Recast

Application Manager Legacy: MECM Detection Rules

Last Modified on 11.20.25

If you are uploading an application with a custom detection method, you can set the detection method as **Windows Installer**, **File attribute check**, or **Registry key value**.

NOTE: The detection method is currently in XML format.

You can also define multiple detection rules by adding different detectionClause sections inside the detectionMethods section and setting the comparison between rules.

Example with two detection rules

In this example, the connector is set as "and", meaning both rules must be met in order for the MECM client to detect the installation.

```
<detectionMethods>
   <detectionClause type="Directory" connector="and" Path="C:\Temp" DirectoryName="Folder 1" />
   <detectionClause type="WindowsInstaller" connector="and" ExpectedValue="19.2.0"
ExpressionOperator="IsEquals" PropertyType="ProductVersion" ProductCode="{176E78BB-985C-427A-AC57-
A9AE850DDDFB}" />
</detectionMethods>
```

Windows installer

Modifiable values:

- **Connector**: If you have multiple detectionClause sections, you can configure the link between detection methods. If you have only one detectionClause section, this value can be either one.
- **ExpectedValue** of the selected PropertyType. In general, this is the version of the Windows installer.
- **ExpressionOperator**: Comparison type
- **PropertyType** of the Windows installer. Recommend leaving as 'ProductVersion'
- **ProductCode** for the Windows installer

Template

```
<detectionMethods>
<detectionClause type="WindowsInstaller" connector="and,or" ExpectedValue="value"
ExpressionOperator="IsEquals,NotEquals,GreaterThan,LessThan,BeginsWith,EndsWith,Contains,NotContains"
PropertyType="ProductVersion" ProductCode="{productcode}" />
</detectionMethods>
```

Example 1 (Specific version)

```
<detectionMethods>
<detectionClause type="WindowsInstaller" connector="and" ExpectedValue="19.2.0" ExpressionOperator="IsEquals"
PropertyType="ProductVersion" ProductCode="{176E78BB-985C-427A-AC57-A9AE850DDDFB}" />
</detectionMethods>
```

Example 2 (Any version)

```
<detectionMethods>
<detectionClause connector="and" type="WindowsInstaller" ProductCode="{111111-2222-4CCD-ABCD-12345678}" Existence="$True" />
</detectionMethods>
```

File attribute check

Modifiable values:

- **Connector**: If you have multiple detectionClause sections, you can configure the link between detection methods. If you have only one detectionClause section, this value can be either one.
- **Path** to a folder containing your target file
- **FileName**
- **ExpectedValue** of the selected PropertyType. In general, this is the Windows installer version.
- **ExpressionOperator**: Comparison type
- **PropertyType** of the target file
- **Value**: Leave this as **$true**
- **Is64Bit**: Add this parameter with **$true** value to indicate that the target is 64-bit path. "This file or folder is associated with a 32-bit application on 64-bit systems." option is selected in detection method if you don't add this value at all.

Template

```
<detectionMethods>
   <detectionClause type="File" connector="and,or" Path="c:\any\directory\path" FileName="any filename under path" ExpectedValue="any value" ExpressionOperator="IsEquals,NotEquals,GreaterThan,LessThan,BeginsWith,EndsWith,Contains,NotContains" PropertyType="Size,DateCreated,DateModified,Version"  Value="$true" Is64Bit="$true" />
</detectionMethods>
```

Example

```
<detectionMethods>
   <detectionClause type="File" connector="and" Path="c:\any\directory\path" FileName="file.exe" ExpectedValue="19.5.5.2" ExpressionOperator="IsEquals" PropertyType="Version"  Value="$true" Is64Bit="$true" />
</detectionMethods>
```

Registry key value

Modifiable values:

- **Connector**: If you have multiple detectionClause sections, you can configure the link between detection methods. If you have only one detectionClause section, this value can be either one.
- **Hive**: Registry hive
- **KeyName**: Path to a target key

- **ValueName**: Name of target value
- **ExpectedValue**: Target value
- **ExpressionOperator**: Comparison type
- **PropertyType** of the target value
- **Value**: Leave this as **$true**
- **Is64Bit**: Add this parameter with a **$true** value to indicate that the target is 64-bit path. "This registry key is associated with 32-bit application on 64-bit systems" option is selected in detection method if you don't add this value at all

Template

```
<detectionMethods>
  <detectionClause type="RegistryKeyValue" connector="and,or"
Hive="LocalMachine,CurrentUser,CurrentConfig,ClassesRoot,Users" KeyName="any\key\path" ValueName="any
value name under key" ExpectedValue="any value" ExpressionOperator="IsEquals,NotEquals,GreaterThan,LessThan"
PropertyType="string,integer,version" Value="$true" Is64Bit="$true"  />
</detectionMethods>
```

Example

```
<detectionMethods>
  <detectionClause type="RegistryKeyValue" connector="and" Hive="LocalMachine"
KeyName="SOFTWARE\Mozilla\Mozilla Firefox" ValueName="CurrentVersion" ExpectedValue="109.0"
ExpressionOperator="IsEquals" PropertyType="string" Value="$true" Is64Bit="$true"  />
</detectionMethods>
```