

## MuleSoft MMC Action Pack

### Actions

The following actions are available in this action pack:

- Mule MMC Create Deployment
- Mule MMC Delete Deployment
- Mule MMC Get Application ID
- Mule MMC Get Deployment Details
- Mule MMC Get Deployment ID
- Mule MMC Get Servers
- Mule MMC Perform Deployment
- Mule MMC Upload Application

These make use of the MuleSoft MMC REST API, documentation for which can be found here: <https://docs.mulesoft.com/mule-management-console/v/3.6/rest-api-reference>.

Note: there are API differences between the different versions of MMC, but all of the above actions use API's that are consistent between the versions. This action pack was tested on MMC 3.2.2 and 3.6.4.

Some of the actions use the Automic WebService Request form UI, but others make use of curl (which needs to be on the path on the server where the WebService REST agent is installed).

### Deploy Application

This is a typical sequence of actions for deploying an application using MuleSoft MMC:

---



- Mule MMC Upload Application
  - Upload the application artefact to MuleSoft MMC
- Mule MMC Get Servers
  - Retrieve from MuleSoft MMC the list of MuleSoft servers configured in this environment
- Create Server IDs String
  - Format the retrieved array of Servers into a string that can be used in next Webservice call
- Mule MMC Create Deployment
  - Create a deployment of the artefact on the servers
- Mule MMC Perform Deployment
  - Perform the actual deployment

For the 'Create Server IDs String' action I created a script object, with the following code:

```

! Script to convert array of servers retrieved with MULE_MMC_GET_SERVERS
! Incoming array name is expected to be &SERVERS_ARRAY#
:DEFINE &SERVERS_STRING#, string
!:FILL &ARRAY#[ ] = GET_VAR(VARA.STATIC.TEST,"KEY01")
:SET &SERVERS_STRING# =
ARRAY_2_STRING(&SERVERS_ARRAY#[ ],"\","\",FILLED)
:SET &SERVERS_STRING# = STR_CAT("\", &SERVERS_STRING#)
:SET &SERVERS_STRING# = STR_CAT(&SERVERS_STRING#, "\")
:PUBLISH &SERVERS_STRING#,, "WORKFLOW"
  
```

You can then use &SERVERS\_STRING# in the prompt of the Mule MMC Create Deployment action.

## Delete Application

Here is an example of deleting an application, only if it already exists:



- Get Deployment ID
  - Specify the name of the application and get the deployment ID
- Delete Deployment

Put a pre-condition on the Delete Deployment so it only runs if `&DEPLOYMENT_ID_RETRIEVED#<>'"`.