecuteForChildren</td>
<td>When set to ‘1’, String Searches, Numeric Value Extractions, dynamic objects and text transforms collects are not called through a PSL execute call. This should be used only if the PATROL Agent is affected by the PSL execute bug. Default: Not set (false).</td>
</tr>
<tr>
<td>fileFindCommand</td>
<td>On Linux and UNIX systems, command line to retrieve all files matching the monitored file mask. Default: find %{FILENAME} -prune.</td>
</tr>
<tr>
<td>fileFindCommandOneDay</td>
<td>On Linux and UNIX systems, command line to retrieve the list of files matching the monitored file mask that were created or modified less than 24 hours ago. Default: find %{FILENAME} -prune -mtime -1.</td>
</tr>
<tr>
<td>fileFindCommandSevenDays</td>
<td>On Linux and UNIX systems, command-line to retrieve the list of files matching the monitored file mask that were created or modified less than 7 days ago. Default: find %{FILENAME} -prune -mtime -8.</td>
</tr>
<tr>
<td>folderUnixLsCommand</td>
<td>On Linux and UNIX systems, command line to retrieve all files in the monitored folder. Default: /bin/ls -atp1 %{FOLDERPATH}.</td>
</tr>
<tr>
<td>folderUnixRecursiveLsCommand</td>
<td>On Linux and UNIX systems, command line to retrieve all files in the monitored folder, including sub-folders. Default: /bin/ls -atpR1 %{FOLDERPATH}.</td>
</tr>
<tr>
<td>forceSnmpSerialization</td>
<td>When set to '1', forces the serialization of the SNMP requests. Several SNMP requests may be sent at the same time on multi-processor computers. Some poorly written SNMP agents may not support this. Default: Not set.</td>
</tr>
<tr>
<td>httpRequestConnection</td>
<td>HTTP request connection method: sopen or telnet. Default: auto-detect.</td>
</tr>
<tr>
<td>LastMatchingLinesNumber</td>
<td>Maximum number of matching lines that should be stored in the PATROL Agent namespace. Default: 50.</td>
</tr>
<tr>
<td>maxConcurrentHostDiscoveryThreads</td>
<td>Maximum number of concurrent Host discovery threads. Default: 50.</td>
</tr>
<tr>
<td>maxConcurrentHostSSHConnections</td>
<td>Maximum number of concurrent SSH connections to a Host. Default: 10.</td>
</tr>
</tbody>
</table>
### Application Classes and Parameters

#### Monitoring Studio KM for PATROL 9.4.01

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>maxConcurrentTCPChannels</td>
<td>Maximum number of concurrent socket channels open when performing a Host TCP availability check. Default: 10.</td>
</tr>
<tr>
<td>maxConcurrentWMIQueries</td>
<td>Maximum number of concurrent WMI queries. Default: 10.</td>
</tr>
<tr>
<td>maxFileSizeRead</td>
<td>Maximum numbers of characters that should be read from a LOG file in a single collect to perform String Searches, Numeric Value Extractions, etc. Remaining text will be read at the next collect. Default: 33554432 bytes (32MB).</td>
</tr>
<tr>
<td>maxParameterValueLength</td>
<td>Maximum number of characters a text parameter should contain. Once this limit is reached, the value is truncated. Default: 1048576.</td>
</tr>
<tr>
<td>newerFileFindCommand</td>
<td>On Linux and UNIX systems, command line to retrieve all files matching the monitored file mask that are newer than the currently monitored file. Default: find %{FILENAME} -prune -newer %{MONITOREDFILE}.</td>
</tr>
<tr>
<td>psCommand</td>
<td>ps command executed on Linux and UNIX systems to retrieve process information.</td>
</tr>
<tr>
<td>psCommand2</td>
<td>Additional &quot;ps&quot; command executed on Linux and UNIX systems to retrieve additional process information.</td>
</tr>
<tr>
<td>psCommand3</td>
<td>ps command executed on Linux and UNIX systems to retrieve process performance data.</td>
</tr>
<tr>
<td>restartFromStartUponFileSizeDecrease</td>
<td>When set to ‘1’, LOG files are read from the start if their size decreases, indicating the file was purged. Default: 1.</td>
</tr>
<tr>
<td>temporaryFolder</td>
<td>Path to the folder where temporary files used by the KM are stored. By default, this is set to C:\Windows\Temp on Windows based PATROL Agents and /var/tmp for UNIX/Linux based Agents.</td>
</tr>
<tr>
<td>windowsWebRequestScriptPath</td>
<td>Path to the VB script used to execute web requests on Windows. Default: Not set (PATROL bin folder\SEN_MS_winHTTP.vbs)</td>
</tr>
<tr>
<td>wmiQueryColumnSeparator</td>
<td>When multiple values are returned by a WMI query, separator used to split the values. Default: &quot;.&quot;</td>
</tr>
</tbody>
</table>

### Specific Configuration Variables

The following table sums up the configuration variables used by Monitoring Studio for each monitored system.

### Group Variables

These configuration variables are stored under /SENTRY/STUDIO/<groupId> in the PATROL Agent’s configuration.
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>collectionErrorCountAutoAcknowledgeTime</td>
<td>When set to '1', process information on the given Host is obtained using the normal (system V) &quot;ps&quot; command instead of the custom AIX command</td>
</tr>
</tbody>
</table>
Host Variables

These configuration variables are stored under /SENTRY/STUDIO/<hostID> in the PATROL Agent’s configuration.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>completeCommand</td>
<td>When set to '1', process information on the given Host is obtained using the normal (system V) &quot;ps&quot; command instead of the custom AIX command.</td>
</tr>
<tr>
<td>dfCommand</td>
<td>df command executed to retrieve filesystem information for the given Linux and UNIX systems.</td>
</tr>
<tr>
<td>dfCommand2</td>
<td>df command executed to retrieve filesystem inode usage information for the given Linux and UNIX systems.</td>
</tr>
<tr>
<td>fileFindCommand</td>
<td>On Linux and UNIX systems, command line to retrieve all files matching the monitored file mask. Default: <code>find %{FILENAME} -prune</code>.</td>
</tr>
<tr>
<td>fileFindCommand_OneDay</td>
<td>On Linux and UNIX systems, command line to retrieve the list of files matching the monitored file mask that were created or modified less than 24 hours ago. Default: <code>find %{FILENAME} -prune -mtime -1</code>.</td>
</tr>
<tr>
<td>fileFindCommand_SevenDays</td>
<td>On Linux and UNIX systems, command line to retrieve the list of files matching the monitored file mask that were created or modified less than 7 days ago. Default: <code>find %{FILENAME} -prune -mtime -8</code>.</td>
</tr>
<tr>
<td>maxFileSizeRead</td>
<td>Maximum numbers of characters that should be read from a LOG file in a single collect to perform String Searches, Numeric Value Extractions, etc. Remaining text left will be read at the next collect. Default: 33554432 bytes (32MB).</td>
</tr>
<tr>
<td>psCommand2</td>
<td>Additional ps command executed to retrieve additional process information for the given Linux or UNIX Host.</td>
</tr>
<tr>
<td>psCommand3</td>
<td>ps command executed to retrieve process performance data for the given Linux and UNIX systems.</td>
</tr>
<tr>
<td>psCommand</td>
<td>ps command executed to retrieve process information for the given Linux or UNIX Host.</td>
</tr>
<tr>
<td>restartFromStartUponFileSizeDecrease</td>
<td>When set to '1', LOG files on this Host are read from the start if their size decreases, indicating the file was purged. Default: 1.</td>
</tr>
<tr>
<td>snmpRetryInterval</td>
<td>Specifies the interval (in milliseconds) at which the SNMP client must retry after 1 second, 5 seconds, 10 seconds and 15 seconds intervals (i.e. after a total of 31 seconds with 4 retries a timeout error will be triggered. (Comma-separated list of intervals in milliseconds). Default: 500, 1000, 2000, 5000, 50000, which means retries after 0.5, 1, 2, 5 and 5 seconds for a total of 13.5 seconds in 5 retries, before a timeout error is triggered. Empty = Not set.</td>
</tr>
</tbody>
</table>
About...

About HTTP Authentication

The HTTP authentication is a login/password-based mechanism implemented in the HTTP protocol itself. A Web server that requires an HTTP authentication will display a dialog box in a Web browser.

⚠️ HTTP authentication has nothing to do with an authentication system of a Web page with a form asking for user credentials like any public Web mail service, for example.

There are several HTTP authentication schemes depending on the way the proxy server has been configured. Here is the list of HTTP authentication supported by Monitoring Studio:

- Basic
- Digest
- NTLM (Windows-integrated)
- Negotiate

For Basic HTTP authentication, the password is sent in a Base64-encoded form and is therefore very easily decoded.

⚠️ Please note that Monitoring Studio requires Java 1.8 (or higher) for HTTP authentication to perform properly.

The proxy authentication also supports Basic, Digest, NTLM, and Negotiate authentication schemes. It can be configured for accessing Web sites using the **Proxy Settings** KM Command.
About Internal Identifiers (IDs)

In Monitoring Studio KM for PATROL, all objects are identified with a unique ID. When configuring Monitoring Studio using a third-party tool, it is important to have a clear understanding of the ID formats.

There are four different ID formats which vary according to the type of objects:

- **Group ID**
- **Host ID**
- **Monitor ID**
- **Monitor Children ID**

**Group ID**

The Group ID is the internal identifier as entered when configuring the Group Settings. It is referred to as `<groupID>`

![Group ID - Example](image)
Host ID

The Host ID is referred to as <hostID> and always follows this format: <groupID>@<hostname>
where:

- <groupID> is the Internal ID as entered when configuring the Group Settings
- <hostname> is the Hostname/IP address/FQDN of the Host on which the technology you wish
to monitor is running

Monitor ID

For all Monitors added from the Host using KM Commands > New > Monitor... the Monitor ID
format is as follows: <groupID>@<hostID>:<monitorType>:<monitorID> where:

- <groupID> is the Internal ID as entered when configuring the Group Settings
- <hostID> is the <groupID>@<hostname> (Hostname/IP address/FQDN)
- <monitorType> is the type of Monitor (Command Line, Web Request, Folder, Process, etc.)
- <monitorID> is the internal ID as entered when configuring the Monitor Settings

Example
If you create a File Monitor with ID "myFile" under the Host "myHost" which belongs to Group
"myGroup", the complete ID of the File Monitoring would be: myGroup@myHost:File:myFile
Monitor Children ID

For all Monitors added from another Monitor using **KM Commands > New >** (String Search, Numeric Value Extraction, Text Pre-Processing, and Dynamic Object Builder), the Monitor Children ID format is as follows: `<parentID>;<monitorType>;<monitorID>` where:

- `<parentID>` is the `<groupID>@<hostID>:<monitorType>:<monitorID>`
- `<monitorType>` is the type of Monitor (String Search, Numeric Value Extraction, Text Pre-Processing, and Dynamic Object Builder)
- `<monitorID>` is the internal ID as entered when configuring the Monitor Settings

**Example**

If you create a **String Search** Monitor with ID "myStringSearch" under the File "myFile" which belongs to the Host "myHost" which itself belongs to the Group "myGroup", the complete ID of the String Search Monitoring would be:

`myGroup@myHost:File:myFile:StringSearch:myStringSearch`
Monitor Children ID - Example
About Performance Counters

A Windows performance object is defined by a performance object name and a counter name, as well as anything from none to several instances. Furthermore, each performance counter contains a unique set of counters and instances. Some objects are built into the system, typically corresponding to the major hardware components, but others will only be accessible if the associated software has been installed.

Here are some definitions of the items as described by Microsoft:

- **Performance object**: A logical collection of counters that is associated with a resource or service that can be monitored. The objects that are typically installed on a system are: cache, memory, paging file, physical disk, process, processor, server, system and thread.
- **Counter**: A value corresponding to a particular aspect of the performance defined for the performance object.
- **Instance**: A term used to distinguish between multiple performance objects of the same type on a computer.

About Processes

What’s a process

In practice, a process is basically a binary code being executed by processors. Processes are launched by the operating system (since the operating system controls the execution flow) and have several properties: PID (unique identifier of a process); Name; UserID; Command line that was used to launch the process (arguments passed to the binary); Environment; CPU and memory usage; Other various OS-specific properties.

How to identify a process

When you monitor an application, you typically want to check that the application's processes are running properly. The problem lies in how to identify the processes of this application, how to recognize them amongst all of the running processes.

The only thing that really identifies a process is its PID (Process ID). But since the PID is an integer number randomly set upon the process startup, most often you cannot use it to identify the processes of an application (unless the application gives you its PID in a so-called PID file).

In general, you recognize application processes by their name if this criterion is enough to distinguish them from other processes. If the name of the process is not sufficient, you can identify application processes by parsing the process’s command lines. This is typically useful with scripts and java processes, whose process names are the same: java, CSCRIPT.EXE, etc.
Process name

Under Windows, the name of a process is basically the file name of the binary file which is being executed: Java.exe, IisAdmin.exe. It always includes the " .EXE" extension. Process names can easily be shown in Windows Task Manager.

Under UNIX, the process name could be either the file name of the binary being executed, including the path or not, or something completely different (e.g. Oracle processes).

⚠️ The naming of processes is highly platform dependent. Linux processes are not named in the same way as on HP-UX servers, for example. Under UNIX, process names can be shown by executing the "ps -e -o name" command line.

Process command line

Every process is launched through a command line, which consists of the file path to the binary which has to be executed, and arguments that have to be passed to the binary: <path to the binary file> <argument1> <argument2> etc.

If the directory of the binary file is in the PATH environment variable, the path may not be included in the command line: <binary file name> <argument1> <argument2> etc.

This is the only way to distinguish Java processes and scripts from others, because their process names are all identical (Java.EXE). Unfortunately, in Windows, there is no easy way to see the command lines of the currently running processes. Under UNIX, processes command lines can be shown by executing the "ps -e -o comm" command.

Process user ID

On both Windows and UNIX systems, processes run "as" a user. Depending on this, the process may be allowed to access various system resources (files, network, databases, etc.). In secured environments, most applications processes have to run as a specific user to let them access the application resources. If the processes run as another user, the application is very likely to fail and not run properly. This is why it could be important to check that the processes of the application you want to monitor are running as the appropriate user.

PID file

A classic way for applications to indicate they are running is to write the PID of their process into a given file. In this case we only need to read this file and check whether the PID written in the file corresponds to a running process. Please note that now the PID file is not provided for all the applications and most Windows applications do not provide PIDs.
About Regular Expressions

Regular expressions are used in Monitoring Studio to define strings to be searched for. A regular expression is:

- A string formatted with a specific syntax.
- It is intended to select some lines in a text, which will match the regular expression.

Regular expressions are commonly used in pattern matching, and especially on UNIX systems with the grep, awk and sed commands. You can use regular expressions in Monitoring Studio in order to:

- Find a process
- Search for strings in a file
- Check a web page
- Parse a table in a database

The following table describes the regular expression syntax that is supported in Monitoring Studio.

<table>
<thead>
<tr>
<th>Character</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>. (dot)</td>
<td>Match any single character</td>
</tr>
<tr>
<td></td>
<td>Example: Err. will match Err01, Err02 or ErrAB, etc.</td>
</tr>
<tr>
<td>[xyz]</td>
<td>Match any character in the brackets</td>
</tr>
<tr>
<td></td>
<td>Example: Err[123] will match Err1, Err2 or Err3</td>
</tr>
<tr>
<td></td>
<td>[Ee]rror will match either error or Error</td>
</tr>
<tr>
<td>[^xyz]</td>
<td>Match any character not in the brackets</td>
</tr>
<tr>
<td></td>
<td>Example: Err[^12345] will match Err0, Err6, Err7, etc. but not Err1</td>
</tr>
<tr>
<td>[a-z]</td>
<td>Match any character in the range in the brackets</td>
</tr>
<tr>
<td></td>
<td>Example: Err[0-9] will match Err0, Err1, etc. and Err9</td>
</tr>
<tr>
<td></td>
<td>Err[A-Z][0-9] will match ErrA0, ErrA1, ErrS9, ErrZ0, etc. but not Err1A</td>
</tr>
<tr>
<td></td>
<td>Err[A-Z0-9] will match ErrA0, ErrA1, etc. and Err1A</td>
</tr>
<tr>
<td>[^a-z]</td>
<td>Match any character not in the range in the brackets</td>
</tr>
<tr>
<td></td>
<td>Example: Application[^0-9] will match ApplicationA, ApplicationB, Application! but not Application1</td>
</tr>
<tr>
<td>*</td>
<td>Match zero or more repetitions of the preceding</td>
</tr>
<tr>
<td></td>
<td>Example: Err[0-9A-F]* will match Err, Err0, ErrA, Err11, ErrBF0001, etc.</td>
</tr>
<tr>
<td></td>
<td>Error.*ApplicationABC will match all lines that contains Error and ApplicationABC further</td>
</tr>
<tr>
<td></td>
<td>(Critical Error 0x000295F0 on ApplicationABC)</td>
</tr>
<tr>
<td>+</td>
<td>Match one or more repetitions of the preceding</td>
</tr>
<tr>
<td></td>
<td>Example: Err[0-9A-F]+ will match Err0, ErrA, Err11, ErrBF0001, etc. but not Err</td>
</tr>
<tr>
<td>^</td>
<td>Match the beginning of the line</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><code>^[Err</code></td>
<td>^Err will match all lines that begin with Err</td>
</tr>
<tr>
<td><code>$</code></td>
<td>Match the end of the line</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>[0-9]+ connections$</code></td>
<td>will match all lines that end with xxx connections where xxx is an integer</td>
</tr>
<tr>
<td><code>\&lt;</code></td>
<td>Match the beginning of a word</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>\&lt;set</code> will match any line that contains a word that begins with set. It will not match a line that only contains the word unset</td>
<td></td>
</tr>
<tr>
<td><code>\&gt;</code></td>
<td>Match the end of a word</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>[Aa]pplication\&gt;</code> will match all lines that contain the word Application or application but not ApplicationAA</td>
<td></td>
</tr>
<tr>
<td><code>\(expression\)</code></td>
<td>Defines an expression which has to be processed as a unit regarding the modifier *, + and</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>\(_[a-zA-Z0-9]+\)</code></td>
<td>will match only sequences like _patrol, _patrol_agent, _patrol_console, etc.</td>
</tr>
<tr>
<td><code>exprA|exprB</code></td>
<td>Match either exprA or exprB</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>\(firewall\)|\(antivirus\)</code></td>
<td>will match all lines that contains either the word firewall or the word antivirus</td>
</tr>
<tr>
<td><code>\</code></td>
<td>Avoid the meaning of the following character</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>\.</code></td>
<td>will match the single character dot (.)</td>
</tr>
<tr>
<td>C:\Program Files will match C:\Program Files</td>
<td></td>
</tr>
</tbody>
</table>
About WMI

Definition

Windows Management Instrumentation (WMI) is a set of specifications from Microsoft for consolidating the management of devices and applications in a network from Windows computing systems. WMI is the Microsoft implementation of Web Based Enterprise Management (WBEM), which is built on the Common Information Model (CIM), a computer industry standard for defining device and application characteristics so that system administrators and management programs can control devices and applications from multiple manufacturers or sources in the same way.

What does it do?

WMI provides users with information about the status of local or remote computer systems. It also supports such actions as the configuration of security settings, setting and changing system properties, setting and changing permissions for authorized users and user groups, assigning and changing drive labels, scheduling processes to run at specific times, backing up the object repository, and enabling or disabling error logging. You can use WMI to manage both local and remote computers.

The word "Instrumentation" in WMI refers to the fact that WMI can get information about the internal state of computer systems, much like the dashboard instruments of cars can retrieve and display information about the state of the engine. WMI "instruments" by modeling objects such as disks, processes, or other objects found in Windows systems. These computer system objects are modeled using classes such as Win32_LogicalDisk or Win32_Process; as you might expect, the Win32_LogicalDisk class models the logical disks installed on a computer, and the Win32_Process class models any processes currently running on a computer. Classes are based on the extensible schema called the Common Information Model (CIM). The CIM schema is a public standard of the Distributed Management Task Force (http://www.dmtf.org/). WMI capabilities also include eventing, querying, views, user extensions to the schema, instrumentation, and more.

WMI Concepts

CIM Repository

CIM stands for Common Information Model and the repository is the WMI schema that stores the class definitions that model WMI-managed resources. The repository holds the information required to work with live resources in the computing environment. It does not contain actual data about these resources since this data is dynamically retrieved as required. It is this schema that allows the wide variety of different resources to be uniformly managed.
Namespace

CIM classes are organized into namespaces. Each namespace in the CIM contains a logical group of related classes representing a specific technology or area of management. Anytime a connection is made to WMI, a namespace must be specified. Only the classes contained within this namespace may be accessed by the connection. The most common namespace used for Windows management is root\cimv2. This contains the classes with the Win32_ prefix representing various components of the Windows operating system and hosting computer. Examples include Win32_Process (running processes in Windows), Win32_LogicalDisk (Windows logical disk drives), and Win32_ComputerSystem (the computer hosting Windows).

The namespace also includes the CIM_DataFile class which can be used to monitor files and folders. The following table lists common namespaces.

<table>
<thead>
<tr>
<th>Namespace</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>root\cimv2</td>
<td>Contains the most useful classes including all Win32_ classes</td>
</tr>
<tr>
<td>root\default</td>
<td>Contains registry events</td>
</tr>
</tbody>
</table>

Class

Every resource managed by WMI is defined by a class. A class is a template for each type of resource and defines the properties that will be collected for that resource. Examples of common WMI classes are shown in the table below:

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Win32_Process</td>
<td>Processes running on a Windows computer</td>
</tr>
<tr>
<td>Win32_ComputerSystem</td>
<td>The computer running a Windows operating system</td>
</tr>
<tr>
<td>CIM_DataFile</td>
<td>A file stored on a disk</td>
</tr>
</tbody>
</table>
Instance

An Instance is a unique occurrence of a particular class. For example, each service installed on a Windows computer is an instance of the Win32_Service class. The C: drive is an instance of the Win32_LogicalDrive class.

<table>
<thead>
<tr>
<th>Instance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Winmgmt</td>
</tr>
<tr>
<td>DisplayName</td>
<td>Windows Management Instrumentation</td>
</tr>
<tr>
<td>PathName</td>
<td>C:\WINDOWS\system32\svchost.exe -k netsvcs</td>
</tr>
<tr>
<td>StartMode</td>
<td>Auto</td>
</tr>
<tr>
<td>State</td>
<td>Running</td>
</tr>
</tbody>
</table>

Property

A property is unique piece of information about an instance. All instances of a class will have the same set of properties although the values each instance’s properties may differ. Sample Properties of the Win32_Service class are shown in the table below:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Unique name of the service.</td>
</tr>
<tr>
<td>DisplayName</td>
<td>Displayed name of the service.</td>
</tr>
<tr>
<td>PathName</td>
<td>The command line path that was executed to start the service.</td>
</tr>
<tr>
<td>StartMode</td>
<td>Startup type of the service (Auto, Manual, or Disabled)</td>
</tr>
<tr>
<td>State</td>
<td>Current state of the service (Running, Stopping, or Stopped)</td>
</tr>
</tbody>
</table>
Basic WMI Queries

Queries may be issued against WMI resources using WMI Query Language (WQL). WQL is a subset of SQL designed to retrieve information from WMI. A simple example of a WMI query would be: SELECT * FROM Win32_Process. This retrieves all attributes (the * is used as a wildcard) for all processes currently running on the computer. Win32_Process is the name of the WMI class for Windows processes.

WMI queries of this type are often issued from a script using Windows Script Host or from any application or tool that can access WMI. Queries retrieve specific information from instances of WMI resources or execute methods against instances to perform such actions as stopping services, or starting processes.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Example code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>SELECT *</td>
<td>Specifies what properties are returned. Typically * is used to simply retrieve all.</td>
</tr>
<tr>
<td>FROM</td>
<td>FROM __InstanceCreationEvent</td>
<td>Specifies the event class to query. This will be the extrinsic or intrinsic event class.</td>
</tr>
<tr>
<td>WHERE</td>
<td>WHERE TargetInstance ISA 'Win32_Process' AND TargetInstance.Name = 'notepad.exe'</td>
<td>Filters the results. For intrinsic events, will usually include the ISA keyword to specify the class of the TargetInstance.</td>
</tr>
</tbody>
</table>

In case you need help to build your WMI query, you could download WMI CIM Studio – which is one of the WMI Administrative tools on the Microsoft site.
Index

-A-

About
  Process 390
  WMI 394

Acknowledge
  Alerts 215
  All 216
  CollectionErrorCount 217

Acknowledgment
  Automatic 118, 132, 177
  Configuration 132

Add
  Credentials 291
  addTimestampToLastMatchingLines 380

After/Before
  Character Offset 328
  Field Number 328
  Field Separators 328
  ID 328
  Line Mode 328
  Line Numbers 328
  Parent ID 328
  Regular Expression 328
  Retained Value 328
  Skip Blank Lines 328
  Specified String 328
  Status Information Map 328
  Unique Separator 328
  Unknown Values are Mapped to 328
  Values Mapped to Failed 328
  Values Mapped to OK 328
  Values Mapped to Suspicious 328

Alarm
  Trigger 172
  Alert 22

Actions 260

Alert Actions 22
  Annotate the graph 266
  Capabilities 266
  Customize 266
  Display a pop-up on the consoles 266
  Do not trigger any alert 225

Email 221
  Example 287
  Execute a command 266
  Global 260
  Log to a file 266
  Macros 276
  Send a basic SNMP trap 266
  Send a custom SNMP trap 266
  Send an e-mail 266
  SMTP Server 221
  Specific 260
  Trigger a WARNING 225
  Trigger an ALARM 225
  Trigger an event 266
  Trigger an INFORMATION 225

Alert Thresholds
  Modify 225
  Set 224, 225

Alerts
  Acknowledge 215

Analysis
  Command Line 81
  Database Query 90, 91, 93, 95, 97, 99
  log file 177
  PSL Command 142
  WBEM Query 144
  WMI Query 146

Application Classes
  SEN_MS_COMMANDLINE 318
  SEN_MS_DBQUERY 90, 91, 93, 95, 97, 99, 320
  SEN_MS_DYNAMIC 172, 322
  SEN_MS_DYNAMIC_CONT 323
  SEN_MS_DYNAMICDISCOVERY 324
  SEN_MS_DYNAMICNUMBER 325
  SEN_MS_DYNAMICSTRING 327
  SEN_MS_DYNAMICVALUEMAP 328
  SEN_MS_FILE 101, 330
  SEN_MS_FILESYSTEM 104, 332
  SEN_MS_FOLDER 106, 334
  SEN_MS_FORMULA 109, 337
  SEN_MS_GROUP 339
  SEN_MS_HOST 341
  SEN_MS_MAIN 298, 343
  SEN_MS_MONITORGROUP 346
  SEN_MS_NAGIOSPERF 347
  SEN_MS_NAGIOSPLUGIN 349
  SEN_MS_NUMBER 182, 351
  SEN_MS_PROCESS 67, 298, 353
Application Classes

SEN_MS_PSLCOMMAND     355
SEN_MS_SNMPTRAP     355
SEN_MS_SNMPPOLLING     113, 357
SEN_MS_STRING     177, 361
SEN_MS_TRANSFORM     363
SEN_MS_VALUemap     188, 365
SEN_MS_WEBREQUEST     132, 371
SEN_MS_WINPERF     136, 373
SEN_MS_WINSERVICE     138, 375
SEN_MS_WMQUERY     146, 377

Architecture 12

Authentication

HTTP     385

Automatic

Acknowledgment     118, 132, 177

- B -

Baselines 379

BMC

BMC Installation Utility 26
BMC Performance Manager 49
BMC Performance Manager Monitoring Studio 41
BMC Portal 49
BMC Software Installation utility 26
bufferMaxBytes     380

Builder

Dynamic Object 172

- C -

CA

Mode 251
Cache

Refresh 80, 104, 132, 298

CIM 144, 146

CIM contains 394
CIM Repository 394
CIM stands 394
CIM_DataFile 394

Class

CIM_DataFile 394
Win32_ComputerSystem 394
Win32_Process 394

Classic

Mode 251

Clone

Host 155
collectionErrorCountAutoAcknowledgeTime 380, 382
collectionHubHeapSizeMax 380
collectionHubHeapSizeMin 380
collectionHubOverrideJavaCommandLine 380

Collectors

Common 298

Column Separator

Blank 177, 182, 188
Comma 177, 182, 188
Point 177, 182, 188

Command

Execute 266
External 158

Command Execution

Never ending 81
Command Line 90, 99, 390

Analysis 81
Specify 158

Commands

About 343

Acknowledge Alerts 320, 330, 332, 334,
337, 341, 347, 351, 353, 357, 365, 367, 369,
373, 375, 377

Acknowledge Alerts... 318, 349, 355

Acknowledge All and Reset 359, 361, 371

Alert actions 318, 320, 324, 330, 332, 334,
337, 339, 341, 349, 351, 353, 355, 357, 359,
361, 365, 367, 369, 371, 373, 375, 377

awk 392

Backup Configuration 343

Clone 341

Copy 318, 320, 324, 330, 332, 334, 337,
339, 341, 346, 349, 351, 353, 355, 357, 359,
361, 363, 365, 367, 369, 371, 373, 375, 377

Cut 318, 320, 324, 330, 332, 334, 337,
341, 346, 349, 351, 353, 355, 357, 359, 361,
363, 365, 367, 369, 371, 373, 375, 377

Debug 343

Delete 318, 320, 324, 330, 332, 334, 337,
339, 341, 346, 349, 351, 353, 355, 357, 359,
361, 363, 365, 367, 369, 371, 373, 375, 377

Delete all 339, 341, 346

Discovery Interval 343
Commands
  Dynamic Object Builder  318, 320, 330, 337, 349, 355, 357, 363, 367, 369
  Edit Credentials  341
  Export Configuration  339
  File Viewer  341
  Force Classic Configuration Mode  343
grep  392
  Group Constants  339
  Host  339
  Import Configuration  343
  Import Nagios Configuration  343
  Java Settings  343
  Monitor  341, 346
  Monitor Group  341
  New  320
  New Credentials  341
  New Group  343
  NString Search Template  324
  Numeric Value Extraction  318, 320, 330, 337, 349, 355, 357, 363, 367, 369, 377
  Numeric Value Extraction Template  324
  Polling Interval  318, 320, 330, 332, 334, 337, 339, 341, 346, 349, 351, 355, 357, 359
  Polling Intervals  337
  Process Viewer  341
  Proxy Settings  343
  Real-time SNMP Trap listener  343
  Remove Missing Items  324
  Rename...  318, 349, 355
  Reset CollectionErrorCount  339, 343
  Restart scan  330
  sed  392
  Show Configuration  343
  Show/Hide Templates  324
  SMTP Settings  343
  SNMP Browser  341
  String Search  318, 320, 330, 337, 349, 355, 357, 363, 367, 369, 377
  Text Pre-Processing  318, 320, 330, 337, 349, 355, 367, 369
  Thresholds Mechanism Selection  343
  Trigger a KM Discovery  343
  View file content  330
  Windows Event Log Reader  341

Community  113
  completeCommandLineOnAIX  380, 382

Configuration
  Automatic Acknowledgment  132
  Backup  231
  Export  233
  Import  238
  Report  304
  Variables  380, 382

Configure
  Group  54
  Host  54
  Monitor  54
  Proxy Settings  222
  SMTP Server  221

Connection  99
  Database  91, 93, 95, 97, 99
  Driver class  99
  Information  90
  JDBC URL  99
  Method  90
  Settings  91, 93, 95, 97, 99

Console Administration  251

Console system
  Requirements  25

Content
  Viewer  303

Copy
Dynamic Objects Macro
  Web Request Macros 276
  WMI Macros 276

- E -

Edit
  Command Line 81
  Credentials 291
  Database Query Analysis 90
  Dynamic Object 172
  File System Monitoring 104
  Folder Monitoring 106
  Group 55
  Instance 250
  Monitored Object 250
  Multi-Parameter Formula 109
  Numeric Value Search 182
  Process Monitoring 67
  SNMP polling 113
  String Search 177
  Text Preprocessing object 158
  Value Map 188
  WBEM Query Analysis 144
  Web Request 122
  Windows event monitoring 132
  Windows Performance Monitoring 136
  Windows Service Monitoring 138
  WMI Query Analysis 146

Email
  Alert Actions 221
  SMTP Server 221

Enable
  Debug Mode 307
  Event 22

Example 67, 81, 132, 158, 225
  Alert Actions 287

Execute
  Command 266

Execution
  Period 81
  Timeout 81, 142

Export
  Configuration 231, 233
  Global Settings 231, 233
  Group Configuration 233
  Group Constants 233
  Monitoring Studio Configuration 231

Extract
  Setup Files 27
  Text from HTML 158

Extraction
  Numeric Value 182

- F -

FAQ
  Although SNMP Trap listening seems to work, no trap matches my criteria 309
  Can I monitor the processor time usage made by a Windows service? 310
  How do I know which version of Monitoring Studio I am running? 309
  I'm unable to poll an SNMP agent (getting a Warning) 310
  I'm unable to see the Monitoring Studio icons in PATROL Central 311
  Infinite loop reported in the PATROL Agent log 311
  Is it possible to monitor the CPU/memory usage of a process tree? 312
  Monitoring Studio does not follow an HTTP redirection (Web) 313
  Monitoring Studio fails to authenticate on my Web server 312
  Monitoring Studio is unable to listen to SNMP traps 309
  Monitoring Studio reads my log file entirely 310
  Not able to post a Web form to my Web server 313

File
  flat 101, 310
  Growth 101
  log 101, 310
  Monitor 204
  Monitoring 101
  PID 390
  Presence 101
  Size 101

File Content
  Viewer 303

File System
  Monitoring 104
  fileFindCommand 380, 382
  fileFindCommandOneDay 380, 382
  fileFindCommandSevenDays 380, 382
  Files
Files
  PID 67
  flat
    File 310
    flat File
      Monitoring 101
Folder
  Monitoring 106
Folder Monitoring
    Create 106
    Edit 106
    folderUnixLsCommand 380
    folderUnixRecursiveLsCommand 380
Force
  Classic Mode 251
  forceSnmpSerialization 380
Format Symbols
  %% 285
  %a 285
  %b 285
  %c 285
  %d 285
  %e 285
  %Ec 285
  %Ex 285
  %Ey 285
  %h 285
  %I 285
  %j 285
  %k 285
  %l 285
  %m 285
  %n 285
  %Od 285
  %Oe 285
  %OH 285
  %OI 285
  %Om 285
  %n 285
  %Od 285
  %Oe 285
  %OH 285
  %OI 285
  %Om 285
  %OS 285
  %OU 285
  %Ow 285
  %Oy 285
  %p 285
  %r 285
  %S 285
  %t 285
  %u 285
  %V 285
  %w 285

%x 285
%y 285
%Z 285

Formula
  Multi-Parameter 109

- G -
General Concepts 16, 17, 19, 20, 22
General Macros 276
Getting Started 12
Global
  Backing-up 231
  Exporting 233
  Settings 238
Group
  Constants 55, 257
  Create 55
  Edit 55
Growth
  Size 101

- H -
hideTemplates 382
Host 17, 58, 113
  Clone 155
  Identify 144, 146
Host availability
  Hostname 58
  Ping 58
  SNMP 58
  SSH 58
  TCP Port 58
HTTP
  Authentication 385
HTTP authentication schemes
  Basic 122
  Digest 122
  Negotiate 122
  NTLM 122
  httpRequestConnectionType 380

- I -
Icinga 1 152
Icons
Infobox 309
Search 361
Service name 375
Signature File 1 341
Signature File 2 341
Signature File 3 341
Signature File 4 341
Skip Blank Lines 325, 351, 365
SNMP Version 357
Specified String 325, 351, 365
SQL Query 320
Startup Mode 375
Status Information Map 365
String 1 359
String 2 359
Support 343
Timeout 318, 355, 357, 359, 361, 371
Trap number 359
Type 324, 325, 327
Unique Separator 325, 351, 365
Unknown Values are Mapped to 365
Used HTTP method 369
Username 318, 357, 367, 377
Values Mapped to Failed 365
Values Mapped to OK 365
Values Mapped to Suspicious 365
Variable 337
Version 343
WBEM Query 367
Website 343
WQL Query 377
Installation 29
BMC Software Installation utility 26
Packages 26
Procedure 30
Setup Files 27
Utility 26
Instance
DisplayName 394
Edit 250
Name 394
PathName 394
StartMode 394
State 394
Integration 48
BMC Portal 49
BMC TrueSight Operations Management 50
Overview 49
Internal ID 79
Internal Identifier 386
Issues 49
Overloaded Application 106

-J-
Java 26
Automatic Detection 218
Manual (other) 218
Settings 218
User Selection 218
JDBC
Driver classes 99
URL 99
JRe 26

-K-
Key Performance Indicators 379
KPI 379

-L-
LastMatchingLinesNumber 380
Lines
Numeric Value 182
Select 177
Linux 291
Listening
SNMP Trap 118
Loading
on your Console 52
Location
Numeric Value 182
Value 188
log
File 287, 310
log File
Analysis 177
Monitoring 101

-M-
Macros 81
%{TIME:...} 285
Alert Actions 276
Manage
Manage
  Credentials 19, 291
  Thresholds 224
Managed Systems
  Requirements 25
Managing Credentials 19, 291
maxConcurrentHostDiscoveryThreads 380
maxConcurrentHostSSHConnections 380
maxConcurrentTCPChannels 380
maxConcurrentWMIQueries 380
maxFileSizeRead 380, 382
maxParameterValueLength 380
Measure 106
MIB
  Management Information Base 113
Microsoft SQL Server Database
    Connection 91
    Settings 91
Minimum Cache Refresh Interval
    Set 80, 104, 132
Modify
  Alert Thresholds 225
  Group Constants 257
  Object ID 253, 255
  Object Name 253, 255
Monitor
  Command Lines 65
  Database Queries 65
    File 204
    File Systems 65
    Files 65
    Folders 65
    log File 287
  Multi-Parameter Formula 65
  Performance Counters 65
  Processes 65
  PSL Commands 65
  Settings 79
  SNMP Agents 65
  SNMP Traps 65
  Technology 55
  WBEM Queries 65
  Web Requests 65
  Windows Events 65
  Windows Services 65
  WMI Queries 65
Monitor Group 17
  Create 67
  Monitor Internal Identifier 386
Monitor Settings 79
Monitored Object
  Delete 249
Monitoring 258
    File 101
    File System 104
    Folder 106
    Nagios Plugins 148
    Pause 258
    Process 67, 69, 70, 73, 75, 76
    Suspend 258
    Windows Event 132
    Windows Event log 132
    Windows Performance 136
    Windows Service 138
Monitoring Features
    Command Line Analysis 172
    Database Query Analysis 172
    File Monitoring and Analysis 172
    SNMP Polling 172
    Text Pre-Processing 172
    WBEM Query Analysis 172
    Web Request Analysis 172
    WMI Query Analysis 172
Monitoring Studio 17
    Settings 55, 67, 81, 104, 106, 113, 118, 132, 136, 138, 142, 144, 146, 158, 177, 182, 188
Monitoring Studio settings 79
Monitoring Tools
    Copy 229
    Cut 229
    Paste 229
    Monitors 17, 67
Move
    Monitoring Tools 229
Multi-Parameter
    Formula 109
MySQL Server Database
    Connection 93
    Settings 93

- N -

Nagios
    Configuration 152
    Import 152
    Monitoring 148
<table>
<thead>
<tr>
<th>Nagios</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plugins 148</td>
</tr>
<tr>
<td>Namespace 394</td>
</tr>
<tr>
<td>newerFileFindCommand 380</td>
</tr>
<tr>
<td>Numeric Value 20</td>
</tr>
<tr>
<td>Extraction 182</td>
</tr>
<tr>
<td>Lines 182</td>
</tr>
<tr>
<td>Location 182</td>
</tr>
<tr>
<td>Search 182</td>
</tr>
<tr>
<td>Numeric value extraction 20</td>
</tr>
<tr>
<td>Numeric Value Extraction Template 176</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>O</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Object</td>
</tr>
<tr>
<td>Edit 250</td>
</tr>
<tr>
<td>ODBC 90</td>
</tr>
<tr>
<td>OpenSSH 58, 291</td>
</tr>
<tr>
<td>Operating Systems</td>
</tr>
<tr>
<td>Compatibility 65</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Option</td>
</tr>
<tr>
<td>Rescaling 182</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Oracle Database</td>
</tr>
<tr>
<td>Connection 95</td>
</tr>
<tr>
<td>Settings 95</td>
</tr>
<tr>
<td>OS</td>
</tr>
<tr>
<td>Compatibility 65</td>
</tr>
<tr>
<td>Overview 12</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>P</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Packages</td>
</tr>
<tr>
<td>Download 26</td>
</tr>
<tr>
<td>Parameters 132, 346</td>
</tr>
<tr>
<td>AvailableCapacity 332</td>
</tr>
<tr>
<td>AvailableCapacityPercentage 332</td>
</tr>
<tr>
<td>AvailableInodesPercentage 332</td>
</tr>
<tr>
<td>ChildCount 67, 312, 353</td>
</tr>
<tr>
<td>CollectionErrorCount 339, 343</td>
</tr>
<tr>
<td>ConnectionStatus 90</td>
</tr>
<tr>
<td>ConsumedCapacityGrowthPercentage 332</td>
</tr>
<tr>
<td>ConsumedCapacityGrowthSpeed 332</td>
</tr>
<tr>
<td>Content 113, 357</td>
</tr>
<tr>
<td>Count 67, 353</td>
</tr>
<tr>
<td>DebugStatus 343</td>
</tr>
<tr>
<td>DeletedFileCount 106</td>
</tr>
<tr>
<td>DeletedFileRate 334</td>
</tr>
<tr>
<td>Delta 325, 347</td>
</tr>
<tr>
<td>DeltaPerSecond 325, 347</td>
</tr>
<tr>
<td>DiscoveryStatus 343</td>
</tr>
<tr>
<td>DiscoveryTime 343</td>
</tr>
<tr>
<td>ExecutionTime 113, 318, 320, 349, 355, 357, 367, 369, 377</td>
</tr>
<tr>
<td>Exists 330, 334</td>
</tr>
<tr>
<td>ExitCode 318</td>
</tr>
<tr>
<td>ExitStatus 318</td>
</tr>
<tr>
<td>ExtractedLine 322</td>
</tr>
<tr>
<td>FileCount 106, 334</td>
</tr>
<tr>
<td>FolderSize 106</td>
</tr>
<tr>
<td>FreeMegabytes 104</td>
</tr>
<tr>
<td>FreeSpacePercent 104</td>
</tr>
<tr>
<td>GrowthPercentage 106, 330, 334</td>
</tr>
<tr>
<td>GrowthSpeed 106, 330, 334</td>
</tr>
<tr>
<td>HandleCount 67, 353</td>
</tr>
<tr>
<td>HostCount 343</td>
</tr>
<tr>
<td>HTTPStatus 369</td>
</tr>
<tr>
<td>HTTPStatusCode 369</td>
</tr>
<tr>
<td>LastChanged 330</td>
</tr>
<tr>
<td>LastMatchingLines 361</td>
</tr>
<tr>
<td>LastModifiedFileElapsedTime 106, 334</td>
</tr>
<tr>
<td>LongestTimeFileRemainsInFolder 106, 334</td>
</tr>
<tr>
<td>MatchingEventCount 132, 371</td>
</tr>
<tr>
<td>MatchingEventRate 132, 371</td>
</tr>
<tr>
<td>MatchingEvents 371</td>
</tr>
<tr>
<td>MatchingLineCount 177, 287, 327, 361</td>
</tr>
<tr>
<td>MatchingLineRate 361</td>
</tr>
<tr>
<td>MatchingProcesses 353</td>
</tr>
<tr>
<td>MatchingTrapCount 118, 359</td>
</tr>
<tr>
<td>MatchingTrapRate 359</td>
</tr>
<tr>
<td>ModifiedFileCount 106</td>
</tr>
<tr>
<td>ModifiedFileRate 334</td>
</tr>
<tr>
<td>MonitorCount 343</td>
</tr>
<tr>
<td>NewFileCount 106</td>
</tr>
<tr>
<td>NewFileRate 334</td>
</tr>
<tr>
<td>OldestModifiedFileElapsedTime 106, 334</td>
</tr>
<tr>
<td>PageFaultsPerSec 67, 353</td>
</tr>
<tr>
<td>PageFileBytes 67, 353</td>
</tr>
<tr>
<td>Percentage 347</td>
</tr>
<tr>
<td>Present 347</td>
</tr>
<tr>
<td>PrivateBytes 67, 353</td>
</tr>
<tr>
<td>ProcessorTime 67, 353</td>
</tr>
<tr>
<td>Refresh 259</td>
</tr>
<tr>
<td>Result 144, 146, 318, 320, 337, 349, 355, 367, 369, 377</td>
</tr>
<tr>
<td>Select 67, 182</td>
</tr>
<tr>
<td>ServiceDetails 375</td>
</tr>
<tr>
<td>Size 330, 334</td>
</tr>
</tbody>
</table>
Parameters 132, 346
  Status 113, 144, 172, 318, 320, 322, 328, 341, 349, 355, 357, 365, 367, 369, 375, 377
  StatusInformation 328, 365
  ThreadCount 67, 353
  TransformResult 158, 204, 363
  UsedSpaceGrowthPercentage 104
  UsedSpaceGrowthSpeed 104
  Value 113, 325, 337, 347, 351, 357, 373
  ValueFound 325, 351
  VirtualBytes 67, 353
  WorkingSet 67, 353

Parsing
  XML LOG File 204

PassPhrase 58, 291

Paste
  Monitoring Tools 229

PATROL
  BMC Impact Manager 266
  BMC Software PATROL Console 25
  Console Server 311
  Event 266
  PATROL administrator 172
  PATROL Agent 30, 311
  PATROL Agents 311
  PATROL Central 30, 311
  PATROL Central Operator 25
  PATROL Configuration Manager 224
  PATROL Console 30
  PATROL Console Servers 30
  PATROL Console Web-edition 25
  PATROL Consoles 266
  PATROL for Event Management 224

PATROL event
  Trigger 287

Pause
  Monitoring 258

Period
  Execution 81

Pick
  Process 67

PID
  File 390
  Files 67

Polling Interval
  Set 298

Port 113

PostgreSQL Database

Connection 97
Settings 97

Pre-processing 204
  Convert JSON to CSV 158
  Convert JSON to Flat Map 158
  Convert multi-line records to single records 158
  Convert XML to CSV 158
  Extract Text from HTML 158
  Text processing through an external command 158

Presence
  File 101
  Private Key 58
  Private Key File 291

Process
  About 390
  Command Line 390
  Criteria 67
  Identify 390
  Monitoring 67, 69, 70, 73, 75, 76
  Name 390
  Parameters 76
  Pick 67
  Status interpretation 76
  User ID 390

Process monitoring method
  List 70
  PID file path 75
  Search criteria 73

Process Viewer
  Tool 300

Product Architecture 17
Product hierarchy 17

Properties
  Command line 67, 390
  Name 67, 390
  PID 67, 390
  Tool 304
  User ID 67, 390

Property
  DisplayName 394
  Name 394
  PathName 394
  StartMode 394
  State 394

Proxy
  Server 122
  Settings 122
Size
  File  101
SMTP Server
  Configure  221
SNMP
  Agent  113
  Polling  113
  SNMP Trap Listener  301
  Table OID  113
  Trap  118
SNMP Browser
  Tool  305
SNMP Browser tool  310
SNMP polling
  Create  113
  Edit  113
SNMP Trap
  Listening  118
SNMP Trap Listener
  Tool  301
SNMP Trap Listener tool  309
snmpRetryIntervals  382
Specific Credentials  19, 291
Specify
  Command Line  158
SSH  291
Streams
  stderr  81
  stdout  81
String  20
  Search  177
String Search  20
  Create  177
  Edit  177
String Search Template  176
Suspend
  Monitoring  258
System Credentials  19, 58
Systems
  Linux  25, 26, 27
  UNIX  25, 26, 27, 30, 390, 392
  Windows  25, 26, 27, 30, 390
  temporaryFolder  380
Test  106
Text
  Pre-Processing  158
Text Preprocessing object
    Create  158
    Edit  158
Threshold  22
Thresholds
  Advanced  225
  Automatic  224
  Custom  67, 79, 81, 90, 104, 106, 109, 113, 118, 122, 132, 136, 138, 144, 146, 177, 182, 188
  Event Management  224
  Manage  224
  Mechanism mode  224
  Simple  225
  Standard  188
  Type  225
  Update  49
Timeout
  Execution  81, 142
Tool
  Process Viewer  300
  Properties  304
  SNMP Browser  305
  SNMP Trap Listener  301
  Windows Event log Reader  302
Traps
  Incoming  309
Trigger
  Alarm  172
  Event  287
Troubleshooting  307
  Although SNMP Trap listening seems to work, no trap matches my criteria  309
  Can I monitor the processor time usage made by a Windows service?  310
  How do I know which version of Monitoring Studio I am running?  309
  I'm unable to poll an SNMP agent (getting a Warning)  310
  I'm unable to see the Monitoring Studio icons in PATROL Central  311
  Infinite loop reported in the PATROL Agent log  311
Troubleshooting 307

Is it possible to monitor the CPU/memory usage of a process tree? 312
Monitoring Studio does not follow an HTTP redirection (Web) 313
Monitoring Studio fails to authenticate on my Web server 312
Monitoring Studio is unable to listen to SNMP traps 309
Monitoring Studio reads my log file entirely 310
Not able to post a Web form to my Web server 313

Web request
Create 122
DELETE 122
Edit 122
GET 122
POST 122
PUT 122

Windows Event
Monitor 132
Monitoring 132
Types 132

Windows Event log
Application 132
Monitor 132
Security 132
System 132

Windows Event log Reader
Tool 302

Windows event monitoring
Create 132
Edit 132

Windows Performance
Monitoring 136

Windows Service
Monitoring 138

Windows Service Monitoring
Create 138
Edit 138

windowsWebRequestScriptPath 380

WMI
About 394
Query 146

WMI Query
FROM 394
SELECT 394
WHERE 394

WMI Query Analysis
Create 146
Edit 146

wmiQueryColumnSeparator 380

Workaround 49

- X -
XML LOG File
XML LOG File
Parsing     204
About Sentry Software™
Sentry Software, a strategic Technology Alliance Partner of BMC Software, provides comprehensive multi-platform monitoring solutions that enable management of the hardware and software aspects of all servers and SANs and covering up to 100% of custom applications within the BMC TrueSight environment. Sentry Software also develops adapters for BMC Atrium Orchestrator that enables IT administrators to automate the execution of common requests and tasks that occur in the daily course of IT operations. Combined with BMC’s servers and network automation tools, the adapters allow IT administrators to implement provisioning and decommissioning workflows that cover all layers of their IT infrastructure. Finally, Sentry Software designs connectors that bring storage capacity metrics into BMC TrueSight Capacity Optimization to ensure IT administrators that their storage infrastructure is properly sized for their current and future needs.

About BMC Software™
BMC Software helps leading companies around the world put technology at the forefront of business transformation, improving the delivery and consumption of digital services. From mainframe to cloud to mobile, BMC delivers innovative IT management solutions that have enabled more than 20,000 customers to leverage complex technology into extraordinary business performance—increasing their agility and exceeding anything they previously thought possible. For more information about BMC Software, visit www.bmc.com.