## Alert Update Connector

### Overview:

The main purpose of the connector is to help customers get around the following two limitations:

1. No ability to specify custom field values, owner, and ticket id for alerts generated by monitors.
2. No ability to override custom field values, owner, and ticket id for alerts generated by rules.

The connector consists of four components:

1. **AlertUpdateConnector.exe** – NT Service which is the actual connector that receives alerts from OpsMgr and modifies them accordingly.
2. **ConnectorConfiguration.exe** – UI application which is used to configure the connector.
3. **AlertUpdateConnector.exe.config** – Configuration file for the connector which contains the following information: The path of the configuration file that contains information about which alerts to modify, the Management server name, and the frequency in seconds that the connector should use when checking for new alerts.
4. **ConnectorConfiguration.xml (or any other file name)** – Configuration file which contains information about which alerts should be modified by the connector.

### Requirements:

1. The connector service needs to run under an account which is a member of the OpsMgr Administrator user role. This is because the connector using certain SDK APIs which are only accessible to users in the Administrator role.

### Setting up the connector:

Setting up the connector is a three step process:

1. Extract the files from the zip file.
2. Create a configuration file which will contain the information required by the connector in order to properly function.
3. Configure the “Alert Update Connector” service.

**Steps to create the configuration file:**

1. Launch ConnectorConfiguration.exe
2. Select the rules or monitors that you are interested in. Right click and select **Specify Properties to modify**. This will bring up a dialog box.
3. Right click in the grid and select **Add Property** to specify the property and value. You can add more than one property
4. After you have done this for all the rules/monitors of interest, select **File->Save Configuration** and save the XML file. (Save it to a secure location, which can be a network share, accessible only by the local admin and members of the OpsMgr Administrator user role)
5. Edit the **AlertUpdateConnector.exe.config** file and specify the full file path in the **ConfigurationFilePath** element.
6. Edit the **AlertUpdateConnector.exe.config** file and specify the Management server name (or the NLB cluster name for a pool of management servers) in the **RootManagementServerName** element.
7. The default polling interval for alerts is 10 seconds. You can modify it by modifying the **PollingIntervalInSeconds** element.

**Steps to configure the service:**

1. Extract the files from the package zip file to a folder of your choice
2. Open the command prompt and navigate to the folder where the connector files were placed
3. Run the following command to install the “Alert Update Connector” service:
   1. 64bit OS – “**C:\Windows\Microsoft.NET\Framework64\v2.0.50727\InstallUtil.exe AlertUpdateConnector.exe**”
4. Run the following command using an account which is a member of the OpsMgr Administrators user role: **AlertUpdateConnector.exe –InstallConnector**. This command will install the connector into the OpsMgr management group.
5. Change the credentials of the “Alert Update Connector” service to an OpsMgr admin account
6. Start the “Alert Update Connector” service
7. Open the operations manager console and navigate to the connectors list which is in the administration tab.
8. Create a subscription for the newly created “Alert Update Connector” (if you don’t see the new connector, refresh the view). The subscription will determine which alerts are processed by the connector.
9. Go to Settings in the administration tab and click on alerts.
10. Create a new resolution state with any name you wish but the ID should be 251. The connector will set the resolution state of all alerts it processes to 251. This is to ensure that the connector does not get any alerts it already processed.

**Troubleshooting the connector:**

In the same package as the connector there is a tracing facility to help trouble-shooting the service when necessary:

1. Copy the TracingGuidsAlertUpdateConnector.txt file to the Tools folder in the OM install directory.
2. Copy the AlertUpdateConnector.tmf file to the Tools\TMF folder in the OM install directory.
3. Follow instructions in the [How to use diagnostic tracing in System Center Operations Manager](http://support.microsoft.com/kb/942864) document.  The output will be in written to the **TracingGuidsAlertUpdateConnector.log** file after the trace data are captured and formatted according to the tracing instruction.

### Frequently asked questions:

1. What should I backup? The only files you should backup are the two configuration files.
2. Is this supported through CSS? At this time the connector is not supported through CSS, the connector is however using fully supported SDK APIs.
3. Does it work with SP1 RC? Yes. The connector will work with OpsMgr RTM as well as SP1 RC.
4. If I update the configuration file using the ConnectorConfiguration.exe utility do I need to restart the connector? You do no need to restart the connector. Every polling interval the connector checks whether the configuration file has changed. If it detects that the configuration file has changed, it will reload the connector.
5. Does the connector configuration file need to be on the same computer as the connector? You do not need to place the connector configuration file (this is the file that contains the info about which alerts to modify) on the same computer as the connector. Only AlertUpdateConnector.exe.config needs to be on the same computer and in the same folder as the connector. The configuration file can be anywhere else that is accessible by the connector.
6. What if I have multiple management groups? If you have multiple management groups, you will need to setup the connector individually for every management group. You can however reuse the connector configuration file between the instances of the connector.
7. What is the impact of the connector on the RMS? The impact should be very minimal as the connector is using a set of APIs that were designed specifically for this purpose.
8. I configured the connector to populate the “WebConsoleURL” but when I open the web console I see an error? This is most likely because you did not add an alert resolution state for a connector. To correct the problem, go to global and add an alert resolution state for ID number 251 and then try using the link again.

<ConnectorConfig GlobalResolutionState="251" **ExcludeResolutionState="255"** >

  <AlertSources>

    <AlertSource Id="b5f11922-44f6-a28c-6f88-cb0e23923ce8" Type="Rule"

       UserName="MyDomain\MyAccount">

      <PropertiesToModify>

        <Property Name="CustomField2" NewValue="AutomationTeam" GroupIdFilter="" />

        <Property Name="CustomField3" NewValue="%SystemDrive%" GroupIdFilter="" />

        <Property Name="ResolutionState" NewValue="251" GroupIdFilter="" />

      </PropertiesToModify>

    </AlertSource>

  </AlertSources>

</ConnectorConfig>

### Functionality added in V2

1. Ability to insert the Web Console Alert URL into a custom field. In order to do this, you need to specify the following value for a custom field: $WebConsoleURL$