

Installation and Usage Guide

The Backbone End-User Performance

Product information	
Name	The Backbone End-User Performance Management Pack
Version	1.3.0.0
Release date	July 2017

Document information	
Name	Installation and Usage Guide for The Backbone End-User Performance Management Pack
Version	1.3.0.0
Release date	July 2017

Owner	
Name	The Backbone B.V.
Address	J.M. Keynesstraat 351 7559 SV Hengelo The Netherlands
Telephone	+31 74 2490446
E-mail	support@thebackbone.nl
Website	http://www.thebackbone.nl/

Table of Contents

1	Introduction.....	6
2	Architecture	7
3	Product requirements	8
3.1	SCOM Management Group.....	8
3.2	EUP Collector.....	8
3.3	Watcher Node	8
4	Installation	10
4.1	Preparation.....	10
4.2	Management Pack import	10
5	Upgrade.....	12
5.1	Preparation.....	12
5.2	Management Pack upgrade	12
5.3	Post upgrade steps	13
6	Configuration	14
6.1	EUP Collector discovery.....	14
6.2	EUP Configuration file.....	21
6.2.1	Simulation instance	21
6.2.2	Watchers	22
6.2.3	Steps.....	22
6.3	Custom configuration scenario's	23
6.3.1	Specifying instance and step thresholds	23
6.3.2	Increasing the EUP Collector interval.....	27
6.3.3	Changing the EUP Collector Path	30
6.3.4	Changing the accepted missing data files	34
7	Simulation data file convention and syntax.....	38
7.1	Filename convention	38
7.2	File syntax.....	38
8	Management Pack explanation	40
8.1	Health Rollup.....	40
8.2	Relationships.....	41
8.3	Views	41
8.4	Reports	42
Appendix 1	Example EUP Configuration file	44
Appendix 2	Example End-User Performance - internet availability and performance	45

Document convention

Font	Definition	Example
<i>Italic</i>	Emphasize Name of a Window File and directory names Process names Values	Action below <i>have to be</i> carried out The <i>configuration window</i> appears <i>Mappings.xml</i> Restart <i>HealthService.exe</i> Set value to <i>True</i>
Bold	Menu items and buttons, dialog screens, menu and icon names	In the menu, choose File , followed by Open
Courier New (10pt)	Content of files. Commands	Use the <code>net start</code> commando

Revision history

Revision	Changes
1.3.0.0	<ul style="list-style-type: none"> • Added: Location property to OMS Log Analytics events • Added: Knowledge base sections for all management pack elements • Added: Missing display strings • Changed: Alert view default sort on CreatedTime • Changed: All monitor definitions are public now • Changed: Collector Data Backlog interval from 300 seconds to 900 seconds • Changed: State view 'End-User Performance Instances' renamed to 'End-User Performance Simulations'
1.2.2.0	<ul style="list-style-type: none"> • Added: Stored Procedures to be used together with Microsoft PowerBI • Changed: default zoom level Simulation Summary Detail report
1.2.0.0	<ul style="list-style-type: none"> • Added: Additional performance monitor for the Simulation and Step class to overcome alert noise if a Simulation or Steps is unhealthy • Added: Data Backlog monitor on the Collector target • Changed: Default collector path is %programfiles%\The Backbone\The Backbone End-User Performance MP\ • Changed: Alert description of the monitor reporting an invalid management group name in the license file • Fixed: Missing data monitor on the Simulation class is working as expected • Fixed: Incorrect object name if performance counters are used from the Data Warehouse • Changed: Simulation Summary report paper format to Letter • Added: Object Picker to the Simulation Summary report • Added: Drilldown functionality inside the Simulation Summary report • Added: Simulation Detail report with Step and threshold information • Added: Stored Procedures to be used together with Microsoft PowerBI • Added: Microsoft OMS integration using data ingestion API • Changed: License file reload each 4 hours so no Microsoft Monitoring Agent restart is needed after replacing the license file
1.1.0.0	<ul style="list-style-type: none"> • Effective instance and step transaction time thresholds are available as performance counters • License verification while the Microsoft Monitoring Agent is running • Display strings for all overridable parameters • Default collector path is changed to %PROGRAMFILES% • Data missing alert description now contains the Watcher Node name and instance name • Events are written in the Operations Manager event log with an own Source • License target is removed after removing the license file • Instance target is created if the watcher location attribute is empty • License target properties information is added correctly after starting the Microsoft Monitoring Agent

Revision	Changes
	<ul style="list-style-type: none">• Average counter in the simulation summary report is changed to a weighted average
1.0.0.0	Initial version: <ul style="list-style-type: none">• Processing of simulation data from one or more Watcher Nodes;• Monitors for the health state of the simulation and simulation steps based on the state and performance;• Performance collection rules for the simulation and simulation step performance;• Monitor to detect missing data in case a simulation is not running anymore;• Simulation summary report;• Monitors for license and product expiration.

1 Introduction

Organizations are interested in the quality of the delivered services, to control their organization (e.g. to act correctly on incidents) and to improve the quality of the delivered services. By default, SCOM is able to monitor the availability and the responsiveness of webpages and (simple) web-based applications.

SCOM is not able to simulate user activities on native applications or on webpages with specific plugins (like Java). To get this information in SCOM, user activities should be simulated by an external program/script and the collected data need to be transferred to SCOM. The Backbone End-User Performance Management Pack makes it possible to import the End-User Performance Simulation data into SCOM for further processing.

The Backbone End-User Performance Management Pack contains a health model for the simulations and corresponding workflows to generate alerts and collect performance information. The alert, state and performance information can be used for several purposes like SLA reports, trend analyzing or quick incident reaction. The Backbone End-User Performance Management Pack contains a predefined report.

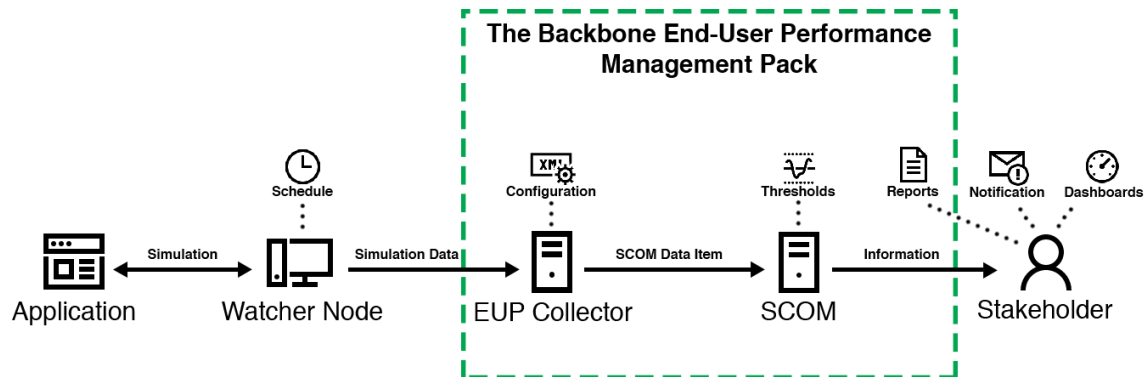
This document describes the basic installation and configuration of The Backbone End-User Performance Management Pack. To gain knowledge about the product and its functionality, this document starts to explain the architecture followed by the installation and configuration of The Backbone End-User Performance Management Pack. At the end, the document describes the format and syntax of the End-User Performance Simulation data files and the functionality of The Backbone End-User Performance Management Pack in detail. Appendix 2 describes the implementation process and gives an example of an End-User Performance implementation.

Additional information about topics like Microsoft OMS integration and using Microsoft PowerBI to visualize End-User Performance information see the online knowledge base, <https://www.enduserperformance.com/eupknowledgebase/>.

2 Architecture

This chapter describes the architecture of The Backbone End-User Performance Management Pack, which gives the relevant information to understand the working and the behavior of the product.

The figure below visualizes the architecture of an End-User Performance solution:



The solution contains several individual roles, together delivering the End-User Performance information. The table below describes all roles and their function.

Role	Function
Application	The application or ICT-services subject to the End-User Performance Simulations. By simulating the end-user behavior the availability and performance of the application is measured.
Watcher Node	System performing End-User Performance Simulations by executing simulation scripts, recording the application response and measuring the application responsiveness. The Watcher Node writes all End-User Performance Simulation data to the EUP Collector for further processing.
EUP Collector	The EUP Collector processes the End-User Performance Simulation data files by transforming the data to SCOM data items and handing them over to the Microsoft Monitoring Agent.
SCOM	SCOM Management Group storing, processing and presenting the End-User Performance Simulation data to the stakeholder
Stakeholder	Stakeholder using the End-User Performance information for his own needs like operational management, SLA management, performance analyzes, etc.

3 Product requirements

The Backbone End-User Performance Management Pack and its related components have some requirements to run properly. This chapter describes these requirements for the SCOM Management Group, the EUP Collector and the Watcher Node. These requirements must be met before the installation is started.

3.1 SCOM Management Group

The requirements for the SCOM Management Group are:

- SCOM 2012 or higher;
- OS requirements according to the version of SCOM;

3.2 EUP Collector

The EUP Collector can be any Windows system with one of the following roles:

- SCOM Management Server;
- SCOM Gateway.
- SCOM Agent or Microsoft Monitoring Agent;

The used Microsoft Monitoring Agent should report too only one management group, a multi-tenant Microsoft Monitoring Agent is not supported.

The following software must be installed on the EUP Collector:

- Microsoft .NET Framework 4.5.

A local folder to store the End-User Performance license, EUP Configuration file and End-User Performance Simulation data files. Default EUP Collector Path:

- `%PROGRAMFILES%\The Backbone\The Backbone End-User Performance MP\`

The Microsoft Monitoring Agent Default Action Account should have modify permissions on the EUP Collector Path (folder). To avoid issues with date and time formats, the Default Action Account should have the same regional settings as the Management Server, the SQL Servers and the SQL Databases.

3.3 Watcher Node

The Watcher Node can be any system that is able to run an End-User Performance script or program and write simulation data to the EUP Collector Path.

A system used as Watcher Node should be as representative to a normal end-user system as possible. The Microsoft Monitoring Agent can be installed on the Watcher Node to monitor the Watcher Node with SCOM. Be aware of policies and processes that can interrupt or influence the simulation script, as if screen savers, power settings (sleep modus), antivirus scan policies and (Windows) update policies.

To avoid issues with date and time formats, the End-User Performance script should write the date and time in the same format as specified in the regional setting of the Default Action Account on the EUP Collector.

4 Installation

This chapter describes the installation of The Backbone End-User Performance Management Pack. First, complete all preparation activities (paragraph 4.1) before importing The Backbone End-User Performance Management Pack files into SCOM. Paragraph 4.2 describes the steps to import The Backbone End-User Performance Management Pack files.

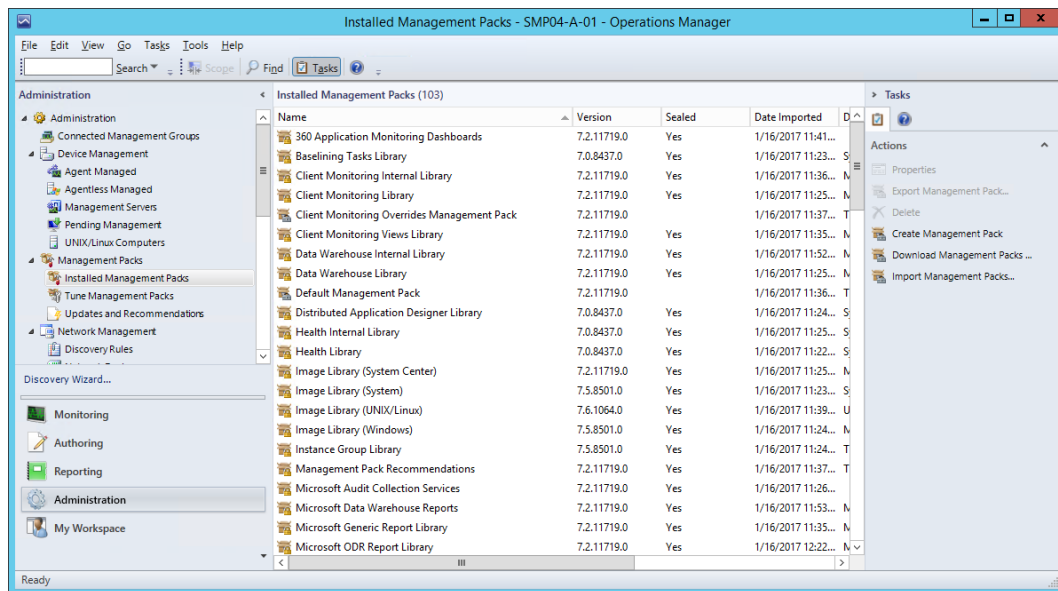
4.1 Preparation

Before The Backbone End-User Performance Management Pack is imported, check the status of the database and its storage. Depending on the amount of End-User Performance Simulation data that will be collected, it might be necessary to increase the size of the database.

4.2 Management Pack import

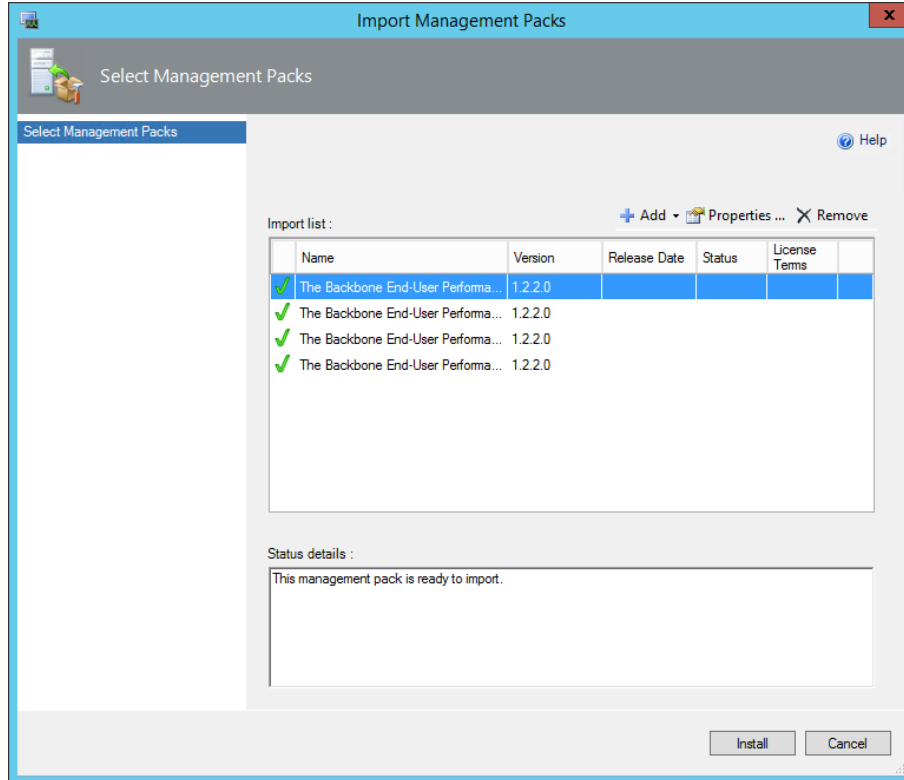
Import The Backbone End-User Performance Management Pack by carry out the following steps:

1. Open the SCOM Console;
2. Go to the **Administration** pane and choose **Installed Management Packs**;



3. Select the action **Import Management Packs**;
4. Navigate to the installation folder and select the following four files:
 - a. TheBackbone.SMP04.Discovery.mp
 - b. TheBackbone.SMP04.Library.mp
 - c. TheBackbone.SMP04.Monitoring.mpb
 - d. TheBackbone.SMP04.Reports.mpb

5. Press **Install** to import The Backbone End-User Performance Management Pack;



6. Press **Close** to close the windows.

5 Upgrade

This chapter describes the upgrade of The Backbone End-User Performance Management Pack. First, complete all preparation activities (paragraph 5.1) before upgrading The Backbone End-User Performance Management Pack. Paragraph 5.2 describes the steps to upgrade The Backbone End-User Performance Management Pack and paragraph 5.3 the steps to be taken after upgrading the management pack.

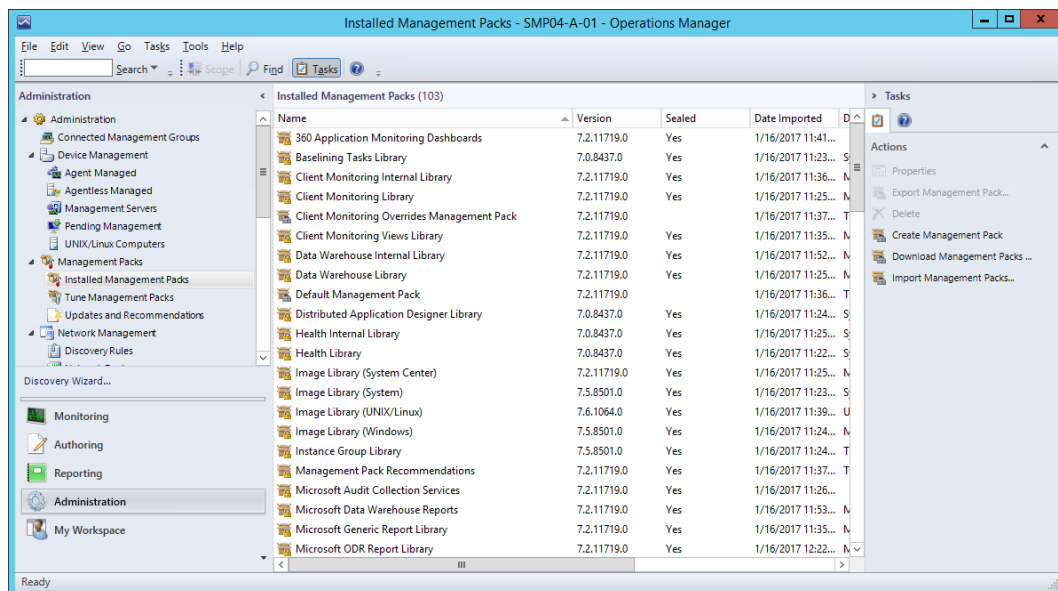
5.1 Preparation

Before The Backbone End-User Performance Management Pack is upgraded, export all management packs containing overrides for The Backbone End-User Performance Management Pack.

5.2 Management Pack upgrade

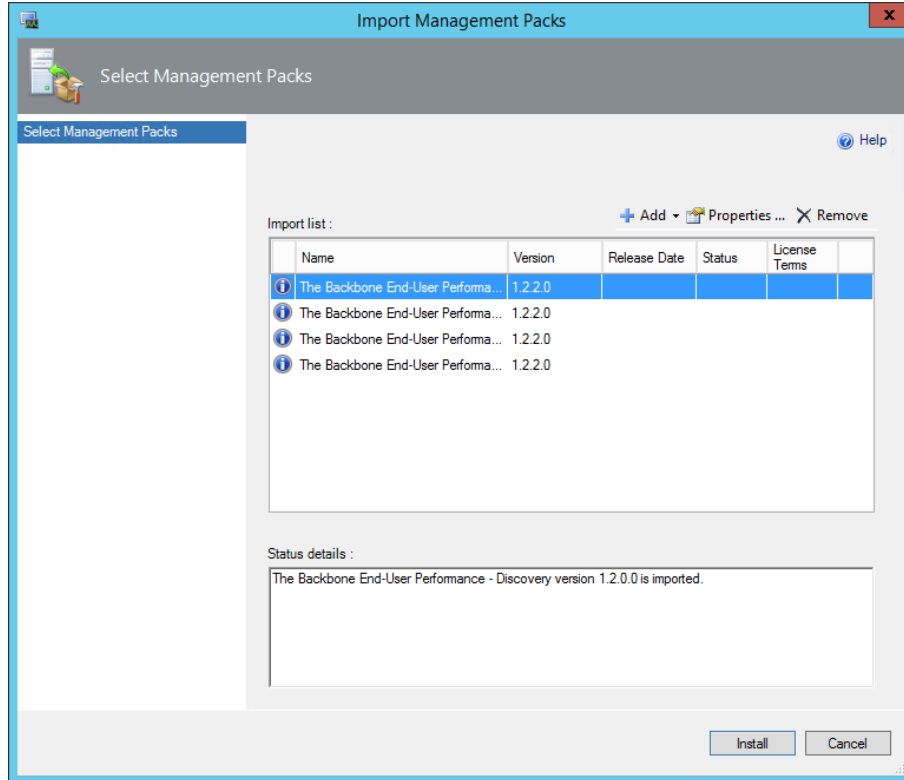
Upgrade The Backbone End-User Performance Management Pack by carry out the following steps:

1. Open the SCOM Console;
2. Go to the **Administration** pane and choose **Installed Management Packs**;



3. Select the action **Import Management Packs**;
4. Navigate to the installation folder containing the new version of the management pack and select the following four files:
 - a. TheBackbone.SMP04.Discovery.mp
 - b. TheBackbone.SMP04.Library.mp
 - c. TheBackbone.SMP04.Monitoring.mpb
 - d. TheBackbone.SMP04.Reports.mpb

5. Press **Install** to upgrade The Backbone End-User Performance Management Pack;



6. Press **Close** to close the window.

5.3 Post upgrade steps

After upgrading two steps might be needed depending on the current configuration and usage.

As of version 1.1.0.0, the default EUP Collector Path is changed to `%PROGRAMFILES%\The Backbone\The Backbone End-User Performance MP\`. If an upgrade from version 1.0.0.0 is performed and no override is created for the EUP Collector Path, the EUP Collector Path will change to the new default value. To make sure the EUP Collector keeps processing files, perform one of the two steps described below:

- Copy the `config.xml` and `license.lic` to the new default EUP Collector Path and make sure all simulation scripts output all data to the same path
- Create an override to configure the EUP Collector to use the old default path. See paragraph 6.3.3 for instructions how to create an override for the EUP Collector Path

In version 1.2.0.0, the *object name* of the performance collection rules is changed. Custom performance views, dashboard widgets, reports, etc. configured to select performance counters based on the simulation name as object name will not work anymore. After upgrading the management pack, those custom visualizations need to be changed to reflect the new object names.

6 Configuration

This chapter describes the configuration of The Backbone End-User Performance Management Pack. Paragraph 6.1 describes the configuration steps to discover an EUP Collector target on a Windows Computer target. The Backbone End-User Performance Management Pack uses an EUP Configuration file on every EUP Collector to configure the collected simulation instances. Paragraph 0 describes the configuration of the End-User Performance Simulation instances. The third paragraph, paragraph 6.3, describes configuration scenarios to adjust the behavior and the thresholds of the instances.

6.1 EUP Collector discovery

Carry out the following steps to configure The Backbone End-User Performance Management Pack:

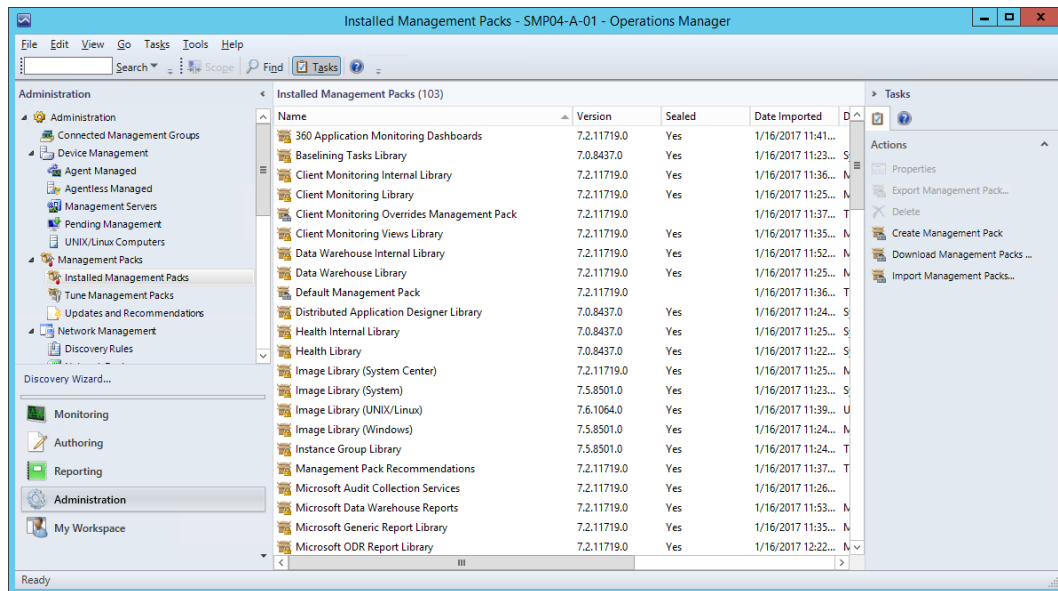
1. Create the EUP Collector Path on every system that should act as an EUP Collector. Ensure that the following access permissions apply:
 - a. The Default Action Account of the Microsoft Monitoring Agent should have at least read and delete permissions;
 - b. Account used to run the simulation scripts, the Watcher Node account, should have at least write permissions;
2. The Watcher Node account, running on the Watcher Node, will access the EUP Collector Path through a SMB share. Ensure that the following access permissions apply:
 - a. The Watcher Node account has write permissions on the SMB share of the EUP Collector Path.
3. Copy the received license file to the EUP Collector Path on each EUP Collector. The license file should have the name *license.lic*. Rename the file if needed.

*Note: The default folder location for the EUP Collector Path is:
%PROGRAMFILES%\The Backbone\The Backbone End-User Performance MP*

The default location of the EUP Collector Path can be changed by creating an override for the collector discovery. This option is explained in paragraph 6.3.3.

The Backbone End-User Performance Management Pack is a sealed management pack and cannot be used to store overrides. Create an unsealed management pack to save overrides for The Backbone End-User Performance Management Pack. Follow the steps below to create the management pack for the overrides:

1. Open the Operations Console;
2. Go to the **Administration** pane and choose **Installed Management Packs**;



3. Select the action **Create Management Pack** to create a new management pack;

The screenshot shows a 'Create a Management Pack' dialog box. The title bar reads 'Create a Management Pack'. Below the title bar is a 'General Properties' section. On the left is a 'Knowledge' tree. On the right is the 'Management Pack General Properties' form. The form has the following fields:

- ID: The Backbone.EUP.Overrides
- Name: The Backbone EUP Overrides
- Version: 1.0.0.0 (with a note 'For example, 1.0.0.0')
- Description: (empty text area)

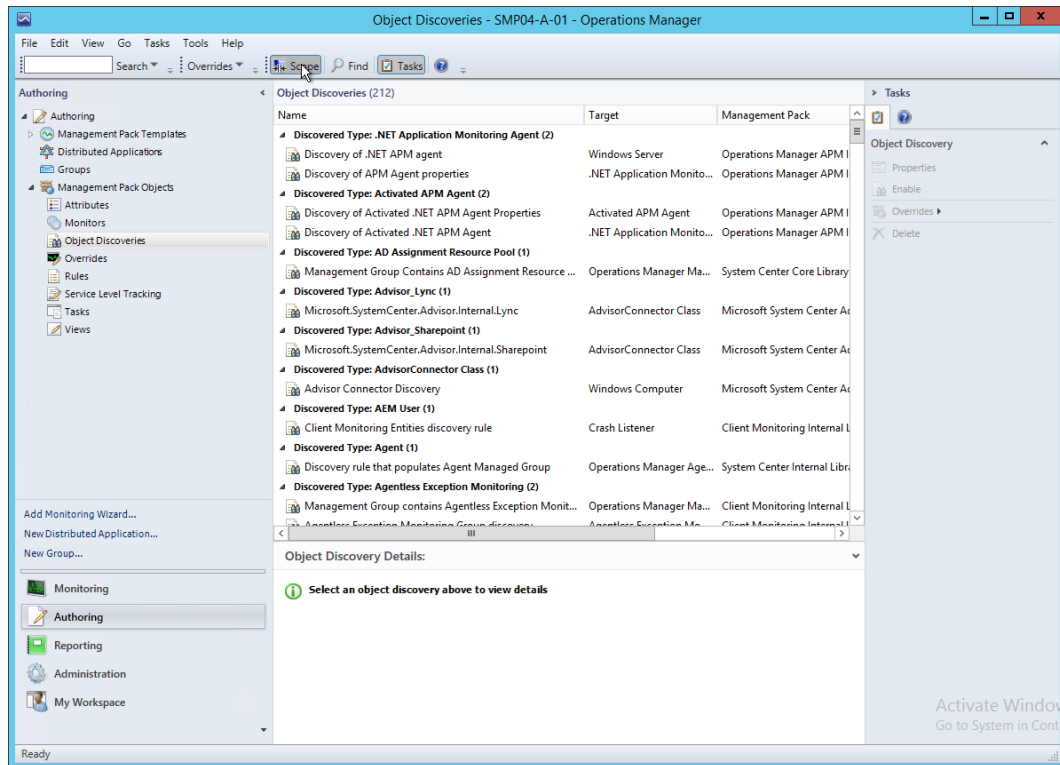
At the bottom of the dialog are four buttons: '< Previous', 'Next', 'Create', and 'Cancel'. The 'Next' button is highlighted with a mouse cursor.

4. Enter the name of the new management pack, for example *The Backbone EUP Overrides*
5. Press **Next**.
6. Press **Create**.

The EUP Collector discovery is disabled by default. To enable the EUP Collector discovery, an override need to be created for a specific Windows Computer target or group of Windows Computer targets. Follow the steps below to enable The Backbone End-User Performance Management Pack on a specific Windows Computer target.

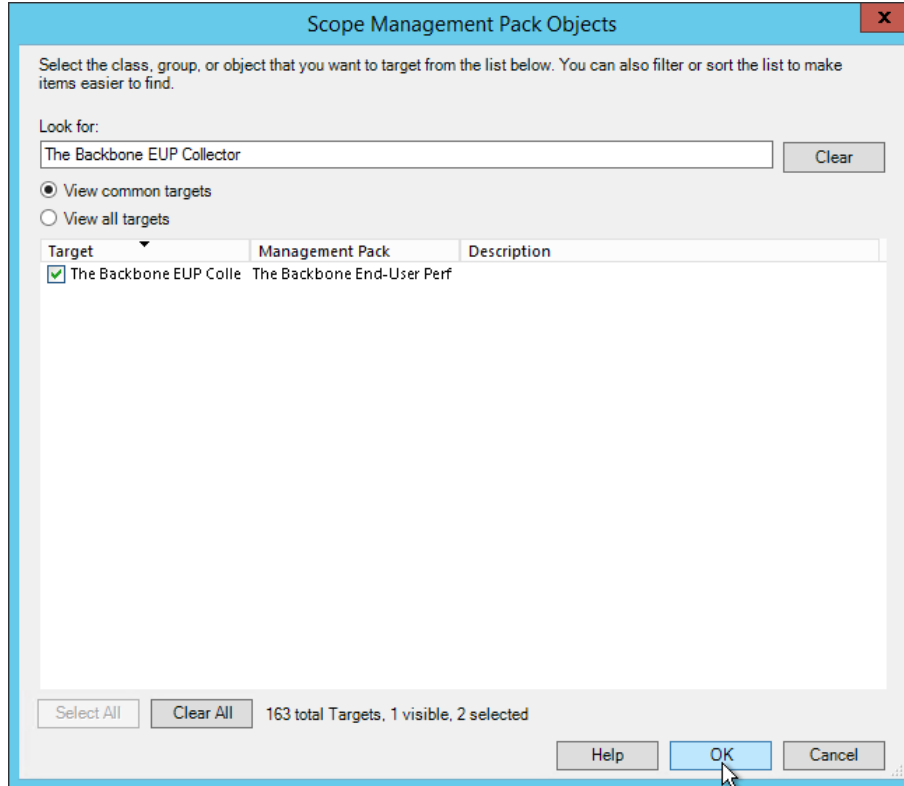
1. Open the Operations Console;
2. Go to the **Authoring** pane;

3. In the Authoring pane, navigate to **Management Pack Object -> Object Discoveries**;



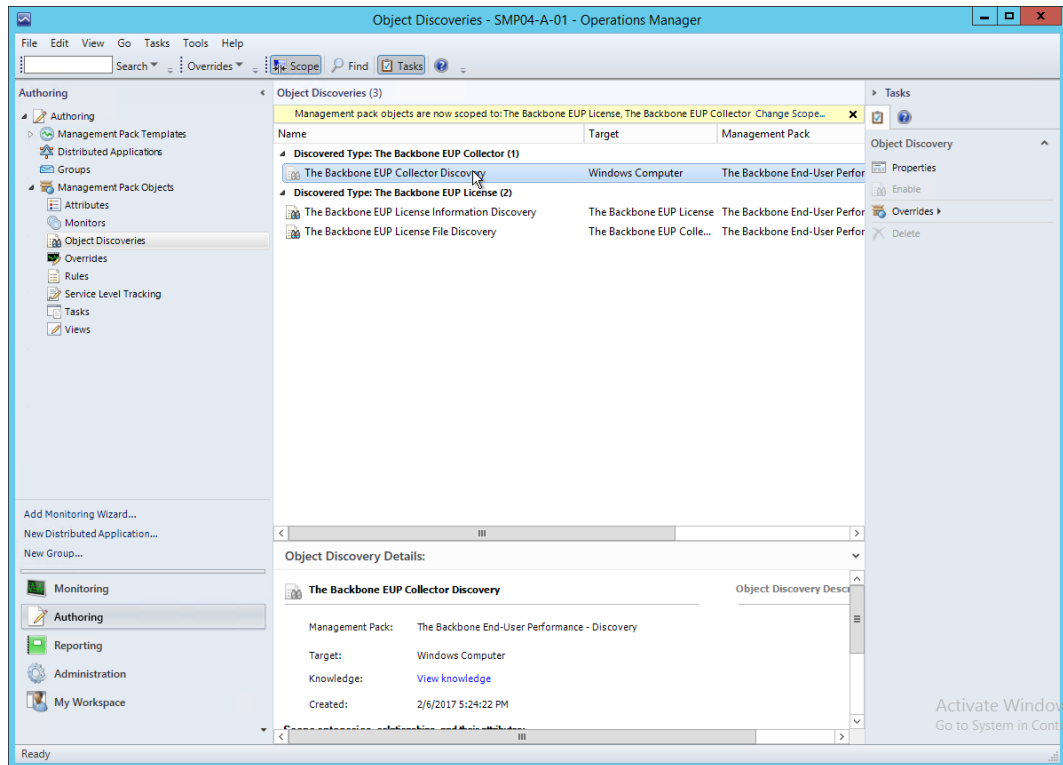
4. Click on **Scope** in the toolbar;

5. Search and select *The Backbone EUP Collector*;

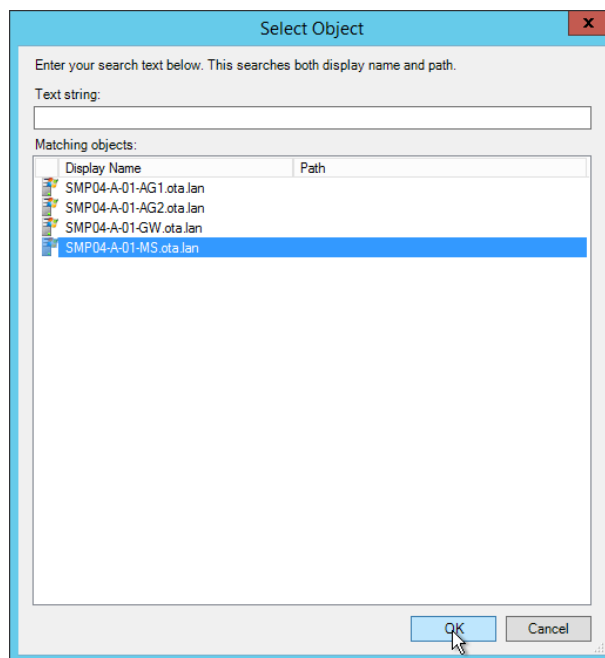


6. Press **OK**.

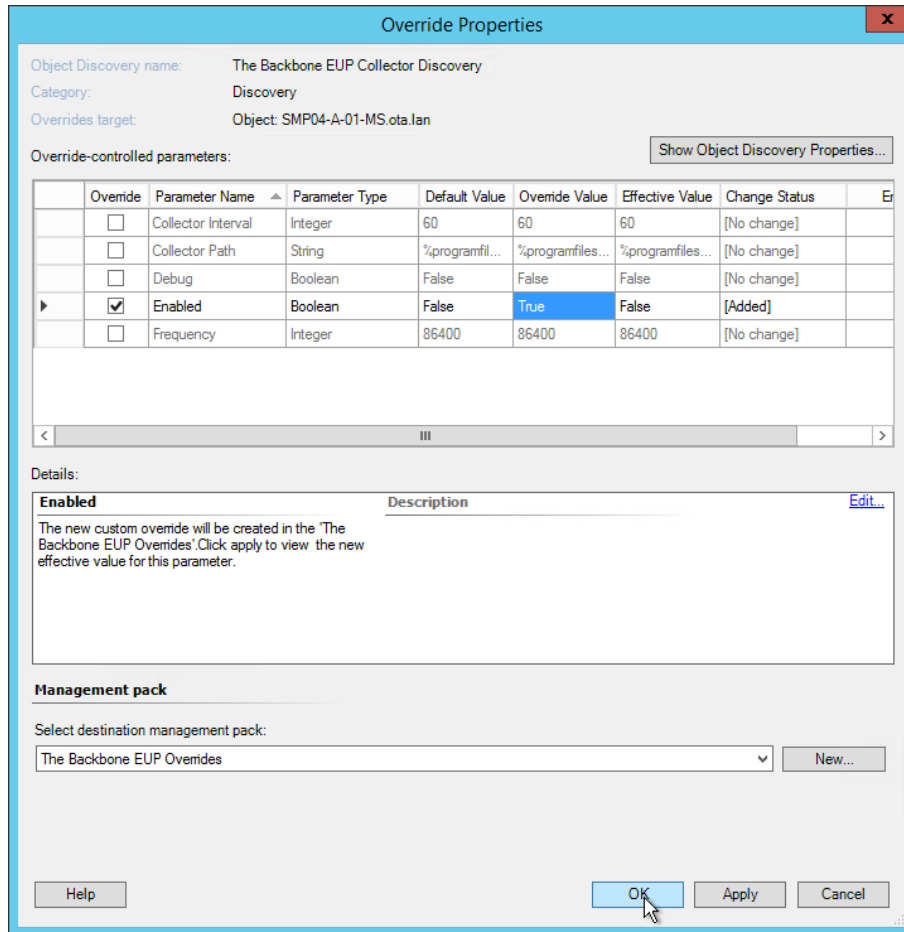
7. Select The Backbone EUP Collector Discovery object;



8. Select **Overrides** -> **Override the Object Discovery** -> For a specific object of class: **Windows Computer**;
9. Select the target that should act as EUP Collector;



10. Enable an override for the parameter **Enabled** and set the override value to *True*.
11. Select the earlier created management pack for overrides.



12. Press the **OK** button to close the window.

After a while, from a few minutes to a few hours, the EUP Collector target should appear as also the license target if the license file is placed in the application folder as described in the first steps of this paragraph.

6.2 EUP Configuration file

The Backbone End-User Performance Management Pack uses the EUP Configuration file to define the configuration of the End-User Performance Simulation instances. The discovery workflow reads the EUP Configuration file and creates targets for the simulation instances and steps based on the file content.

Details of the EUP Configuration file are:

Name	Config.xml
Type	XML
Location	EUP Collector Path
Default location	%PROGRAMFILES%\The Backbone\The Backbone End-User Performance MP\

The EUP Configuration file has the following syntax:

```
<?xml version="1.0" encoding="UTF-8"?>
<config>
  <instance id="" name="" frequency="">
    <watchers>
      <watcher location=""></watcher>
      ...
    </watchers>
    <steps>
      <step id="" name="" order=""/>
      ...
    </steps>
  </instance>
  ...
</config>
```

6.2.1 Simulation instance

The file starts with the root element `<config>`. The element `<config>` can contain one or more `<instance>` elements. The element `<instance>` describes the simulation instance that is monitored and is used to process and order the received End-User Performance Simulation data that is stored in the EUP Collector Path. The element `<instance>` has no element value. The table below shows the attributes of the element `<instance>`.

Attribute	Type	Mandatory	Explanation
id	String	Yes	The identifier of the simulation instance. The id can be a name, a number or a combination of both. The id should correspond with the simulation instance id that is used in the End-User Performance Simulation data files, which are written in the EUP Collector Path. Every simulation instance should have a unique id within the EUP Collector processing the simulation instance.
name	String	Yes	User-friendly name of the simulation instance, used for the display name of the corresponding target in SCOM. The use

Attribute	Type	Mandatory	Explanation
			of a descriptive but short name increases the readability of information in SCOM.
frequency	Integer	Yes	The interval in seconds in which SCOM should expect a new End-User Performance Simulation data file. SCOM uses this value to detect missing simulation data files.

The element `<instance>` has no value or attributes and contains the following two child elements:

1. `<watchers>`
2. `<steps>`

6.2.2 Watchers

The element `<watchers>` has no value or attributes and contains one or more elements named `<watcher>`. The element `<watcher>` specifies the FQDN or Hostname of the Watcher Node. The table below describes the attribute of the element `<watcher>`.

Attribute	Type	Mandatory	Explanation
location	String	No	User-friendly location name of the Watcher Node, used for the display name of the Watcher Node's location in SCOM. The use of a descriptive but short name increases the readability of information in SCOM.

The element `<watcher>` does not contain any child elements.

6.2.3 Steps

The element `<steps>` has no value or attributes and contains one or more elements named `<step>`. The element `<step>` has three attributes that specify the name and the order of the step. The element `<step>` does not have a value. The table below describes the attribute of the element `<step>`.

Attribute	Type	Mandatory	Explanation
id	String	Yes	The identifier of the step. The id can be a name, a number or a combination of both. The id should correspond with the simulation instance id that is used in the End-User Performance Simulation data files, which are written in the EUP Collector Path. Every step in a simulation instance should have a unique id.
Name	String	Yes	User-friendly location name of the step, used for the display name of the step in SCOM. The use of a descriptive but short name increases the readability of information in SCOM.
Order	Integer	Yes	Order of step occurrence within the simulation instance. This information is used to sort targets in reports in a logical order.

The element `<step>` does not contain any child elements.

Appendix 1 contains an example EUP Configuration file.

6.3 Custom configuration scenario's

The default behavior of the End-User Performance Management Pack can be adjusted using overrides. The most common overrides, see list below, are explained in this paragraph.

- Specifying instance and step thresholds;
- Increasing the EUP Collector Interval;
- Changing the EUP Collector Path;
- Changing the accepted missing data files.

6.3.1 Specifying instance and step thresholds

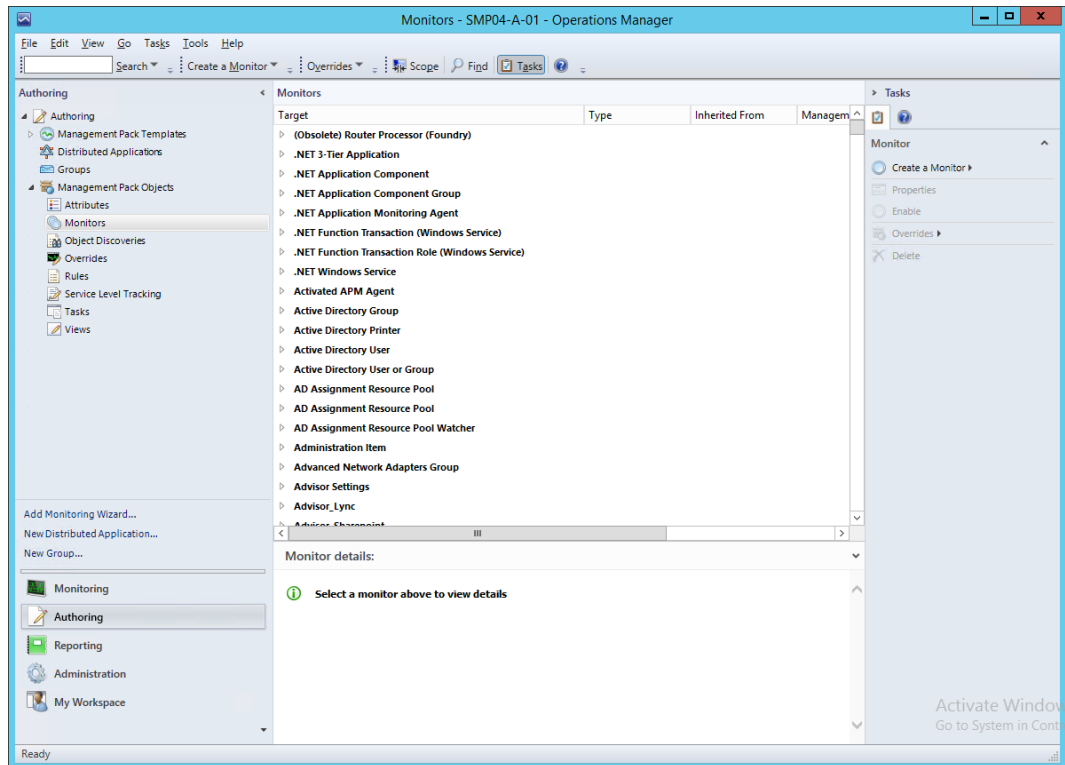
The Backbone End-User Performance Management Pack monitors the availability and performance for each End-User Performance Simulation instance and each corresponding step.

For performance monitoring, The Backbone End-User Performance Management Pack has two overridable parameters to specify when a state change should occur. The thresholds can be configured on an instance and step level and for a specific instance or step, a group of instance and / or steps or for all instances and steps. The complete simulation instance, every single step and for a group of simulation instances or steps. The thresholds can be changed using overrides, stored in the management pack created for overrides in paragraph 6.1.

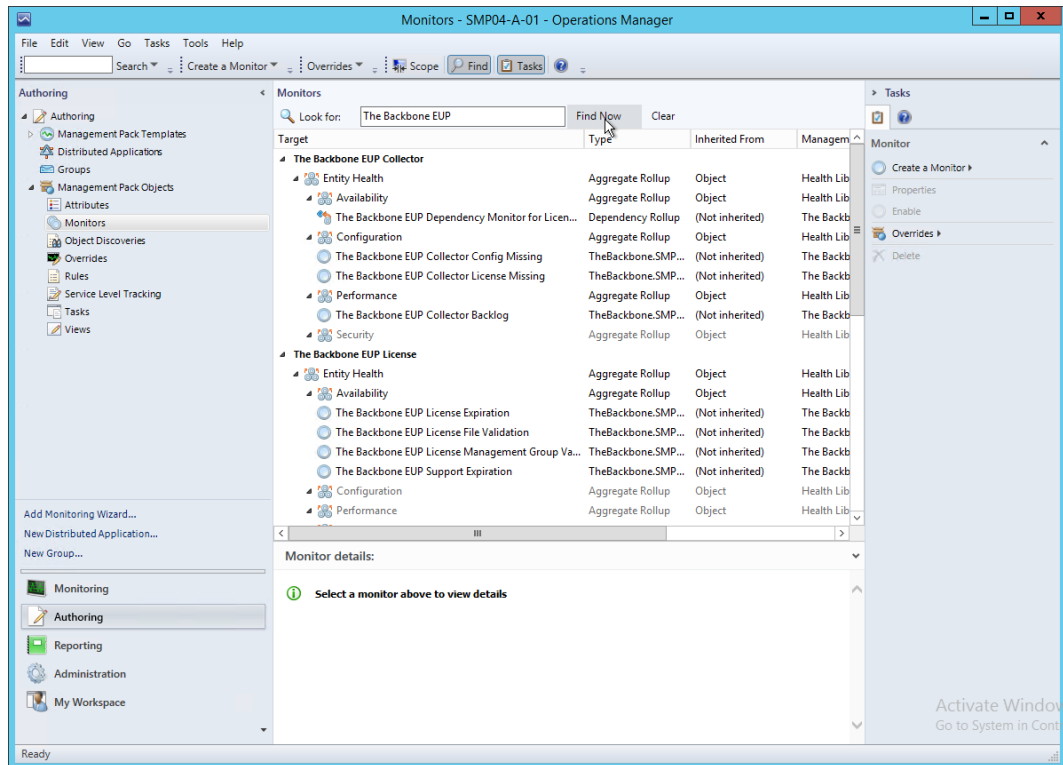
Follow the steps below to change the performance thresholds of the simulation instances or the steps.

1. In the SCOM Console, go to the **Authoring** pane;
2. In the Authoring pane, navigate to **Management Pack Object -> Monitors**;
3. Select the **Monitors**;

4. Ensure that the **Scope** in the toolbar is not selected;

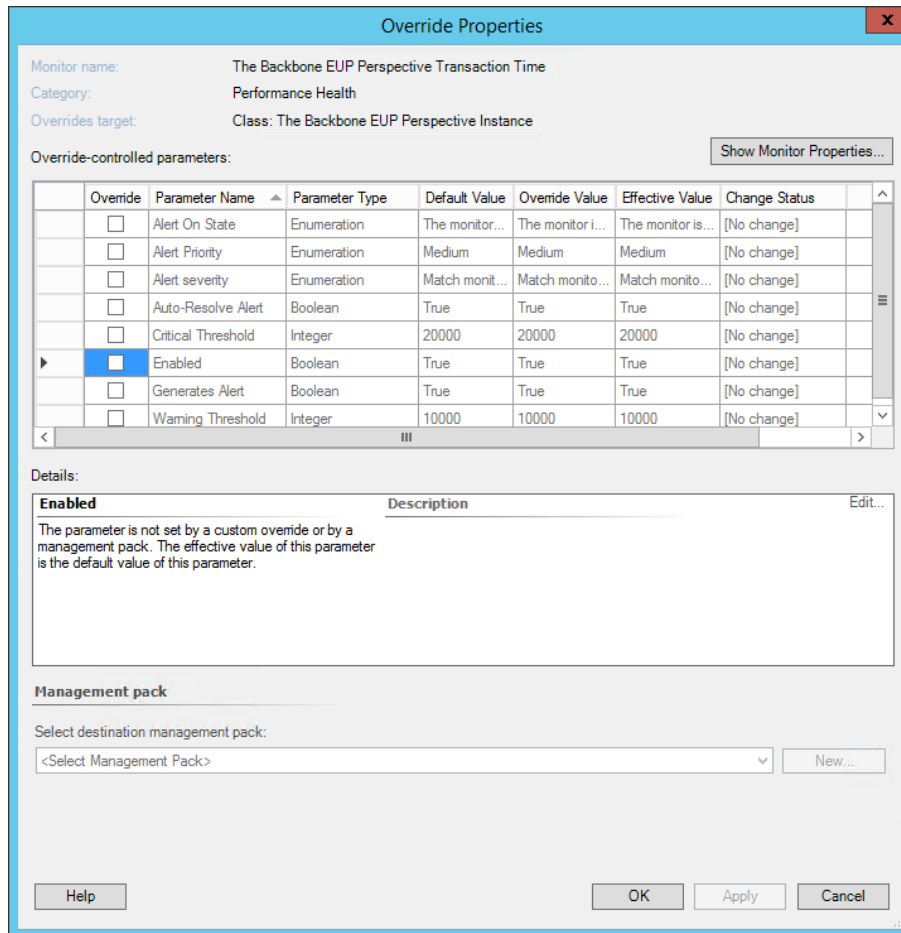


5. Search for *The Backbone EUP*;



6. Expand **The Backbone EUP Perspective Instance** or **The Backbone EUP Perspective Step** to change the thresholds for a simulation instance or step respectively;
7. Expand the **Entity Health** and expand the **Performance**;
8. Click on **The Backbone EUP Perspective Transaction Time** to change the thresholds for a simulation instance. Click on **The Backbone EUP Perspective Step Transaction Time** to change the thresholds for a step;
9. Select **Overrides** -> **Override the Monitor** -> **For a(II) (specific) object(s)/group ...**

10. Select, if needed, the object(s), instance(s) or step(s) that needs to be changed. A *Override Properties* Window for that object appears (The windows below gives a properties window of a step);



This window shows two threshold parameters, the *Warning Threshold* and the *Critical Threshold*. The *Warning Threshold* determines the response time in which the simulation instance or step should react to get a healthy state. If the response time is between the *Warning Threshold* and the *Critical Threshold*, the target will get a *Warning* state. The target will get a *Critical* state when the response time exceeds above the *Critical Threshold*.

11. Enable an override for the parameter that needs to be overwritten;
12. Select the management pack to store the override;
13. Press the **OK** button to close the window.

Repeat the steps above for all simulation instances and steps that require a customized threshold.

The tables below give the details of the parameters.

Parameter Name	Warning Threshold
Parameter Type	Integer
Default value	Instance: 10000 milliseconds Step: 2000 milliseconds

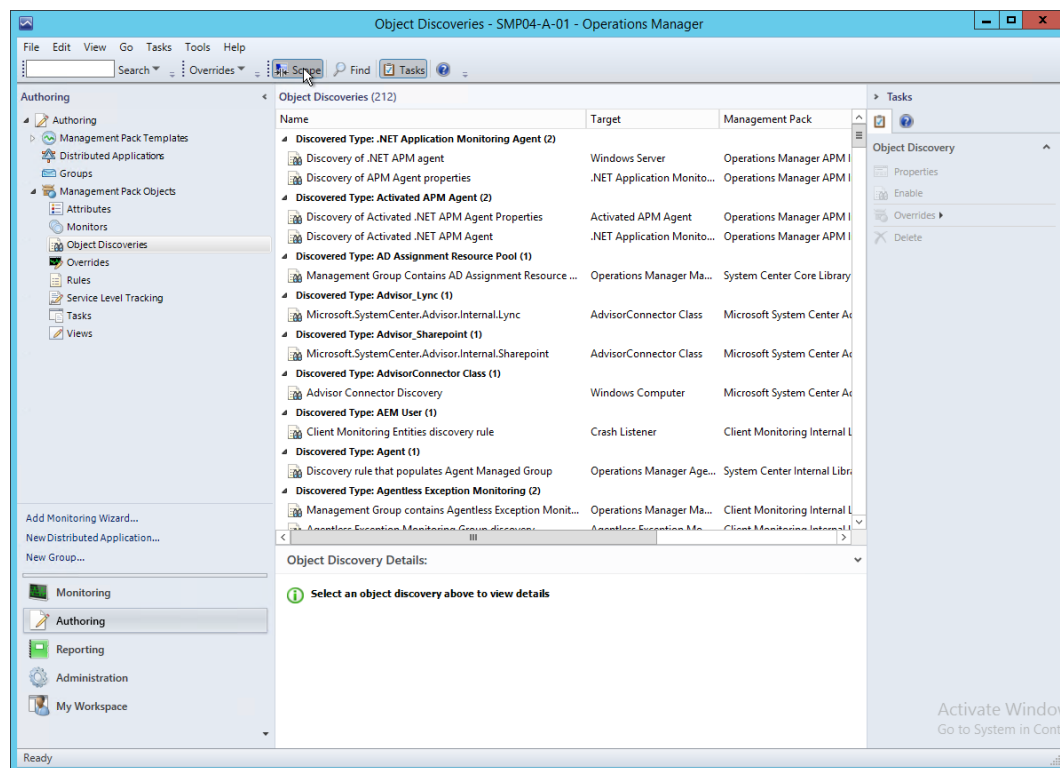
Parameter Name	Critical Threshold
Parameter Type	Integer
Default value	Instance: 20000 milliseconds Step: 4000 milliseconds

6.3.2 Increasing the EUP Collector interval

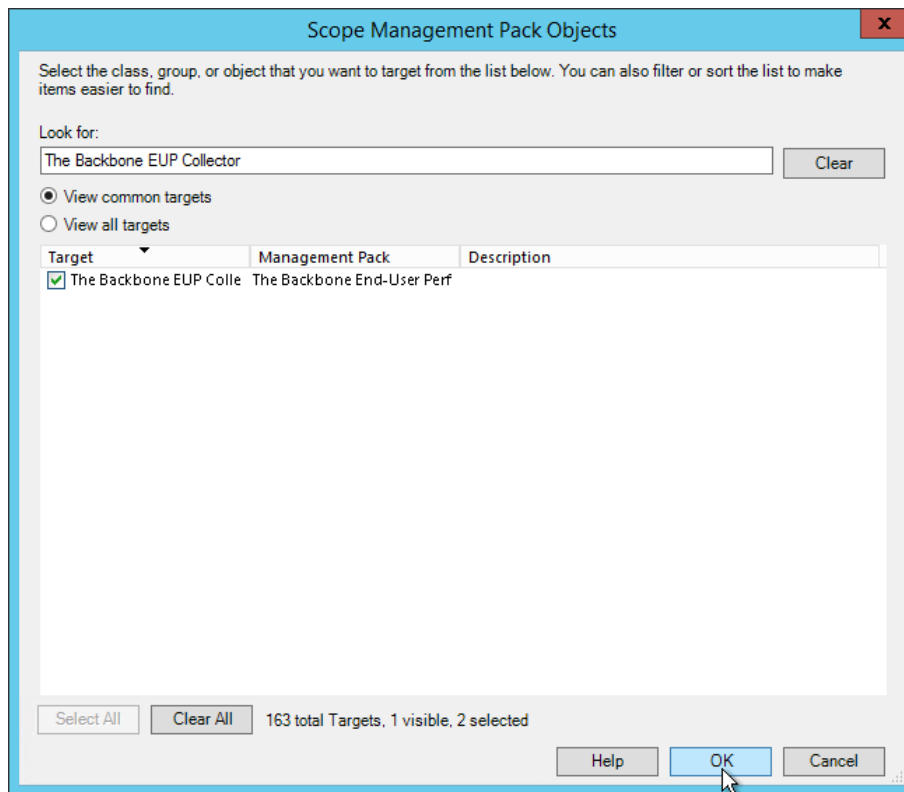
The parameter *Collector Interval* specifies the interval in seconds that The Backbone End-User Performance Management Pack checks the EUP Collector Path for new files. Every interval, the EUP Collector will process one file. If more files exist, the EUP Collector selects the newest. Adjust the parameter to increase or decrease the frequency to process the files.

Follow the steps below to increase the EUP Collector Interval:

1. Open the Operations Console;
2. Go to the **Authoring** pane;
3. In the Authoring pane, navigate to **Management Pack Object -> Object Discoveries**;

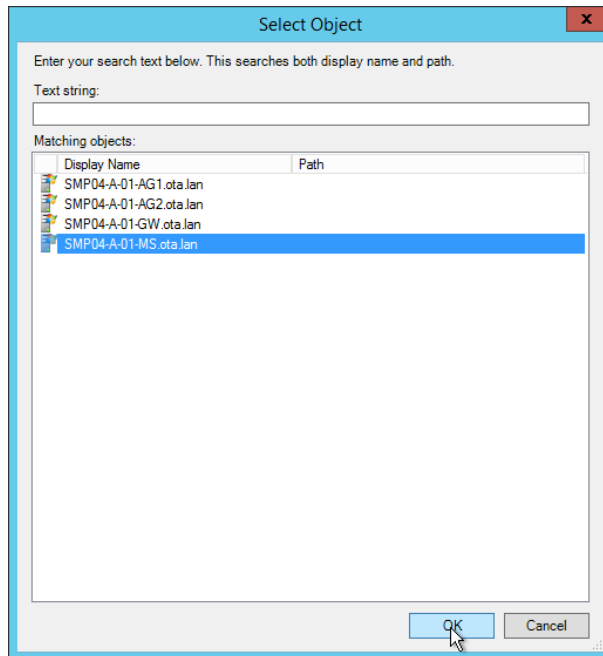


4. Click on **Scope** in the toolbar;
5. Search and select *The Backbone EUP Collector*;

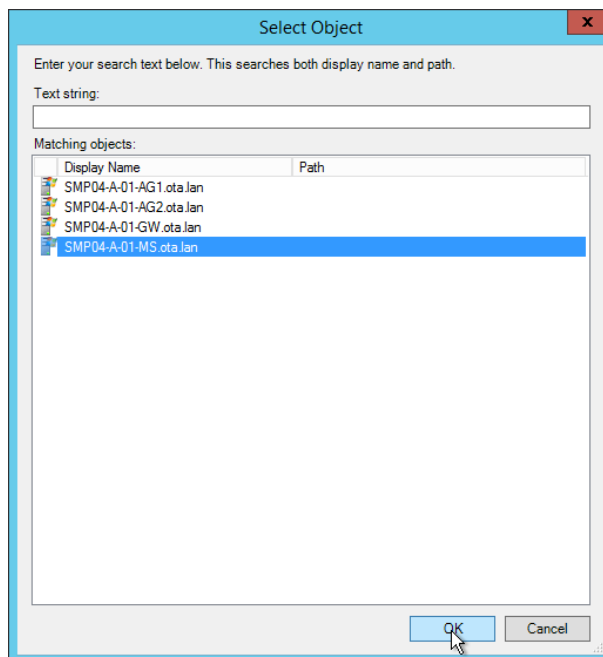


6. Press **OK**;

7. Select *The Backbone EUP Collector Discovery* object;

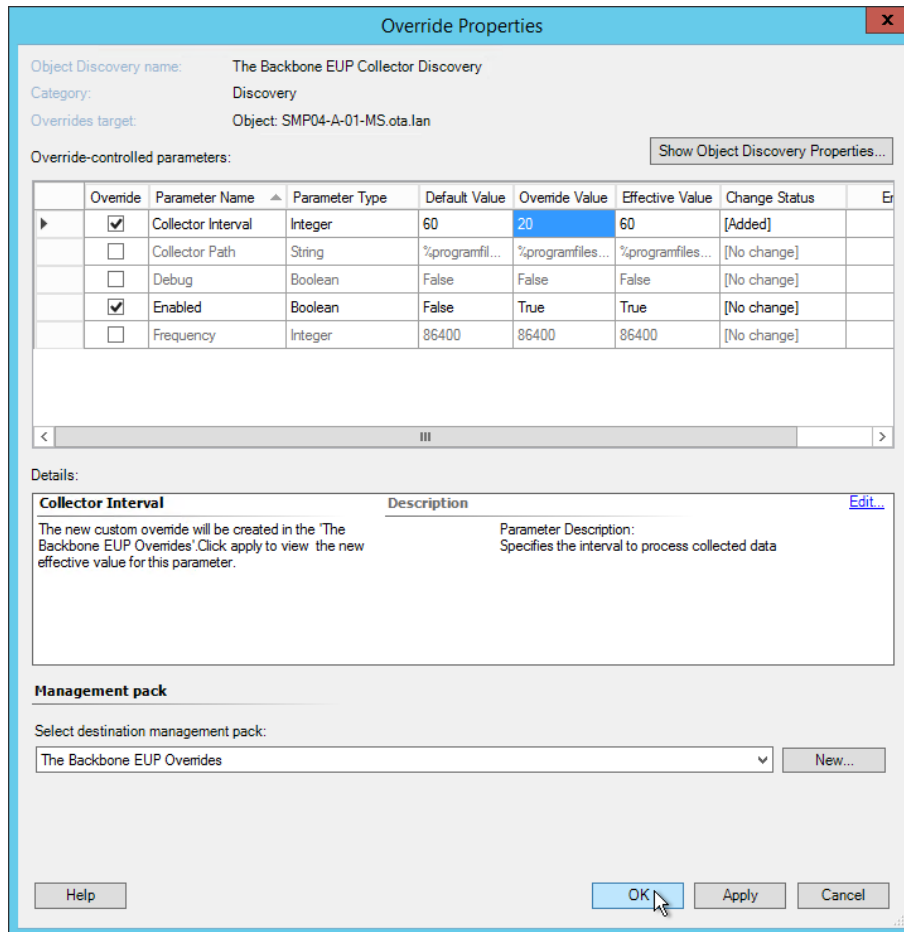


8. Select **Overrides** -> **Override the Object Discovery** -> **For a specific object of class: Windows Computer**;
9. Select the target that for which the EUP Collector interval should be changed;



10. Enable an override for the parameter **Collector Interval** and set the override value to the desired interval.

11. Select the earlier created management pack to store the override;



12. Press the **OK** button to close the window.

The table below gives the details of the parameter.

Parameter Name	Collector Interval
Parameter Type	Integer
Default value	60

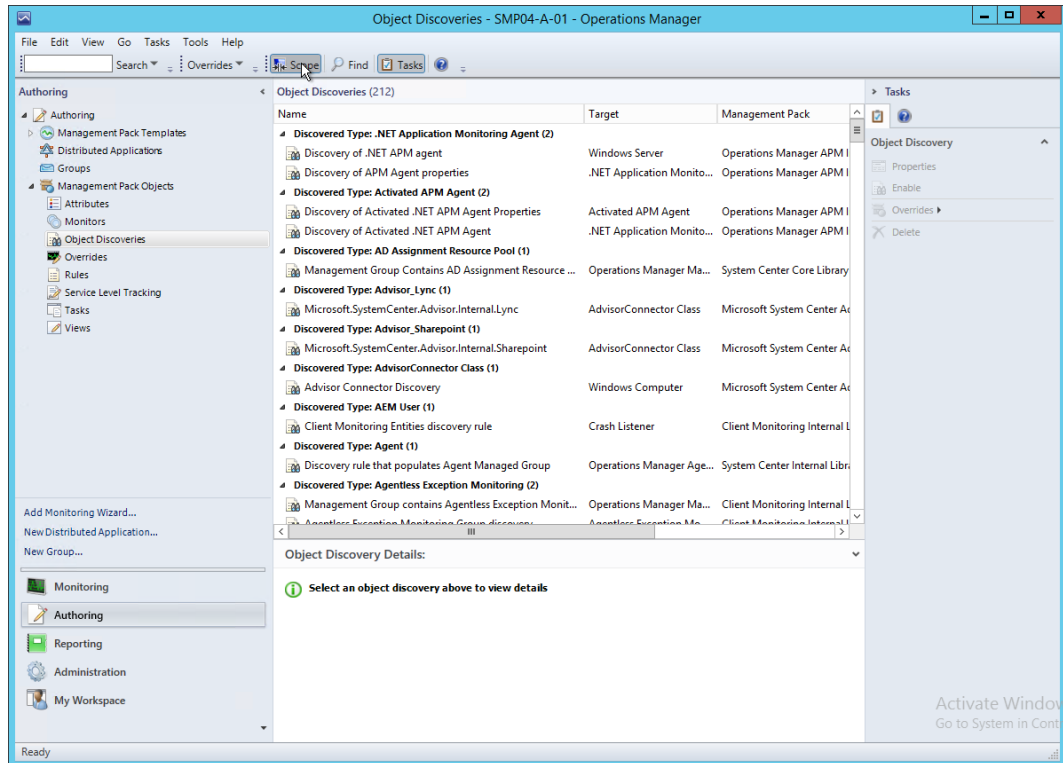
6.3.3 Changing the EUP Collector Path

The EUP Collector Path is the path used to store the End-User Performance Simulation data files, the configuration file and the license file. The EUP Collector Path can be changed to any folder, as long as the following is taken into account:

1. As mentioned in paragraph 3.2, the default Action Account should have modify rights on the EUP Collector Path (folder and files).
2. The Watcher Node(s) should have write access to the folder to store the End-User Performance Simulation data files.

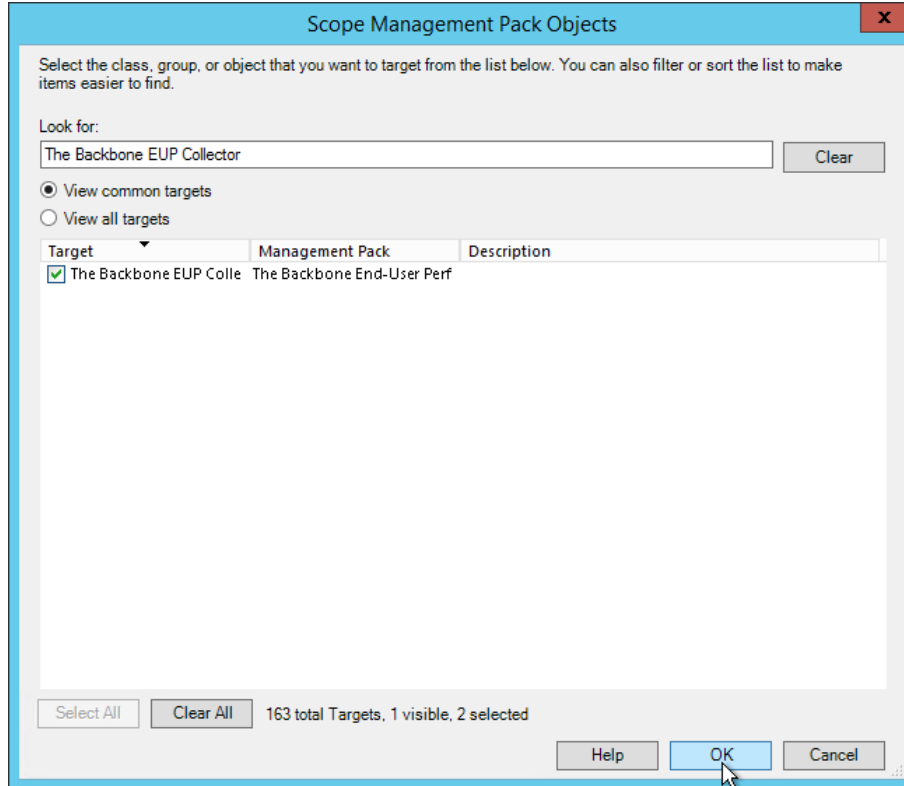
Follow the steps below to change the EUP Collector Path:

1. Open the Operations Console;
2. Go to the **Authoring** pane;
3. In the Authoring pane, navigate to **Management Pack Object -> Object Discoveries**;



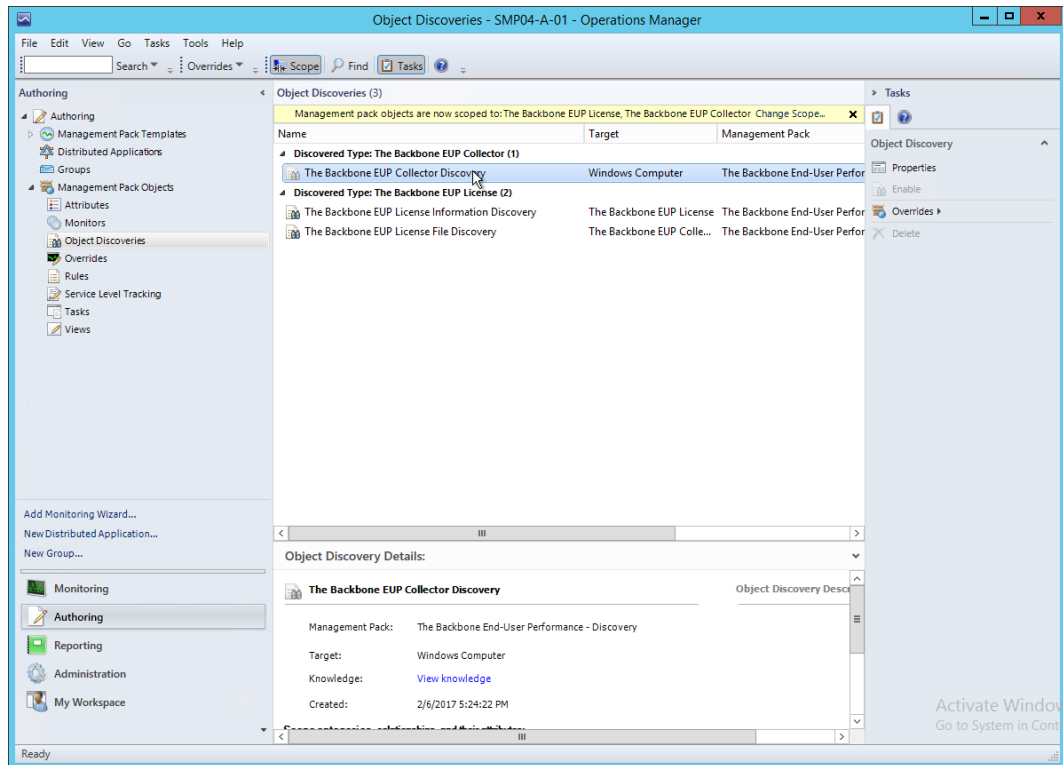
4. Click on **Scope** in the toolbar;

5. Search and select *The Backbone EUP Collector*;

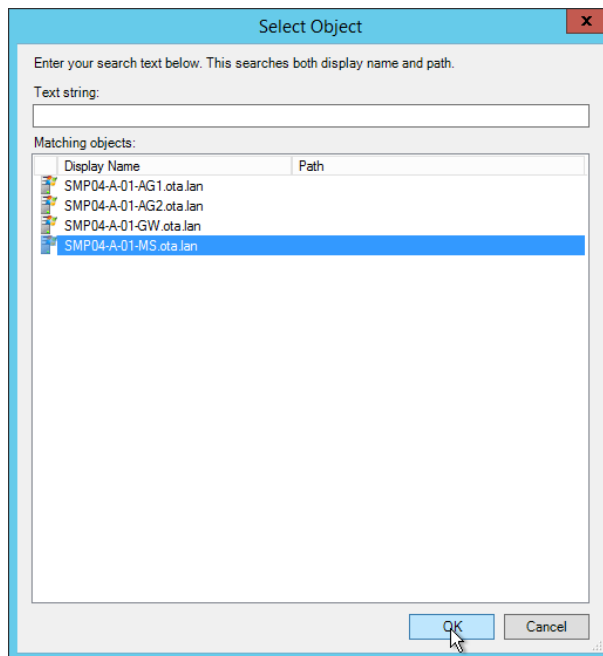


6. Press **OK**.

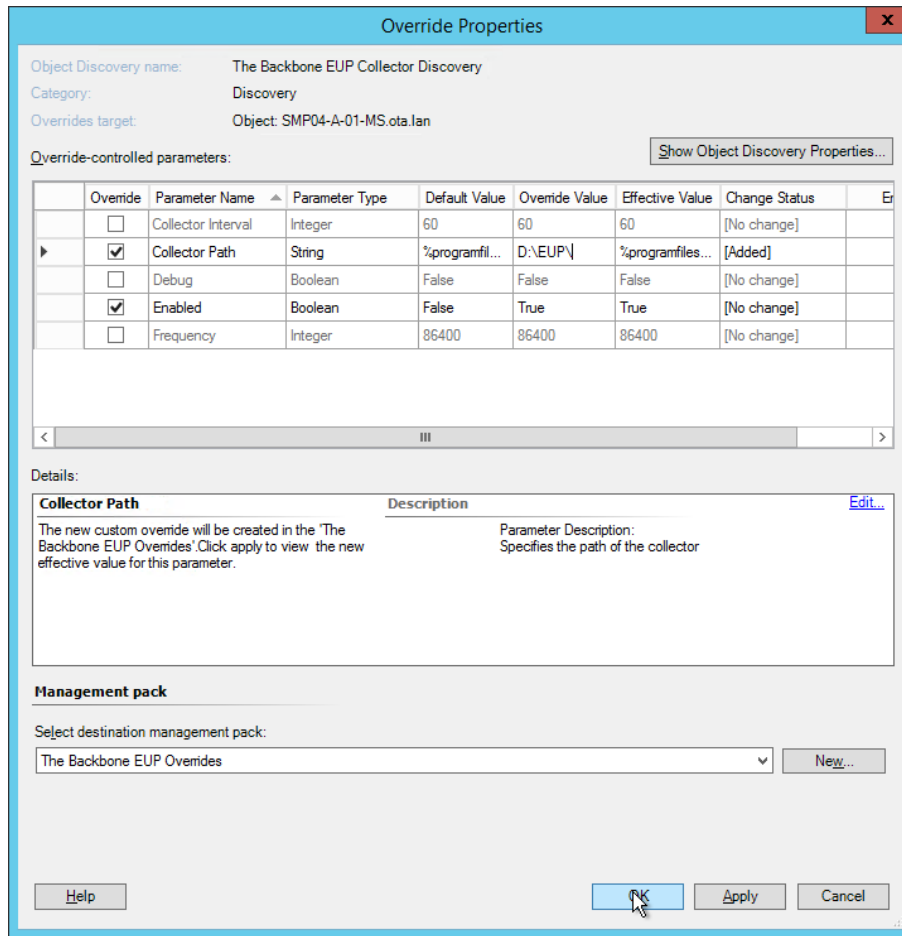
7. Select *The Backbone EUP Collector Discovery* object;



8. Select **Overrides** -> **Override the Object Discovery** -> For a specific object of class: **Windows Computer**;
9. Select the target for which the EUP Collector Path should be changed;



10. Enable an override for the parameter **Collector Path** and set the override value to the desired path;
11. Select the earlier created management pack for override;



12. Press the **OK** button to close the window.

The table below gives the details of the parameter.

Parameter Name	Collector Path
Parameter Type	String
Default value	%PROGRAMFILES%\The Backbone\The Backbone End-User Performance MP\

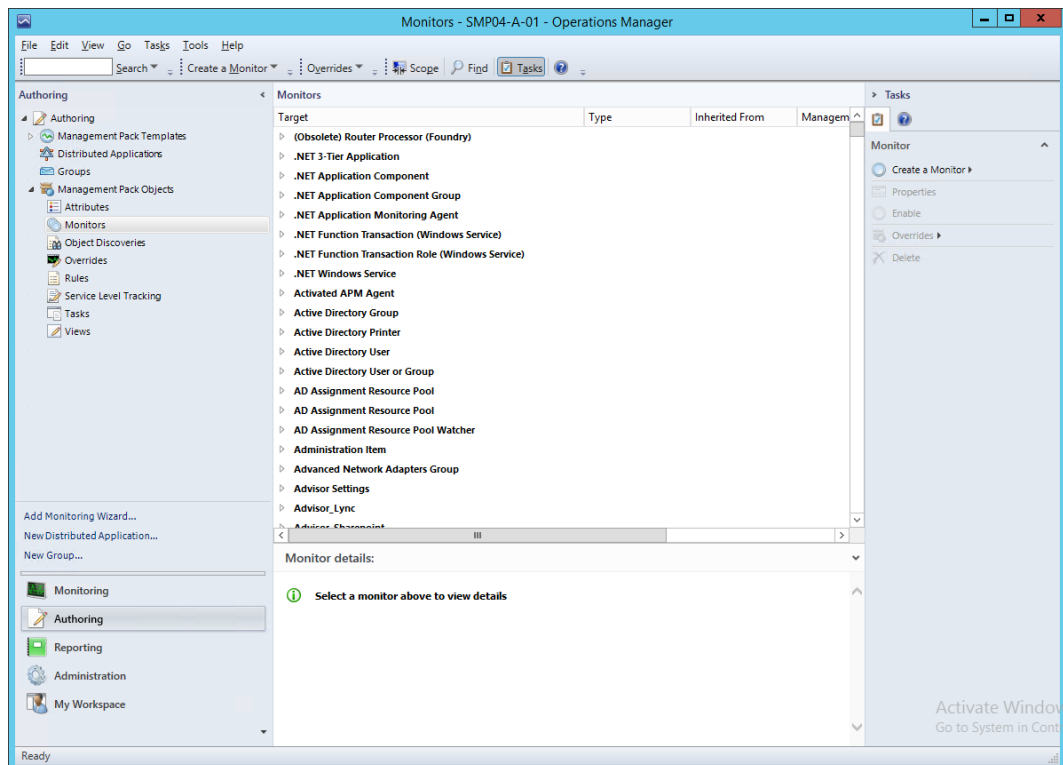
6.3.4 Changing the accepted missing data files

An End-User Performance Simulation is running with a specific interval, each run should result in a data file with simulation results. One or missing data files during a period of time must be investigated and resolved, frequent causes are Watcher Node shutdown, no automatic logon and a hanging simulation script due to an unknown application behavior.

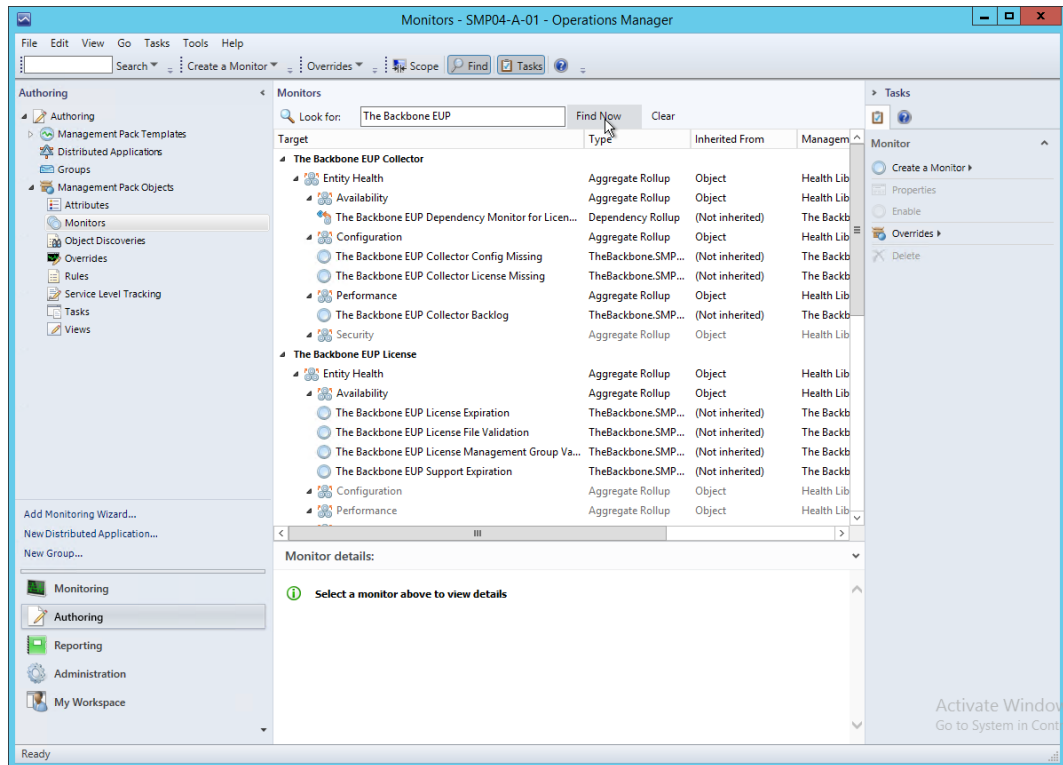
The End-User Performance Management Pack is helping to detect such situations by generating an alert if, for a period of time, no simulation data is received. This period is specified by the instance *frequency* in the EUP Configuration file and the count of acceptable missed data files as an overridable parameter on the monitor *The Backbone EUP Perspective Data Missing*.

Follow the steps below to change the acceptable count of missed data files:

1. Open the Operations Console;
2. In the **Authoring** pane, navigate to **Management Pack Object -> Monitors**;
3. Select the **Monitors**;
4. Ensure that the **Scope** in the toolbar is not selected;

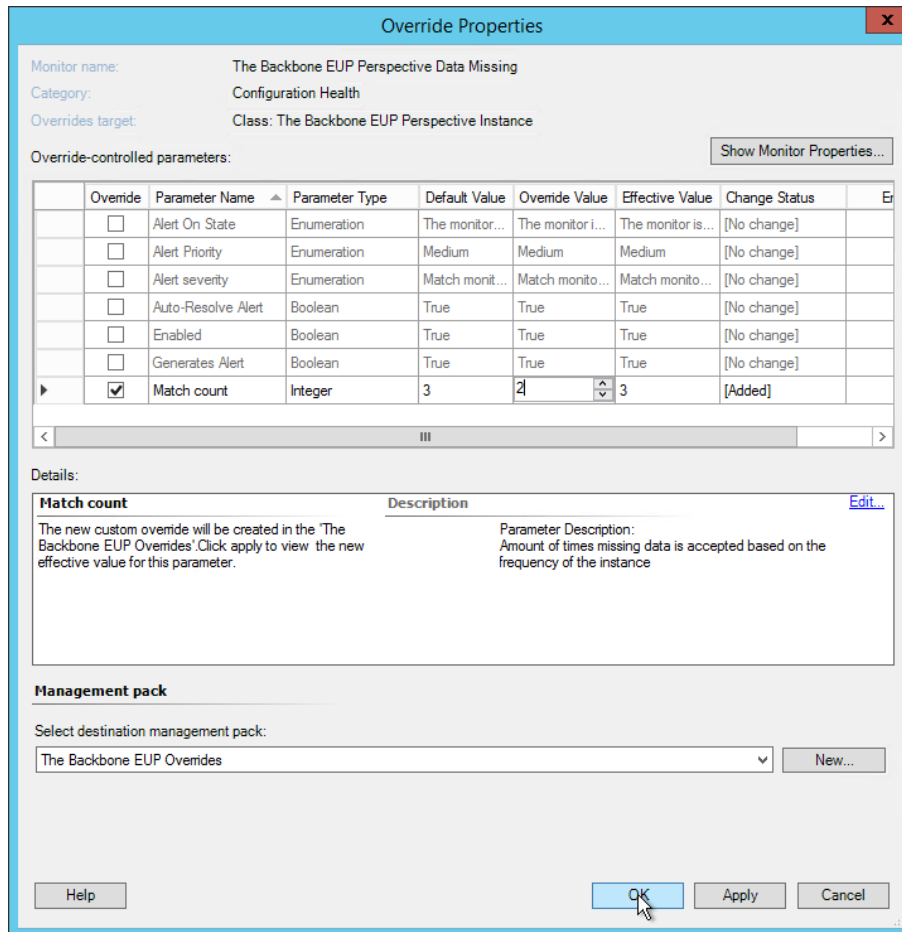


5. Search for *The Backbone EUP*;



6. Expand **The Backbone EUP Perspective Instance**;
7. Expand the **Entity Health** and expand the **Configuration**;
8. Click on **The Backbone EUP Perspective Data Missing**;
9. Select **Overrides** -> **Override the Monitor** -> For a(II) (specific) object(s)/group

10. After specifying the override scope, the *Override Properties* window will be shown



11. Enable an override for the parameter **Match Count** and specify the acceptable missing data files
12. Select the management pack to store the override.
13. Press the **OK** button to close the window.

The table below gives the details of the parameter.

Parameter Name	Match Count
Parameter Type	Integer
Default value	3

7 Simulation data file convention and syntax

The Watcher Node uses End-User Performance Simulation data files to store the results of a simulation. This chapter describes the filename convention in paragraph 7.1 and the syntax of the data files paragraph 7.2.

7.1 Filename convention

The End-User Performance Simulation data file should be stored in the root of the EUP Collector Path. There are no restrictions on the name of the file, except that the file needs the extension `.eum`.

Note: To create End-User Performance Simulation data files with a unique name, it is advised to use a GUID or a detailed date and timestamp in the filename.

Note: The Backbone End-User Performance Management Pack imports all files if the file extension starts with `.eum`. This means that a file with file extension `.eumm_simulation1` is imported as well.

7.2 File syntax

The End-User Performance Simulation data file is a text file in Unicode format. Each file starts with "Start data" and ends with "End data", both on a single line. The "Start data" and "End data" lines are used by the EUP Collector to verify if a file is ready to process. Without these lines, the EUP Collector will ignore the file.

The simulation data is written between the "Start data" and the "End data". Every line contains information about one simulation instance or step. There is no specific order in which the data should be written to the file.

The data elements of the simulation instance and steps are delimited with a pipe character ("|"). Every line containing simulation data should have the syntax described below:

```
<Date and Time>|<Watcher Node>|<Instance ID>|<Step ID>|<Measured time>|<Status>|<Context>
```

The table below describes each element within a single line.

Name	Type	Mandatory	Explanation
Date and Time	Datetime	Yes	The date and time the simulation or step is executed. The used date and time format should be according to the regional settings of the EUP Collector and the Default Action Account used by the Microsoft Monitoring Agent.
Watcher Node	String	Yes	The FQDN or Hostname of the Watcher Node that corresponds with the FQDN or Hostname that is

Name	Type	Mandatory	Explanation
			specified in the EUP Configuration file (see the configuration of the Watchers in chapter 0).
Instance ID	String	Yes	The ID of the simulation instance corresponding with the ID's specified in the EUP Configuration file (see the configuration of the simulation instance in chapter 0).
Step ID	String	Yes	The ID of the step corresponding with the ID's specified in the EUP Configuration file (see the configuration of the Steps in chapter 0). The step ID is left empty when the measured time represents the simulation instance time.
Measured time	Integer	Yes	The measured time in milliseconds. When the status is <i>false</i> , the value should be negative (-1). Negative values are excluded from statistics and avoids that the given (invalid) time negatively biased the presented information.
Status	Boolean	Yes	The status of the measured step or simulation instance to indicate whether the measured step is valid or invalid. Valid steps and simulation instances are indicated with a <i>true</i> or a 1, and invalid steps and simulation instances with a <i>false</i> or a 0.
Context	String	No	The context value is an optional string that can be used to add additional information to the alert or health state of the target. An example of additional information is the error message that is received by the simulation script.

Note: The Microsoft Monitoring Agent reads the data and performs a cast to the specified type (datetime, string or boolean). The format of the End-User Performance Simulation data in the text file should match the type that is specified for the element.

Note: Strings do not have to be places between quotes.

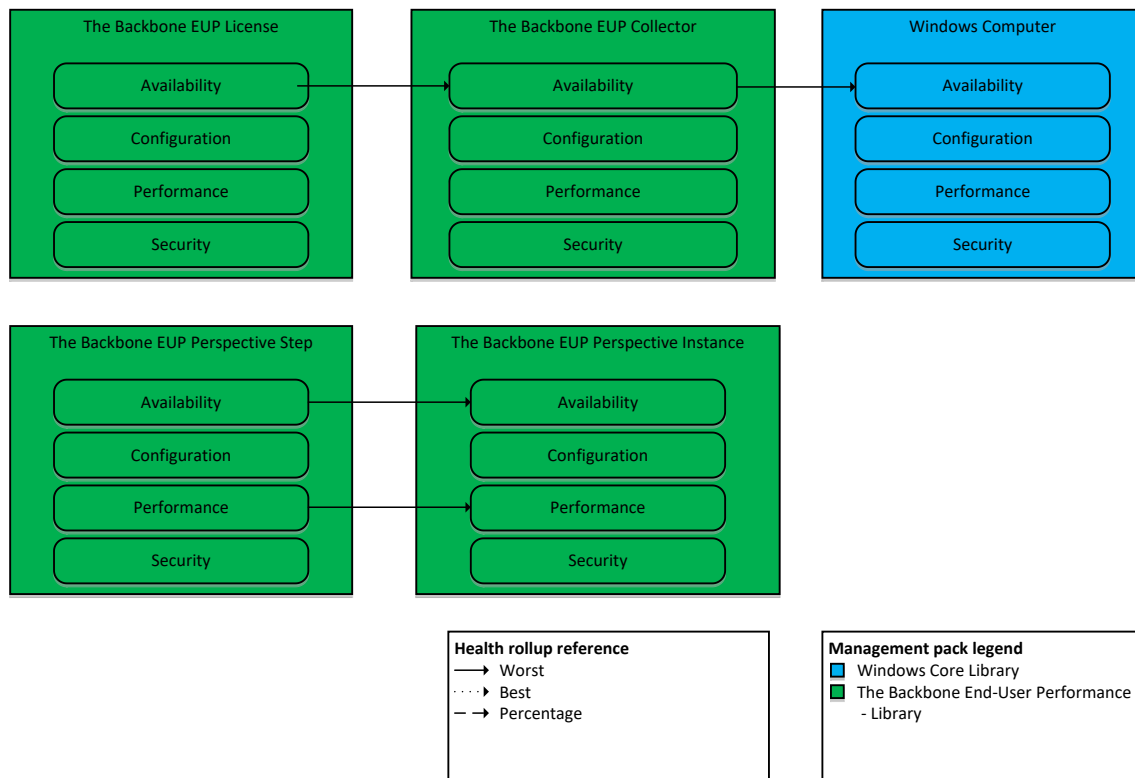
Note: End-User Performance Simulation data may be delivered in multiple files. This can be used when a simulation instance consists of multiple simulation scripts that are started nested. This construction is for example simulating an application running in a remote session

8 Management Pack explanation

This chapter describes the working of The Backbone End-User Performance Management Pack by explaining the Health Rollup and the Relationship in paragraph 8.1 and Paragraph 8.2 respectively. Paragraph 8.3 and Paragraph 8.4 describe the available Views and Reports packaged in The Backbone End-User Performance Management Pack.

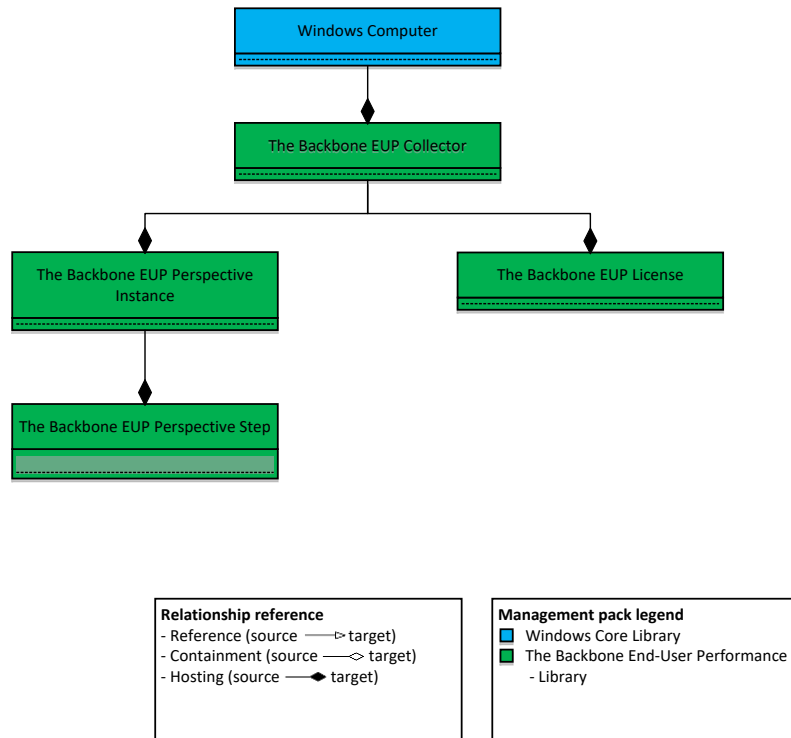
8.1 Health Rollup

The Backbone End-User Performance Management Pack contains four classes, the figure below shows the health rollup for the classes:



8.2 Relationships

The diagram below shows the relationship between the classes of The Backbone End-User Performance Management Pack.



8.3 Views

The Backbone End-User Performance Management Pack contains the following views in the Monitoring Pane:

Name	Active Alerts
Type	Alert view
Explanation	Shows all active alerts (resolution state <> 255) raised from the simulation instances of <i>The Backbone EUP Collector Class</i> and its hosting targets.

Name	Collectors
Type	State view
Explanation	Shows all End-User Performance targets and their states.

Name	End-User Performance Simulations
Type	State view
Explanation	Shows all End-User Performance Simulation instances and their states.

Name	End-User Performance
Type	Performance view
Explanation	Shows all performance counters for the End-User Performance Simulation instances and End-User Performance Steps.

Name	End-User Performance Simulation Steps
Type	State view
Explanation	Shows all End-User Performance steps and their states.

Name	Licenses
Type	State view
Explanation	Shows all End-User Performance licenses found on the End-User Performance Collectors.

8.4 Reports

The following reports are available in the management pack:

Name	Simulation Summary
Explanation	Presents a list of all End-User Performance Simulation instances and the corresponding availability and performance details within the selected period. Drilldown functionality is available to view details of a specific targets or specific time window.

About The Backbone

The Backbone, based in The Netherlands (Hengelo), is the Dutch market leader in delivering SCOM based monitoring solutions. With more than 10 years' experience, The Backbone knows how to support companies in ensuring their business continuity by monitoring their core information systems.

IT Services are more and more delivered from systems that are located in own data centers and a hybrid cloud. Moving to a hybrid cloud environment creates more connections and integrations between systems and applications. The increase in complexity creates the need for a complete overview of the environment by implementing monitoring from several perspectives like a technical, application and end-user perspective. With the right organizational embedding a complete implemented Monitoring Solution gives insights in the availability, performance and security level of the core applications, portals and infrastructure, with an own viewpoint for each stakeholder.

Appendix 1 Example EUP Configuration file

The code block below is an example EUP Configuration file used to specify one instance with two Watcher Nodes and two steps. The file should be located in the EUP Collector Path.

```
<?xml version="1.0" encoding="UTF-8"?>
<config>
  <instance id="1" name="EUP1" frequency="300">
    <watchers>
      <watcher>watcher1</watcher>
      <watcher location="Netherlands">watcher2</watcher>
    </watchers>
    <steps>
      <step id="1" name="step1" order="1"></step>
      <step id="2" name="step2" order="2"></step>
    </steps>
  </instance>
</config>
```

Appendix 2 Example End-User Performance - internet availability and performance

Introduction

The setup of an End-User Performance monitoring consists of several parts that needs to be aligned to each other. This appendix describes the steps to implement an End-User Performance Simulation instance of internet availability and performance monitoring. The example described in this appendix is rather small and easy, comparing to End-User Performance of business applications. The following aspects often increase the complexity of real world End-User Performance implementations:

- The use of multiple Watcher Nodes;
- An increased number of steps;
- Applications that are started in a remote session, like Citrix or RDS;
- Complex steps to simulate.

Although the complexity varies from implementation to implementation. The steps to implement the End-User Performance remain the same.

Global steps

An End-User Performance implementation has the following steps:

1. Start with collecting the monitoring requirements;
2. Installation and configuration of the EUP Collector Node and other system related actions (e.g. service accounts and ACLs);
3. Installation and configuration of the Watcher Node(s) and other system related actions (e.g. service accounts);
4. Development of the simulation script(s);
5. Testing the simulation script on the Watcher Node(s).
6. Scheduling the simulation script;
7. Creating the EUP Configuration file;
8. Adjusting monitoring thresholds.

Requirement collection

In this first step, the End-User Performance requirements are collected and documented, using one or more intakes with the stakeholders. To develop an End-User Performance script, it is important that for every step the action(s) taken and the expected response(s) are well defined.

For this example, the End-User Performance of internet availability and performance, the following requirements are used to develop the EUP Configuration file and the script:

1. The access time of the following websites is measured:
 - a. www.google.nl
 - b. www.microsoft.com
 - c. www.amazon.com
2. The DNS resolving is taken into account during the simulation.
3. The websites are accessed via Internet Explorer.

4. Every website displayed in a clean Internet Explorer Window.
5. Every single webpage should be displayed within 3 seconds (3000ms). The warning threshold shall be set to 3000ms. The critical threshold shall be set to 5000ms
6. For displaying 3 websites, the total time should not exceed the 6 seconds (6000ms). The average webpage loading and displaying time should be within 2 seconds. The warning threshold shall be set to 6000ms. The critical threshold shall be set to 10000ms

Note: The average time to display one webpage is 2 seconds. The displaying time of one single step is set one-second higher to avoid alerts when the display of one page is delayed for that moment. The purpose of the monitoring is internet availability and not website responsiveness. When all three webpages are delayed, there might be an issue with the internet connection.

7. Exception information, which is collected during the start of an Internet Explorer Window, shall be stored.
8. The verification if a website is loaded correctly is done based on a specific part of text on the website:
 - a. www.google.nl should contain the string: "// Google Inc."
 - b. www.microsoft.com should contain the string: "Zoeken op Microsoft.com"
 - c. www.amazon.com should contain the string: "About Amazon"
9. When the verification fails, the page body shall be stored.
10. The internet availability and performance is tested every 10 minutes.
11. The Watcher Node is a bare-metal desktop computer with the same specifications as the desktops that are used by the administrative staff.
12. The Operating System installed on the Watcher Node is the default Operating System installed on the laptops and desktops that are used by employees.
13. The Watcher Node should be an ordinary end-user system with the same specifications as the systems that are used by the end users.
14. The Watcher Node shall be located in the MER of the office building in Hengelo.

Installation and configuration EUP Collector Node

One of the existing servers will get the EUP Collector role. On this server the required directory, SBM share and users are created. The user used to start the End-User Performance Simulation script on the Watcher Node should have write access to EUP Collector Path. Depending on the network topology, the ACLs needs an update in order to enable the SMB protocol between the Watch Node and the EUP Collector.

Installation and configuration Watcher Node

As mentioned in the requirements, the Watcher Node should be an ordinary end-user system with the same specifications as the systems that are used by the end-users. Windows 10 is rolled out on a desktop computer. The Watcher Node and Windows 10 are configured in a way that:

- The system turns on automatically after an AC power loss.
- The standard installation procedure of the OS and standard applications is used.

- The user used to start the End-User Performance Simulation script is logged on automatically after startup.
- The screen of the user is locked immediately after logon.
- All pop-ups of standard applications are disabled.

During the development of the End-User Performance Simulation script it can be useful to temporary skip the auto logon and screen lock to test the simulation script directory on the Watcher Node. For the final test, the Watcher Node should be placed on the location that is agreed with the stakeholders. As specified in the requirements, the Watcher Node is placed in the MER of the building.

Development of the simulation script

A script is developed to simulate a user opening 3 webpages. This script uses the AutoIt UDF library for The Backbone End-User Performance, called "EndUserMonitor.au3". The library consists of common functions to collect instance and step information and to write the information to an End-User Performance Simulation data file. Before starting with the development, the latest version of the UDF Library for The Backbone End-User Performance Management Pack is downloaded from the download page of The Backbone. The AutoIt script mentioned below is developed to simulate a user who is opening 3 webpages:

```
#CS =====
Title .....: End-User Performance Monitor Script for webpages
AutoIt Version : 3.3.14.2
Language .....: English
Description ...: Script to simulate a user monitoring the performance and content
                  of 3 webpages
Author(s) .....: The Backbone - The Netherlands
#CE =====

#include <IE.au3>
#include "EndUserMonitor.au3"

#CS =====

Variable initialization

#CE =====

Local Enum $eStepID, $eWebPage, $eContent
Local $aWebPages[3][3]
$aWebPages[0][$eStepID] = "Google"
$aWebPages[0][$eWebPage] = "www.google.com"
$aWebPages[0][$eContent] = "// Google Inc."
$aWebPages[1][$eStepID] = "Microsoft"
$aWebPages[1][$eWebPage] = "www.microsoft.com"
$aWebPages[1][$eContent] = "Zoeken op Microsoft.com"
$aWebPages[2][$eStepID] = "Amazon"
$aWebPages[2][$eWebPage] = "www.amazon.com"
$aWebPages[2][$eContent] = "About Amazon"

Local $sFilePath = "D:\EUP\"
Local $sInstanceId = "I01"

Local $sHTML = ""
Local $oIE = Null

#CS =====

Flush DNS

#CE =====
```

```

RunWait(@ComSpec & " /C IPCONFIG /FLUSHDNS", "", @SW_HIDE)

#CS =====
Instance initialization
#CE =====

_EUP_Init($sFilePath)
_EUP_InstanceStart($sInstanceId)

#CS =====
Steps
#CE =====

; Call the function OpenSite for each row of the $aWebPages Array. The function
; OpenSite simulates one step.
For $i = 0 to UBound($aWebPages) - 1
    OpenSite ($aWebPages[$i][$eStepID], $aWebPages[$i][$eWebPage], _
        $aWebPages[$i][$eContent])
Next

#CS =====
Instance stop
#CE =====

_EUP_InstanceStop(True)

#CS =====
Data Write
#CE =====

_EUP_DataWrite ()
Exit (0)

#CS =====
Functions
#CE =====
#CS =====

OpenSite:
Carries out one step of the simulation. Opens the website and checks the content.
#CE =====

Func OpenSite ($sStep, $sSite, $sContent)

    Local $oIE

    _EUP_StepStart($sStep)

    $oIE = _IECreate($sSite)

    If @error Then
        _EUP_StepStop($sStep, False, "IE Error: " & ReturnLastError (@error))
    Else
        ; Reads the body of the HTML page.
        ; Checks if the body of the loaded page contains the value of $sContent.
        ; 0 means that the string is not found in the body text and that the
        ; expected page is not loaded.

```



```

    $sHTML = _IEBodyReadHTML($oIE)
    If StringInStr ($sHTML, $sContent) = 0 Then
        _EUP_StepStop($sStep, False, "Content of Webpage does not match: " & $sHTML)
    Else
        ; Webpage was loaded successfully.
        _EUP_StepStop($sStep, True)
    EndIf
EndIf

_IEQuit($oIE)
EndFunc

#CS =====

ReturnIEError:
Returns the error messages related to the IE object Error Code.

#CE =====

Func ReturnIEError ($ErrorCode)
    Switch $ErrorCode
        Case 1
            Return "General Error"
        Case 2
            Return "COM Error in Object reference"
        Case 3
            Return "Invalid Data Type"
        Case 4
            Return "Invalid Object Type"
        Case 6
            Return "Load Wait Timeout"
        Case 8
            Return "Access Is Denied"
        Case 9
            Return "Client Disconnected"
    EndSwitch

    ; If the error code does not match, return "Unknown error code"
    Return "Unknown IE error code: " & $ErrorCode
EndFunc

```

After a successful simulation, the End-User Performance Simulation data file should look like:

```

Start data
12-7-2016 13:59:40|PCWATCHER1.TBB.LOCAL|I01|Google|910|True|
12-7-2016 13:59:41|PCWATCHER1.TBB.LOCAL|I01|Microsoft|2241|True|
12-7-2016 13:59:44|PCWATCHER1.TBB.LOCAL|I01|Amazon|1893|True|
12-7-2016 13:59:40|PCWATCHER1.TBB.LOCAL|I01||5044|True|
End data

```

The script is compiled to an executable and stored in a directory on the Watcher Node. This executable will be started by the task scheduler.

Scheduling the simulation script

After the End-User Performance Simulation script is developed and tested on the Watcher Node, the End-User Performance Simulation script can be scheduled using the Windows task scheduler. Take the following into account when the End-User Performance Simulation script is scheduled:

- The End-User Performance Simulation script may only start when the previous run is finished.

Creating the EUP Configuration file

The EUP Configuration file contains all information about the End-User Performance Simulation instances and its steps. The ID and name are set to "I01" and "Internet availability and performance" respectively. Since the simulation should be scheduled every 10 minutes, the frequency is set to "600".

The `<watcher>` element specifies the FQDN of the watcher: `pcwatcher1.tbb.local`. For the friendly name of the location, the city of the office is used: `Hengelo`.

The step IDs are set to "Google", "Microsoft" and "Amazon". The friendly name contains the URL of the webpage.

The EUP Configuration file to simulate the internet availability and performance, should be:

```
<?xml version="1.0" encoding="UTF-8"?>
<config>
  <instance id="I01" name="Internet availability and performance" frequency="600">
    <watchers>
      <watcher location="Hengelo">pcwatcher1.tbb.local</watcher>
    </watchers>
    <steps>
      <step id="Google" name="www.google.com" order="1"/>
      <step id="Microsoft" name="www.microsoft.com" order="2"/>
      <step id="Amazon" name="www.amazon.nl" order="3"/>
    </steps>
  </instance>
</config>
```

Place the EUP Configuration file in the EUP Collector Path of the EUP Collector.

Adjusting monitoring thresholds

The default thresholds don't match with the requirements. Overrides are created to change the thresholds below:

- Steps Lower Threshold (Warning) to 3000ms;
- Steps Upper Threshold (Critical) to 5000ms;
- Simulation instance Lower Threshold (Warning) to 6000ms;
- Simulation instance Upper Threshold (Critical) to 10000ms.