
Actifio Resiliency Director RESTful APIs



Copyright, Trademarks, and other Legal Matter

Copyright © 2021 Google LLC. All rights reserved.

Actifio™, OnVault™, and VDP™ are trademarks of Google LLC.

All other brands, product names, goods and/or services mentioned herein are trademarks or property of their respective owners.

Contents

Preface	ix
Actifio Appliances.....	ix
The Actifio Now Customer Portal.....	ix
Actifio Support Centers.....	ix
Chapter 1 – Actifio Resiliency Director RESTful APIs	1
Using the Resiliency Director RESTful APIs.....	1
<i>Actifio Resiliency Director RESTful APIs</i>	1
Configuring the Actifio Resiliency Director – IRestConfig.....	2
<i>configureRd</i>	2
Accessing the Actifio Resiliency Director – IRestLogin	3
<i>login</i>	3
<i>getRDDetails</i>	4
<i>getServerIdentity</i>	4
<i>getVersion</i>	5
<i>logout</i>	5
<i>regenerateUuid</i>	6
Managing the Actifio Appliances – IRestStorageServer	6
<i>addStorageServer</i>	6
<i>fetchAuthenticationSource</i>	8
<i>fetchEmailServerConfig</i>	9
<i>fetchStorageServers</i>	9
<i>updateStorageServer</i>	11
<i>removeStorageServer</i>	12
Managing Virtual Management Servers – IRestServer.....	12
<i>createServer</i>	12
<i>connectServer</i>	13
<i>getServers</i>	14
<i>updateServer</i>	15
<i>deleteServer</i>	16
<i>fetchserverbyCDS</i>	16
<i>fetchserversbyrecoveryplan</i>	18
Managing the Actifio Resiliency Director Server – IRestArmServer.....	19

<i>getArmServers</i>	19
<i>createArmServer</i>	20
<i>updateArmServer</i>	20
<i>deleteArmServer</i>	21
<i>unRegisterRdDirector</i>	21
<i>updateVersionFlagOnRDServer</i>	22
<i>updateArmServerUuid</i>	23
<i>updateArmServerIpAddress</i>	23
<i>validateRegistration</i>	24
<i>validatePeerRd</i>	24
Managing User Credentials – IRestUserCredentials	26
<i>addnewusercredentials</i>	26
<i>fetchusercredentials</i>	27
<i>editusercredentials</i>	27
<i>deleteusercredentials</i>	28
Managing Application Groups – IRestAppGroup	29
<i>fetchMembersForNewApplicationGroup</i>	30
<i>fetchVMDetailsFor</i>	31
<i>fetchVMDetailsForEdit</i>	34
<i>getApplicationGroupVMCount</i>	38
<i>getVMNICDetails</i>	39
<i>createAppGroup</i>	40
<i>getAppGroups</i>	51
<i>removeAppGroup</i>	61
<i>getVirtualMachines</i>	61
<i>fetchVMAndVappsForAppGroup</i>	64
<i>updateAppGroup</i>	66
<i>fetchAppGroupDetailsWithVms</i>	68
<i>fetchCount</i>	71
<i>getPreDefinedDelay</i>	71
<i>updatePeerAppGroupCount</i>	72
<i>cancelAppGrpCreation</i>	72
<i>updatePeerRecoveryPlanId</i>	73
<i>updatetargetclusterid</i>	73
<i>fetchmappedcds</i>	74
<i>fetchapplicationchildmembers</i>	75
<i>fetchbackupdetails</i>	76
Managing Organizations – IRestOrganization	77
<i>createOrganization</i>	77
<i>deleteOrganizations</i>	78
<i>getCdsOrgs</i>	78
<i>getOrganizations</i>	79
<i>updateOrganization</i>	81
Managing Recovery Plans – IRestRecoveryPlan	82

<i>getRecoveryPlans</i>	83
<i>createRecoveryPlan</i>	84
<i>updateRecoveryPlan</i>	94
<i>deleteRecoveryPlans</i>	98
<i>getRecoveryPlanAppGroup</i>	98
<i>getAppGroupFromPrimary</i>	102
<i>getAppGroupWithNewSequence</i>	105
<i>getRecoveryPlanResourcePools</i>	109
<i>executeRecoveryPlan</i>	110
<i>getRecoveryplanState</i>	110
<i>getAppGroupForEdit</i>	111
<i>getAllStatesForRecoveryPlan</i>	121
<i>resetRecoveryPlanState</i>	121
<i>getRecoveryPlanCountGroupByState</i>	122
<i>getRecoveryPlanSchedule</i>	122
<i>getRecoveryPlanStatesForGivenIds</i>	123
<i>invalidateRecoveryPlan</i>	124
<i>getPortGroupsForRecoveryPlan</i>	125
<i>fetchDedupeAsyncStatus</i>	126
<i>fetchRecoveryPlansByDeDupeAsyncStatus</i>	127
<i>cancelRecoveryPlanExecution</i>	128
<i>disableRecoveryPlan</i>	129
<i>getFetchInvalidRecoveryPlanNotification</i>	129
<i>executeSingleVMRecovery</i>	130
<i>executeSingleApplicationRecovery</i>	130
<i>fetchVmExecutionStatus</i>	131
<i>executeSingleScript</i>	132
<i>fetchVMForInvalidRecoveryPlan</i>	132
<i>fetchCompleteRecoveryPlanData</i>	133
<i>executeonce</i>	135
<i>fetchtargethostnameList</i>	140
Managing Alerts – IRestRoleManager	141
<i>createRole</i>	141
<i>deleteRoles</i>	141
<i>getCDSRoles</i>	142
<i>getRoles</i>	143
<i>updateRole</i>	144
<i>getRights</i>	145
Managing Virtual Management Server Resources– IRestVMSResources	146
<i>createPortGroup</i>	146
<i>createResourcePool</i>	147
<i>deletePortGroup</i>	148
<i>deleteResourcePool</i>	149
<i>discoverPortGroupsForMapping</i>	149
<i>discoverResourcepoolsForMapping</i>	151

<i>fetchHostOrClusterFromServer</i>	152
<i>fetchPortGroupOrgs</i>	152
<i>fetchResourcePoolOrgs</i>	153
<i>getPortGroupsForMapping</i>	153
<i>getResourcePoolsforMapping</i>	155
<i>refreshResourcePool</i>	156
<i>updatePortGroup</i>	157
<i>updateResourcePool</i>	157
Managing Alerts - IRestAlerts	158
<i>getAlerts</i>	158
<i>getLatestAlerts</i>	158
Managing Dashboard - IRestDashboard	160
<i>fetchrecoveryplancount</i>	160
<i>getRecoveryPlanStatesForGivendateRange</i>	161
<i>fetchappcount</i>	163
<i>fetchapplicationgroupcount</i>	163
<i>getreportcountbytype</i>	163
<i>fetchhealthinfo</i>	164
<i>fetchsysteminfo</i>	165
<i>fetchapplicationtypecount</i>	165
Viewing events-IRestEvents	166
<i>getdetailreport</i>	166
Viewing Logs - IRestLogs	170
<i>fetchCDSLogs</i>	170
<i>getBRDLogFile</i>	171
Fetching VM Count - IRestLicense	172
<i>getOrganizationVMCount</i>	172
Fetching Client Properties - IRestClientProperties	174
<i>getIntervalProperties</i>	174
Managing Files - IRestFileManager	175
<i>deleteFileOnServer</i>	175
<i>deleteScriptsByIds</i>	175
<i>fetchFileListOnServer</i>	176
<i>uploadFile</i>	176
<i>getSupportedExternalScript</i>	177
<i>updateScriptHostData</i>	177
Upgrading Resiliency Directory - IRestRDUpgrade	179
<i>fetchUpdateHistory</i>	179
<i>getReleaseNote</i>	180
<i>listRDUpdate</i>	180
<i>removeRDUpdates</i>	180
<i>upgradeRDServer</i>	181

<i>uploadUpdateFile</i>	181
<i>checkUpdateInProgress</i>	182
Additional APIs- IRestMiscellaneous	183
<i>fetchRecoveryTypes</i>	183
<i>fetchAppclassProperties</i>	185
<i>fetchappnamepresentonhost</i>	187
Downloading Help - IRestHelpFileDownload	188
<i>getHelpFile</i>	188
<i>Update Password</i>	188
Managing System	189
Reboot	189
Shutdown	189
Using the Resiliency Director Cloud Recovery RESTful APIs	190
Managing AGM	190
<i>Add AGM</i>	190
<i>List AGM</i>	191
<i>Edit AGM</i>	191
<i>Remove AGM</i>	192
Managing Appliance Groups	193
<i>Add Appliance Group</i>	193
<i>Edit Appliance Group</i>	194
<i>List Appliance Group</i>	196
<i>Remove Appliance Group</i>	198
Managing Clouds	199
<i>Add Cloud</i>	199
<i>Edit Cloud</i>	199
<i>List Cloud</i>	200
<i>Delete Cloud</i>	201
Managing Networks	202
<i>Add Network</i>	202
<i>Edit Network</i>	203
<i>List Network</i>	203
<i>Delete Network</i>	204
<i>Add Appliance to an Appliance Group</i>	205
Managing Application Groups	206
<i>FetchVmDetails</i>	206
<i>Add Application Group to a VM or System state</i>	208
<i>Add Application Group to a VM and CAF Database</i>	210
<i>Update Application Group</i>	215
<i>Remove Application Group</i>	221
Managing Recovery Plans	221

<i>Add Recovery Plan</i>	222
<i>Update Recovery Plan</i>	234
<i>Fetch Recovery Types</i>	241
<i>Remove Recovery Plan</i>	243
<i>Update Password</i>	243
Appendix A: JavaScript Example	245
Appendix B: Java Example	249

Preface

This guide provides step-by-step instructions on how to deploy and use the Actifio Resiliency Director RESTful APIs to add, edit and delete the Actifio appliances, virtual management servers, application groups and so on. It assumes you have read **Getting Started with Actifio Copy Data Management** and have a grasp of the basic concepts associated with an Actifio appliance.

Your Actifio appliance's Documentation Library contains detailed, step-by-step, application-specific instructions on how to protect and access your data. Each guide is in PDF format and may be viewed, downloaded, and printed on demand. The following guides will be of particular interest:

- ***Virtualizing and Protecting Copy Data with the Application Manager***

Actifio Appliances

Unless otherwise specified, all features and functions described in this document apply to both Actifio CDS and Actifio Sky appliances.

The Actifio Now Customer Portal

During the configuration and initialization of your Actifio appliance your Actifio representative provided you with a user name and password for the Actifio Now customer portal.

From the customer portal you can obtain detailed reports about your Actifio appliance as well as search the portal's knowledge base for answers to specific questions.

To log into the Actifio Now customer portal:

1. Go to: <https://now.actifio.com>
2. When prompted, enter the user name and password provided by your Actifio representative.

Actifio Support Centers

To contact an Actifio support representative, you can:

- Send email to: support@actifio.com
- Call:

From anywhere: +1.315.261.7501

US Toll-Free: +1.855.392.6810

Australia: 0011 800-16165656

Germany: 00 800-16165656

New Zealand: 00 800-16165656

UK: 0 800-0155019

1 Actifio Resiliency Director RESTful APIs

The Actifio Resiliency Director RESTful APIs work with JSON data. It has defined its own error code and error messages in the response JSON. Any additional information is included in the body of the return call, JSON-formatted.

Using the Resiliency Director RESTful APIs

Before executing any commands you must open a session by logging into the Actifio Resiliency Director. Login returns a session ID that you need for all subsequent commands. REST APIs issued to the Actifio Resiliency Director are authenticated with a session ID. The session ID is unique for every session and is valid till a configurable time interval.

The Actifio Resiliency Director RESTful APIs require all communications to be in SSL (HTTPS) and UTF-8 encoded. The Actifio Resiliency Director RESTful APIs use network port 443 for connectivity which is configurable.

Actifio Resiliency Director RESTful APIs

The Actifio Resiliency Director Server/Collector RESTful APIs are classified into different categories based on the functionality. Each interface contains respective APIs. Following are the functionality based interfaces for Actifio Resiliency Director Server/Collector:

Note: *sessionid parameter is required as a header parameter for all RESTful APIs, except login, serveridentity, version, and rddetails. However, logout API requires sessionid as query parameter.*

Note: *All the passwords in Actifio Resiliency Director's RESTful APIs must be sent in base 64 encoded format.*

[Configuring the Actifio Resiliency Director - IRestConfig](#) on page 2
[Accessing the Actifio Resiliency Director - IRestLogin](#) on page 3
[Managing the Actifio Appliances - IRestStorageServer](#) on page 6
[Managing Virtual Management Servers - IRestServer](#) on page 12
[Managing the Actifio Resiliency Director Server - IRestArmServer](#) on page 19
[Managing Application Groups - IRestAppGroup](#) on page 29
[Managing Organizations - IRestOrganization](#) on page 77
[Managing Recovery Plans - IRestRecoveryPlan](#) on page 82
[Managing Alerts - IRestRoleManager](#) on page 141
[Managing Virtual Management Server Resources- IRestVMSResources](#) on page 146
[Managing Alerts - IRestAlerts](#) on page 158

[Viewing Logs - IRestLogs](#) on page 170

[Fetching VM Count - IRestLicense](#) on page 172

[Fetching Client Properties - IRestClientProperties](#) on page 174

[Managing Files - IRestFileManager](#) on page 175

[Upgrading Resiliency Directory - IRestRDUpgrade](#) on page 179

[Downloading Help - IRestHelpFileDownload](#) on page 188

Configuring the Actifio Resiliency Director - IRestConfig

This interface and corresponding RESTful API allows you to configure the Resiliency Director Collector or Resiliency Director Server. This interface contains the following RESTful APIs:

[configureRd](#)

Note: {HOST} in the URL implies hostname for Resiliency Director Server or Resiliency Director Collector depending on the deployment.

[configureRd](#)

Use this API to configure the Actifio Resiliency Director before it is functional.

Method: POST

Request

<https://{HOST}/armnew/api/config/config>

Body Param

```
{
  "rdIp": "172.29.10.191",
  "rdName": "localhost.localdom",
  "dns": "192.168.192.10,192.168.225.2",
  "netMask": "255.255.255.0",
  "gateway": "172.29.10.100",
  "ntpServer": "172.29.11.141",
  "timezone": "US/Eastern",
  "rdType": "Primary",
  "password": "password"
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": "ConfigData is updated with RD Server IPAddress 172.29.10.191"
}
```


Parameters	Configuration
configParams	Configuration details for the Actifio Resiliency Director system installation.

Accessing the Actifio Resiliency Director - IRestLogin

This interface and its corresponding RESTful APIs allow you to validate the login details, server identity, Resiliency Director version, and end the active session. This interface contains the following RESTful APIs:

[login](#)
[getRDDetails](#)
[getServerIdentity](#)
[getVersion](#)
[logout](#)
[regenerateUuid](#)

login

Use this API to authenticate the Actifio Resiliency Director user for login.

Method: POST

Request

https://{HOST}/armnew/api/login

Body Param

```
{
  "username": "admin",
  "password": "cGFzc3dvcmQ="
}
```

Note: Password has to be in encoded format.

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "sessionId": "12ca75a4-eb0b-4ca9-b385-ca42e9d43eaf",
    "username": "admin",
    "timestampLong": 0,
    "uniqueid": 0,
    "rights": [
      "Appliance Manage",
      "Appliance View",
      "Application Group Manage",
      "Application Group View",
      "Collector Manage",
      "Collector View",
      "Download RD Logs",
      "External Script View",
      "External Scripts Manage",
      "Organization Manage",
    ]
  }
}
```

```

        "Organization View",
        "Port group Manage",
        "Port group View",
        "Recovery Plan Manage",
        "Recovery Plan Run",
        "Recovery Plan View",
        "Remote Hosts Manage",
        "Remote Hosts View",
        "Reports Run",
        "Resource pool Manage",
        "Resource pool View",
        "Security Manage",
        "Security View",
        "Server Manage",
        "Server View",
        "Upgrade Manage",
        "Upgrade View",
        "Virtual Management Server Manage",
        "Virtual Management Server View"
    ],
    "hasAllOrgs":true,
    "sessionTimeoutInMin":60,
    "timezone":"US/Eastern",
    "admin":true
}
}

```

Parameters	Description
loginParams	Login details of the user.

getRDDetails

Use this API to get the Actifio Resiliency Director details.

Method: GET

Request

https://{HOST}/armnew/api/rddetails

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "installedVersion": "{\"major\":\"6.1\",\"minor\":\"2\",\"createdAt\":\"04/14/2015  
11:48\",\"svnRevision\":\"314\",\"sprintNumber\":\"5\",\"patchNumber\":\"0\"}",
    "rduuid": "ee280065-24a9-488a-887a-72adec2da60e"
  }
}

```

getServerIdentity

Use this API to get the server identity.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/serveridentity

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "serverIdentity": "primary/secondary"
  }
}
```

getVersion

Use this API to get version detail of the Actifio Resiliency Director.

Method: GET

Request

`https://{HOST}:{PORT}/armnew/api/version`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "major": "1.0",
    "minor": "5.0",
    "createdAt": "01/13/2015 20:12",
    "svnRevision": "2056",
    "sprintNumber": "11",
    "patchNumber": "0"
  }
}
```

logout

Use this API to logout from the session. The server exception (`com.actifio.asrv.share.ArmServerException`) is thrown when the session id provided in parameter is not found in the session pool.

Method: GET

Request

`https://{HOST}:{PORT}/armnew/api/logout?sessionid=1234`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
}
```

```
"responseData": ""
}
```

Parameters	Description
sessionId	Session id as a header parameter for which session should be logged-out.

regenerateUuid

This method regenerates the Actifio Resiliency Director UUID and returns the Actifio Resiliency Director detail.

Method: GET

Request

https://{HOST}/armnew/api/regenerateuuid

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "installedVersion": "{\"major\":\"6.1\",\"minor\":\"2\",\"createdAt\":\"04/14/2015 11:48\",\"svnRevision\":\"314\",\"sprintNumber\":\"5\",\"patchNumber\":\"0\"}",
    "rduuid": "ee280065-24a9-488a-887a-72adec2da60e"
  }
}
```

Managing the Actifio Appliances - IRestStorageServer

This interface and its corresponding RESTful APIs allows you to add, list, edit, and delete the Actifio Appliances used by the Actifio Resiliency Director. This interface contains the following RESTful APIs:

- [addStorageServer](#)
- [fetchAuthenticationSource](#)
- [fetchEmailServerConfig](#)
- [fetchStorageServers](#)
- [updateStorageServer](#)
- [removeStorageServer](#)

addStorageServer

Use this API to add an Actifio appliance in the Actifio Resiliency Director database with the parameters provided in the request.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/cds/add (HTTPS)

Body Param

```
{
  "hostOrIp": "lorem.com",
  "username": "user",
  "password": "pwd",
}
```

```

"port": 6969,
"useHttps": false,
"firmwareVersion": "2.1.1",
"listOrganizations": [
  {
    "organizationId": 1101,
    "name": "ALL",
    "description": "ALL organization, all objects belong to ALL organization.",
    "listCDSOrgs": [],
    "createDate": "06/16/2015 03:51:19",
    "modifyDate": "06/16/2015 04:41:51",
    "childOrgs": null,
    "parentOrgId": 3,
    "scheduleEnable": true,
    "email": "user@testemail.com"
  }
],
"model": "Lorem-ipsum-dolor",
"emailConfigured": false,
"isAuthenticationSource": true,
"vendorkey": "vendorKey"
}

```

Response

```

{
  "status": "success",
  "statusCode": "1005",
  "statusMessage": "Actifio Appliance added successfully",
  "responseData": ""
}

```

Parameters	Description
sessionId	Session id as a header parameter.
storageServerToAdd	Actifio appliance that you want to add in the Resiliency Director database.

fetchAuthenticationSource

Use this API to list a storage server (Actifio Appliance) which is selected as an authentication source.

Method: GET

Request

https://{HOST}/armnew/api/cds/getauthenticationsource-- (HTTPS)

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "id": 1111,
    "hostOrIp": "1.1.1.1",
    "username": "aaa",
    "password": "",
    "port": 0,
    "useHttps": false,
    "firmwareVersion": "",
    "model": "",
    "vendorkey": "vendorKey",
    "emailConfigured": false,
    "authenticationSource": true,
    "listOrganizations": [
      {
        "organizationId": 1101,
        "name": "ALL",
        "description": "ALL organization, all objects belong to ALL organization.",
        "listCDSOrgs": null,
        "createDate": "06/16/2015 03:51:19",
        "modifyDate": "06/16/2015 04:41:51",
        "childOrgs": [],
        "parentOrgId": 3,
        "scheduleEnable": true,
        "email": "user@testemail.com"
      }
    ]
  }
}
```

Parameters	Description
sessionid	Session id as a header parameter.

fetchEmailServerConfig

Use this API to configure the email notification settings for the required Actifio Appliance.

Method: POST

Request:

https://{HOST}/armnew/api/cds/getemailserverconfig (HTTPS)

Body Param

```
{
  "hostOrIp": "172.29.10.70",
  "username": "admin",
  "password": "password",
  "vendorkey": "vendorKey"
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": "{\"emailConfigured\":true}"
}
```

Parameters	Description
storageServerData	Actifio Appliance for which you want to check the email server configuration.
sessionid	Session id as a header and body parameters.

fetchStorageServers

Use this API to fetch the list of available Actifio Appliance(s) in the Resiliency Director database. If the method parameter is null, the method returns all the Actifio Appliances from the database. If the method parameter is not null, the method returns only the Actifio appliance(s) whose id(s) are provided as an input.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/cds/fetch - To fetch all Actifio Appliance(s) from database (HTTPS).

https://{HOST}:{PORT}/armnew/api/cds/fetch?ids=1,2,3 - To fetch Actifio Appliance(s) by id(s) (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 31746,
      "hostOrIp": "172.27.46.45",
      "username": "admin",
      "password": "",
      "port": 0,
    }
  ]
}
```

```

        "useHttps":false,
        "firmwareVersion":"",
        "model":"",
        "vendorkey":"vendorKey",
        "clustername":"rdsources",
        "version":"10.0.0.2957",
        "certificate":"Cpu(s): 0.3%us, 0.1%sy, 0.0%ni, 99.5%id, 0.1%wa, 0.0%hi, 0.0%si,
0.0%stMem: 8061188k total, 2744536k used, 5316652k free, 567708k buffers",
        "listApplianceGrps":[

        ],
        "status":"Online",
        "listOrganizations":[

        ],
        "clusterId":145389359335,
        "selected":false,
        "emailConfigured":false,
        "authenticationSource":true
    },
    {
        "id":74781,
        "hostOrIp":"172.27.46.46",
        "username":"admin",
        "password":"",
        "port":0,
        "useHttps":false,
        "firmwareVersion":"",
        "model":"",
        "vendorkey":"vendorKey",
        "clustername":"RDTargetSky",
        "version":"10.0.1.3600",
        "listApplianceGrps":[
            {
                "appliancegroupid":5245,
                "appliancegroupname":"AGGrp1"
            }
        ],
        "status":"Online",
        "listOrganizations":[

        ],
        "clusterId":142966846547,
        "selected":false,
        "emailConfigured":false,
        "authenticationSource":false
    }
]
}

```

Parameters	Description
storageServerIdsToFetch	Comma separated Actifio Appliance id(s) for which you want to fetch the data.
sessionId	Session id as a header parameter.

updateStorageServer

Use this API to edit an existing Actifio Appliance details with the parameters provided in the request.

Method: POST

Request

https://{HOST}/armnew/api/cds/update -- (HTTPS)

Body Param

```
{
  "id": 1,
  "hostOrIp": "lorem.com",
  "username": "user",
  "password": "pwd",
  "port": 6969,
  "useHttps": false,
  "firmwareVersion": "2.1.1",
  "listOrganizations": [
    {
      "organizationId": 1101,
      "name": "ALL",
      "description": "ALL organization, all objects belong to ALL organization.",
      "listCDSOrgs": null,
      "createDate": "06/16/2015 03:51:19",
      "modifyDate": "06/16/2015 04:41:51",
      "childOrgs": [],
      "parentOrgId": 3,
      "scheduleEnable": true,
      "email": "user@testemail.com"
    }
  ],
  "model": "Lorem-ipsum-dolor",
  "emailConfigured": false,
  "isAuthenticationSource": true,
  "vendorkey": "vendorKey"
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}
```

Parameters	Description
storageServerToUpdate	Body parameters of the Actifio Appliance that you want to update in the Resiliency Director database.
sessionId	Session id as a header parameter.

Note: No need to send password if it is not changed.

removeStorageServer

Use this API to delete the Actifio Appliance from the database with the parameters provided in the request.

Method: GET

Request

https://{HOST}/armnew/api/cds/remove?ids=1,2,3 (HTTPS)

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}
```

Parameters	Description
storageServerIdsForDelete	Comma separated id(s) of the Actifio Appliance(s) that you want to delete.

Managing Virtual Management Servers - IRestServer

This interface and its corresponding RESTful APIs allow you to add, list, edit, and delete the virtual management server used by the Actifio Resiliency Director. This interface contains the following RESTful APIs:

- [createServer](#)
- [connectServer](#)
- [getServers](#)
- [updateServer](#)
- [deleteServer](#)
- [fetchserverbyCDS](#)
- [fetchserverbyCDS](#)
- [fetchserversbyrecoveryplan](#)

createServer

Use this API to add a virtual management server in the Actifio Resiliency Director database with the parameters provided in the request.

Method: POST

Request

https://{HOST}/armnew/api/server/add (HTTPS)

Body Param

```
{
  "hostOrIp": "172.29.10.164",
  "username": "acme",
  "password": "acme",
  "listOrganizations": [
    {
      "id": 5146,
      "name": "neworg",
      "description": null,

```

```

        "createdate": null,
        "modifydate": null,
        "email": null,
        "scheduleEnable": false,
        "parentOrgId": 0,
        "listCDSOrgs": null,
        "childOrgs": null
    }
},
"release": "1.1",
"port": 0,
"discovered": true,
"deploymentPlatform": "platform",
"noOfRecoverPlan": 0
}

```

Response

```

{
  "status": "success",
  "statusCode": "1002",
  "statusMessage": "Virtual management server added successfully.",
  "responseData": ""
}

```

Parameters	Description
serverToAdd	Body parameters of a virtual management server that you want to add in the Resiliency Director database.

connectServer

Use this API to test the connection to the virtual management server.

Method: POST

Request

https://{HOST}/armnew/api/server/connect

Body Param

```

{
  "hostOrIp": "172.29.11.244",
  "username": "test",
  "password": "test123",
  "port": "443"
}

```

Response

```

{
  "status": "success",
  "statusCode": "10416",
  "statusMessage": "Successfully connected to vcenter.",
  "responseData": ""
}

```

Parameters	Description
sessionId	Session id as a header parameter.

getServers

Use this API to list the available virtual management servers in the Actifio Resiliency Director database. If the method parameter is null, the method returns all the available virtual management servers from database. If the method parameter is not null, the method returns only the virtual management server(s) whose id(s) are provided as an input.

Method: GET

Request

`https://{HOST}/armnew/api/server/fetch` - To fetch all servers from database (HTTPS).

`https://{HOST}/armnew/api/server/fetch?ids=1,2,3` - To fetch virtual management servers by id(s) (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 5218,
      "hostOrIp": "172.29.11.244",
      "username": "bikas",
      "password": "",
      "port": 443,
      "discovered": true,
      "noOfRecoverPlan": 0,
      "listOrganizations": [
      ],
      "serverUuid": "47bc8446-eae8-4063-aae5-efd63a639477",
      "status": "Online",
      "selected": false
    },
    {
      "id": 18475,
      "hostOrIp": "172.29.11.185",
      "username": "root",
      "password": "",
      "port": 443,
      "discovered": true,
      "noOfRecoverPlan": 0,
      "listOrganizations": [
      ],
      "serverUuid": "EA100FD6-F127-45FE-A22F-488C97E9CE6A",
      "status": "Online",
      "selected": false
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.

Parameters	Description
ids	Comma separated virtual management server id(s) for which you want to fetch the data.

updateServer

Use this API to edit an existing virtual management server details in the Actifio Resiliency Director database with the parameters provided in the request.

Method: POST

Request

https://{HOST}/armnew/api/server/update (HTTPS)

Body Param

```
{
  "id": 1,
  "hostOrIp": "172.29.12.84",
  "username": "root",
  "password": "password",
  "port": 26,
  "release": "1.0.0",
  "deploymentPlatform": "vmware",
  "noOfRecoverPlan": 0,
  "discovered": true,
  "listOrganizations": [
    {
      "id": 5146,
      "name": "neworg",
      "description": null,
      "createdate": null,
      "modifydate": null,
      "email": null,
      "scheduleEnable": false,
      "parentOrgId": 0,
      "listCDSOrgs": null,
      "childOrgs": null
    }
  ]
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}
```

Parameters	Description
serversToUpdate	Body parameters of the virtual management server that you want to update in the Actifio Resiliency Director database.

Note: No need to send password if it is not changed.

deleteServer

Use this API to delete the virtual management server from the Resiliency Director database.

Method: GET

Request

`https://{HOST}/armnew/api/server/remove?ids=1,2,3 (HTTPS)`

Response

```
{ "status": "success", "statusCode": "1000", "statusMessage": "Operation performed successfully", "responseData": "" }
```

Parameters	Description
sessionid	Session id as a header parameter.
ids	Comma separated id(s) of the virtual management server(s) that you want to delete.

fetchserverbyCDS

Use this API to retrieve the commonly registered VCenter among a given Actifio CDS ids.

Method: GET

Request

`https://{HOST}/armnew/api/server/fetchserverbycds?ids=21605,21607`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 21598,
      "hostOrIp": "172.29.11.185",
      "username": "root",
      "password": "",
      "port": 0,
      "release": "1.1",
      "discovered": true,
      "deploymentPlatform": "platform",
      "noOfRecoverPlan": 0,
      "listOrganizations": [],
      "selected": false
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
cdsId	CDS Ids

fetchserversbyrecoveryplan

Use this API to retrieve the mapped Virtual Management Servers for a recovery plan.

Method: GET

Request

https://{HOST}/armnew/api/server/
fetchserversbyrecoveryplan?recoveryplanid=44085&appgroupids=21626&_=1487065017842

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 5138,
      "hostOrIp": "172.29.10.179",
      "username": "root",
      "password": "",
      "port": 0,
      "release": "1.1",
      "discovered": true,
      "deploymentPlatform": "platform",
      "noOfRecoverPlan": 0,
      "listOrganizations": [
        {
          "id": 5140,
          "name": "calsoft",
          "description": null,
          "createdate": null,
          "modifydate": null,
          "email": "user@testemail.com"
        }
      ]
    }
  ]
}
```

Parameters	Description
appgroupids	List of Application group ids mapped to recovery plan.
recoveryplanid	Id of the recovery plan.
sessionid	Session id as a header parameter.

Managing the Actifio Resiliency Director Server – IRestArmServer

This interface and its corresponding RESTful APIs allow you to execute REST requests to CSP (Cloud Service Provider) for Actifio Resiliency Director Server. This interface contains the following RESTful APIs:

- [getArmServers](#)
- [createArmServer](#)
- [updateArmServer](#)
- [deleteArmServer](#)
- [unRegisterRdDirector](#)
- [updateVersionFlagOnRDServer](#)
- [updateArmServerUuid](#)
- [updateArmServerIpAddress](#)
- [validateRegistration](#)
- [validatePeerRd](#)

getArmServers

Use this API to fetch the Resiliency Director Server(s) managed by CSP (Cloud Service Provider) site. If the method parameter is null, the method returns all the available Resiliency Director Servers from database. If the method parameter is not null, the method returns only the Resiliency Director Server(s) whose id(s) have been provided as an input.

Method: GET

Request:

<https://{HOST}/armnew/api/armserver/fetch> - To fetch all Resiliency Director server from database (HTTPS).

<https://{HOST}/armnew/api/armserver/fetch?ids=1,2,3> - To fetch Resiliency Director server by id(s) (HTTPS).

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "hostOrIp": "172.16.10.195",
      "description": "",
      "customerId": 11273,
      "customerName": "custSag",
      "userName": "admin",
      "password": "Password",
      "port": 443,
      "peerUniqueId": null,
      "armServerId": 11275,
      "createdOn": 1421190229514
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.

Parameters	Description
ids	Customer ids – list of customer id(s) whose associated Resiliency Director Server data is to be fetched.

createArmServer

Use this API to create a Resiliency Director Collector in CSP database with the parameters provided in request, and also to add Resiliency Director Server to given Resiliency Director Collector database.

Method: POST

Request

https://{HOST}/armnew/api/armserver/add (HTTPS)

Body Param

```
{
  "hostOrIp": "172.16.10.202",
  "description": "Server description",
  "customerId": "11273",
  "userName": "admin",
  "password": "Password",
  "port": "443"
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}
```

Parameters	Description
session	Session id as a header parameter.
rdServerToAdd	Resiliency Director Server that you want to add in CSP database.

updateArmServer

Use this API to update the Actifio Resiliency Director Server in CSP database with the parameters provided in request.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/armserver/update (HTTPS)

Body Param

```
{
  "id": "11256",
  "hostOrIp": "172.16.10.202",
  "description": "ss",
  "customerId": "11273",
  "userName": "admin",

```

```

    "password": "Password",
    "port": "443"
}

```

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}

```

Parameters	Description
armserverToUpdate	Resiliency Director Server that you want to update in the database.
session	Session id as a header parameter.

deleteArmServer

Use this API to delete a Resiliency Director Server(s) from the CSP database with the parameters provided in request.

Method: GET

Request:

https://{HOST}:{PORT}/armnew/api/armserver/remove?ids=1,2,3

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}

```

Parameters	Description
ids	Resiliency Director server id(s) that you want to delete.
session	Session id as a header parameter.

unRegisterRdDirector

Use this API to unregister the Resiliency Director Server from the database.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/armserver/unRegisterRdDirector (HTTPS)

Body Param

```

{
  "id": 1,
  "hostOrIp": "172.16.10.202",
  "description": "RD Server description",
}

```

```

"customerId": "11273",
"userName": "admin",
"password": "",
"port": "443"
}

```

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}

```

Parameters	Description
rdServerToUnRegister	Resiliency Director server details that you want to delete from the Resiliency Director Collector database.
session	Session id as a header parameter.

updateVersionFlagOnRDServer

Use this API to update the Resiliency Director version flag on peer Resiliency Director.

Method: POST

Request

<https://{HOST}:{PORT}/armnew/api/armserver/validaterdversionflag> (HTTPS)

Body Param

```

{
  "id": "11256",
  "hostOrIp": "172.16.10.202",
  "description": "ss",
  "customerId": "11273",
  "userName": "admin",
  "password": "Password",
  "port": "443"
}

```

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}

```

Parameters	Description
sessionId	Session id as a header parameter.
rdServerToUpdate	Resiliency Director Server details that you want to update in the database.

updateArmServerUuid

Use this API to update the Resiliency Director version flag on peer Resiliency Director.

Method: POST

Request

`https://{HOST}:{PORT}/armnew/api/armserver/updaterduuid?rduuid="5036e87c-70ab-99e4-57d7-4feba7716d89"1 && hostorip="172.29.11.152" - (HTTPS)`

Body Param

```
{}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "rduuid": 11256,
    "hostorip": "172.16.10.202"
  }
}
```

Parameters	Description
rduuid	UUID of the arm server which need to be updated.
hostorip	Host or IP of the arm server.
sessionid	Session id as a header parameter.

updateArmServerIpAddress

Use this API to update the IP address of the Resiliency Director appliance on the registered peer Resiliency Director appliance.

Method: POST

Request:

`https://{HOST}:{PORT}/armnew/api/armserver/updaterdip?rduuid="38f271ea-9fe8-43ef-85bf-fccd48f6c116" && newipaddress="172.16.10.203"`

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionid	Session id as a header parameter.

Parameters	Description
rduuid	UUID of the Resiliency Director appliance.
newipaddress	Modified IP address of the Resiliency Director appliance.

validateRegistration

Use this API to validate the registration of Resiliency Director collector using an IP address.

Method: GET

Request

`https://{HOST}/armnew/api/armserver/validateRegistration?ip=172.29.10.123 -(HTTPS)`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "hostOrIp": "172.29.11.108",
    "description": "",
    "listOrganizations": null,
    "organizationList": null,
    "userName": "admin",
    "password": "",
    "port": 0,
    "peerUniqueId": null,
    "invalidRdVersion": false,
    "rdVersion": "6.2.0.804",
    "createdOn": "10/14/2015 05:32:05",
    "armServerId": 8305
  }
}
```

Parameters	Description
IP	IP address of Resiliency Director Collector.
sessionId	Session id as a header parameter.

validatePeerRd

Use this API to validate peer Resiliency Director version compatibility and send its own version details in response.

Method: POST

Request

`https://{HOST}/armnew/api/validatePeerRd (HTTPS)`

Response

```
{
  "status": "success",
  "statusCode": "1000",
```

```

"statusMessage": "Operation performed successfully",
"responseData": {
  "versionTO": {
    "major": "1.0",
    "minor": "5.0",
    "createdAt": "01/13/2015 20:12",
    "svnRevision": "2056",
    "sprintNumber": "11",
    "patchNumber": "0"
  },
  "ipAddress": "172.29.11.145",
  "rduuid": "ee280065-24a9-488a-887a-72adec2da60e"
}
}

```

Parameters	Description
sessionId	Session id as a header parameter.
peerRDVersionWithIPTO	Resiliency Director version compatibility details.

Managing User Credentials – IRestUserCredentials

This interface and its corresponding APIs allows you to execute REST request to Resiliency Director for creating and managing user credentials of virtual machines. This interface contains the following RESTful APIs:

[addnewusercredentials](#)
[fetchusercredentials](#)
[editusercredentials](#)
[deleteusercredentials](#)

addnewusercredentials

Use this API to create new user credentials for virtual machines in the Resiliency Director Server database. The request must be in the JSON format. You can use these credentials while creating application groups and recovery plans.

Method: POST

Request

https://{HOST}/armnew/api/usercredentials/add

Body Param

```
{
  "name": "WindowsCredentials",
  "username": "testuser",
  "password": "test123"
}
```

Response

```
{
  "status": "success",
  "statusCode": "10415",
  "statusMessage": "UserCredentials added successfully",
  "responseData": "5249"
}
```

Parameters	Description
sessionid	Session id as a header parameter.
name	Name for the user credentials to be created.
username	Desired username.
password	Password for the user credentials.

fetchusercredentials

Use this API to get the list of available user credentials in the Actifio Resiliency Director Server database with the parameters provided in request.

Method: GET

Request

https://{HOST}/armnew/api/usercredentials/fetch?_=1549009276055

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 5239,
      "name": "Win",
      "username": "Administrator",
      "password": "",
      "systemid": "30ec0016-3be6-4f15-9d9f-3015283009b1",
      "peercredentialsid": 8352
    },
    {
      "id": 5237,
      "name": "Windows",
      "username": "Administrator",
      "password": "",
      "systemid": "30ec0016-3be6-4f15-9d9f-3015283009b1",
      "peercredentialsid": 8408
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.

editusercredentials

Use this API to modify the existing user credentials for virtual machines in the Resiliency Director Server database. The request must be in the JSON format. You can use these credentials while creating application groups and recovery plans.

Method: POST

Request

https://{HOST}/armnew/api/usercredentials/update

Body Param

```
{
  "id": 5249,
  "name": "WindowsCredentialsNew",
  "username": "testuser",
  "password": "test123",
  "peercredentialsid": 6178,
  "systemid": "4eae1996-a68b-412d-b5f8-33088e20758d"
}
```

```
}
```

Response

```
{
  "status": "success",
  "statusCode": "10417",
  "statusMessage": "User Credentials updated successfully",
  "responseData": ""
}
```

Parameters	Description
sessionId	Session id as a header parameter.
id	Unique ID of the user credentials.
name	Name for the user credentials to be created.
username	Desired username.
password	Password for the user credentials.

deleteusercredentials

Use this API to remove the existing user credentials of the virtual machines from the Actifio Resiliency Director Server database.

Method: GET

Request

https://{HOST}/armnew/api/usercredentials/remove?ids=5249&_=1548216693296

Response

```
{
  "status": "success",
  "statusCode": "10418",
  "statusMessage": "User Credentials deleted successfully",
  "responseData": ""
}
```

Parameters	Description
sessionId	Session id as a header parameter.
id	ID of the user credentials that you want to delete.

Managing Application Groups – IRestAppGroup

This interface and its corresponding APIs allows you to execute REST requests to Resiliency Director Server for application groups. This interface contains the following RESTful APIs:

- fetchMembersForNewApplicationGroup
- fetchVMDetailsFor
- fetchVMDetailsForEdit
- getApplicationGroupVMCount
- getVMNICDetails
- createAppGroup
- getAppGroups
- removeAppGroup
- getVirtualMachines
- updateAppGroup
- fetchAppGroupDetailsWithVms
- getPreDefinedDelay
- updatePeerAppGroupCount
- cancelAppGrpCreation
- updatePeerRecoveryPlanId
- updatetargetclusterid
- fetchmappedcds
- fetchapplicationchildmembers
- fetchbackupdetails

fetchMembersForNewApplicationGroup

Use this API to fetch application group members (VMs and vApps) for given Actifio Appliance and vCenter server.

Method: GET

Request

https://{HOST}/armnew/api/applicationgroup/
fetchnewmember?cdsid=5205&serverid=&appgroupname=AGSqlDB&recoverytype=Recover SQL Server
Database&agmid=124567&uniqueKey=1548925640655&_id=1548925458719

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "memberUuId": "msdb",
      "memberName": "msdb",
      "uniqueGridIndex": "1549270662539",
      "vm": false,
      "executionStatus": "",
      "cdsAppId": "6330",
      "cdsClusterId": "1415039054",
      "targetClusterId": "0",
      "clustername": "RD_HYD_SRC",
      "validForAppGroup": false,
      "displayAppName": "msdb (win2012_sql_new) ",
      "displayAppType": "SQL Database",
      "apptype": "SqlServerWriter",
      "sourceHostName": "win2012_sql_new",
      "recoveryType": {
        "recoveryTypeId": 4,
        "name": "Recover SQL Server Database",
        "recoveryTypeOrder": "2"
      },
      "infoRequired": false,
      "vmalreadyPresentInRecPlan": false
    },
    {
      "memberUuId": "WINVVM",
      "memberName": "WINVVM",
      "uniqueGridIndex": "1549270662540",
      "vm": false,
      "executionStatus": "",
      "cdsAppId": "231634",
      "cdsClusterId": "1415039054",
      "targetClusterId": "0",
      "clustername": "RD_HYD_SRC",
      "validForAppGroup": false,
      "displayAppName": "WINVVM (win2012r2-1) ",
      "displayAppType": "SQL Instance",
      "apptype": "SqlInstance",
      "sourceHostName": "win2012r2-1",
      "recoveryType": {
        "recoveryTypeId": 4,
        "name": "Recover SQL Server Database",
        "recoveryTypeOrder": "2"
      },
    },
  ]
}
```

```

        "infoRequired": false,
        "vmAlreadyPresentInRecPlan": false
    }
}
}

```

Parameters	Description
sessionId	Session id as a header parameter.
cdsId	Actifio Appliance id required to create the application group. The <i>cdsId</i> parameter must have a copy data storage in the Resiliency Director database.
serverId	Virtual management server id required to create a application group. The <i>serverId</i> parameter must have a server in the Resiliency Director database.
agmid	Actifio Appliance AGM id is required to create application group for cloud.
appgroupName	Name assigned to the application group.
recoverytype	Types of recovery option. Values for recovery option: - Recover VM From VMware VM Backup - Create new VM from VMware VM - Recover Oracle Database - Recover SQL Server Database - Recover File System Note: The response may vary based on the selected recovery type.
uniqueKey	Value it can be time stamp too.

fetchVMDetailsFor

Use this API to get details of the application group member. A list for application group members with their order in application group. Return details for members with same order pass as input.

Method: POST

Note: This API has POST method because GET method does not accepts body parameter as input.

Request

https://{HOST}/armnew/api/applicationgroup/
 fetchvmdetails?serverId=5207&uniqueKey=1548926123562&appgroupName=AG

Body Param

```
{
  "memberOrderIndex": 1,
  "delay": 0,
  "vmMembers": [
    {
      "memberUuId": "502ab92d-7e57-8f43-891c-07a07d85e429",
      "memberName": "Rhel1_1549271228018",
      "uniqueGridIndex": "1549270918148_1549271228018",
      "vm": true,
      "executionStatus": "",
      "cdsAppId": "106809",
      "cdsClusterId": "1415039054",
      "targetClusterId": "0",
      "clustername": "RD_HYD_SRC",
      "validForAppGroup": false,
      "displayAppName": "Rhel1_1549271228018",
      "displayAppType": "VM",
      "apptype": "VM",
      "sourceHostName": "rhel1",
      "recoveryType": {
        "recoveryTypeId": 2,
        "name": "Create new VM from VMware VM",
        "recoveryTypeOrder": "1"
      },
      "infoRequired": false,
      "vmalreadyPresentInRecPlan": false,
      "sequence": "1"
    }
  ],
  "appMembers": []
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "vmMembers": [
        {
          "memberUuId": "502ab92d-7e57-8f43-891c-07a07d85e429",
          "memberName": "Rhel1_1549271228018",
          "vm": true,
          "executionStatus": "",
          "cdsAppId": "106809",
          "cdsClusterId": "1415039054",
          "targetClusterId": "0",
          "validForAppGroup": true,
          "displayAppName": "Rhel1_1549271228018",
          "displayAppType": "VM",
          "apptype": "VM",
          "recoveryType": {
            "recoveryTypeId": 2,
            "name": "Create new VM from VMware VM",
            "recoveryTypeOrder": "1"
          },
          },
        ]
      }
    ]
  }
```

```

"infoRequired": false,
"id": 0,
"vappName": "",
"numberOfCpus": 1,
"actualNumberOfNics": 1,
"testNumberOfNics": 1,
"testMemoryInMbs": 2048,
"actualMemoryInMbs": 2048,
"storageLocation": "[datastore1tb] RHEL1/RHEL1.vmx",
"shutdownOrder": 0,
"keyValue": "vm-16635",
"numberOfVirtualDisks": 2,
"hddInGbs": 22,
"hostName": "localhost.localdomain",
"reasonForInvalidVM": "",
"cpuResourceAllocation": {
  "reservation": 0,
  "limit": -1,
  "shares": 1000,
  "sharesLevel": 1,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"memoryResourceAllocation": {
  "reservation": 0,
  "limit": -1,
  "shares": 20480,
  "sharesLevel": 1,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"testCPUAllocation": 0,
"actualCPUAllocation": 0,
"prescript": "",
"postscript": "",
"username": "",
"passwd": "",
"nicNames": [
  {
    "deviceConfigId": 4000,
    "macAddress": "00:50:56:aa:b4:2c",
    "nicType": "VirtualVmxnet3",
    "interfaceLabel": "Network adapter 1",
    "pciSlotNumber": 192
  }
],
"vmAllNicDataList": [],
"sourceIp": "172.29.10.210",
"sourceSubnet": "",
"sourceGateway": "172.29.10.100",
"osName": "linux",
"coresPerSocket": 1,
"testCoresPerSocket": 1,
"virtualSocket": 1,
"testVirtualSocket": 1,
"sequenceNumber": 0,
"critical": false,
"vappStartupOrder": 0,
"vmAlreadyPresentInRecPlan": false
}

```

```

    ],
    "appMembers": []
  }
]
}

```

Parameters	Description
sessionId	Session id as a header parameter.
serverId	Server ID in which these members are located. The <i>serverId</i> parameter must have a server in the Resiliency Director database.
appgroupname	Application group name.
uniqueKey	Value which is passed in fetch new member request.
agmid	Actifio Appliance AGM id is required for cloud.
appGroupMembersForDetails	Member list (uuid, name, isvm) for which you want to fetch the details.

fetchVMDetailsForEdit

Use this API to edit the details of an existing application group. A list for application group members with their order in application group. Return details for members with same order pass as input.

Method:POST

Note: This API has POST method because GET method does not accept body parameter as input.

Request

https://{HOST}/armnew/api/applicationgroup/
fetchvmdetailsforedit?serverid=5207&uniqueKey=1549271647135&appgroupname=AG&appGroupId=8311

Body Param

```

[
  {
    "memberOrderIndex": 1,
    "delay": 0,
    "vmMembers": [
      {
        "memberUuId": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",
        "memberName": "Win2012r2-2_1548932687404",
        "uniqueGridIndex": "1549271632422",
        "vm": true,
        "executionStatus": "",
        "cdsAppId": "231511",
        "cdsClusterId": "1415039054",
        "targetClusterId": "1415040293",
        "clustername": "RD_HYD_SRC",
        "validForAppGroup": false,
        "displayAppName": "Win2012r2-2_1548932687404",
        "displayAppType": "VM",

```



```

"apptype": "VM",
"recoveryType": {
  "recoveryTypeId": 2,
  "name": "Create new VM from VMware VM",
  "recoveryTypeOrder": "1"
},
"infoRequired": true,
"id": 0,
"vappName": "",
"numberOfCpus": 0,
"actualNumberOfNics": 0,
"testNumberOfNics": 0,
"testMemoryInMbs": 0,
"actualMemoryInMbs": 0,
"shutdownOrder": 0,
"numberOfVirtualDisks": 0,
"hddInGbs": 0,
"hostName": "",
"ip": "",
"cupResourceAllocation": {
  "reservation": 0,
  "limit": 0,
  "shares": 0,
  "sharesLevel": 0,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"memoryResourceAllocation": {
  "reservation": 0,
  "limit": 0,
  "shares": 0,
  "sharesLevel": 0,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"testCPUAllocation": 0,
"actualCPUAllocation": 0,
"prescript": "",
"postscript": "",
"username": "",
"passwd": "",
"nicNames": [],
"nicForStaticIp": "",
"nicForDns": "",
"defaultDnsIpToconfigure": [],
"sourceIp": "",
"portGroup": [],
"osName": "",
"preExternalScript": [],
"postExternalScript": [],
"coresPerSocket": 0,
"testCoresPerSocket": 0,
"virtualSocket": 0,
"testVirtualSocket": 0,
"sequenceNumber": 0,
"critical": false,
"vappStartupOrder": 0,
"vmalreadyPresentInRecPlan": false,
"sequence": 1
}

```

```

    ],
    "appMembers": []
  }
]

```

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "vmMembers": [
        {
          "memberUuId": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",
          "memberName": "Win2012r2-2_1548932687404",
          "vm": true,
          "executionStatus": "",
          "cdsAppId": "231511",
          "cdsClusterId": "1415039054",
          "targetClusterId": "1415040293",
          "validForAppGroup": true,
          "displayAppName": "Win2012r2-2_1548932687404",
          "displayAppType": "VM",
          "apptype": "VM",
          "recoveryType": {
            "recoveryTypeId": 2,
            "name": "Create new VM from VMware VM",
            "recoveryTypeOrder": "1"
          },
          "infoRequired": false,
          "id": 0,
          "vappName": "",
          "numberOfCpus": 4,
          "actualNumberOfNics": 1,
          "testNumberOfNics": 1,
          "testMemoryInMbs": 4096,
          "actualMemoryInMbs": 4096,
          "storageLocation": "[datastore1tb] Win2012r2-2/Win2012r2-2.vmx",
          "shutdownOrder": 0,
          "keyValue": "vm-21579",
          "numberOfVirtualDisks": 1,
          "hddInGbs": 40,
          "hostName": "winvwm",
          "ip": "",
          "reasonForInvalidVM": "",
          "cupResourceAllocation": {
            "reservation": 0,
            "limit": -1,
            "shares": 4000,
            "sharesLevel": 1,
            "numberOfCpus": 0,
            "corePerSocket": 0
          },
          "memoryResourceAllocation": {
            "reservation": 0,
            "limit": -1,
            "shares": 40960,

```

```

        "sharesLevel": 1,
        "numberOfCpus": 0,
        "corePerSocket": 0
    },
    "testCPUALlocation": 0,
    "actualCPUALlocation": 0,
    "prescript": "",
    "postscript": "",
    "username": "",
    "passwd": "",
    "nicNames": [
        {
            "deviceConfigId": 4000,
            "macAddress": "00:50:56:aa:e1:80",
            "nicType": "VirtualVmxnet3",
            "interfaceLabel": "Network adapter 1",
            "pciSlotNumber": 192
        }
    ],
    "vmAllNicDataList": [
        {
            "deviceConfigId": "4000",
            "interfaceLabel": "Network adapter 1",
            "enforcedStaticIp": false,
            "enforcedStaticDnsIp": false,
            "gateway": "",
            "macAddress": "00:50:56:aa:e1:80",
            "nicType": "VirtualVmxnet3",
            "pciSlotNumber": 192
        }
    ],
    "nicForStaticIp": "",
    "nicForDns": "",
    "defaultDnsIpToconfigure": [],
    "sourceIp": "172.29.10.145",
    "sourceSubnet": "",
    "sourceGateway": "172.29.10.100",
    "osName": "windows",
    "coresPerSocket": 1,
    "testCoresPerSocket": 1,
    "virtualSocket": 4,
    "testVirtualSocket": 4,
    "sequenceNumber": 0,
    "critical": false,
    "vappStartupOrder": 0,
    "vmalreadyPresentInRecPlan": false
    }
    ],
    "appMembers": []
}
]
}

```

Parameters	Description
session	Session id as a header parameter.

Parameters	Description
serverId	Server id on which these members are located. There should be a server with this id in Resiliency Director database.
appgroupname	Application group name.
uniqueKey	Value which is passed in fetch new members request.
appGroupMembersForDetails	Member (include uuid, name, isvm) list for which details needs to be fetched.
appgroupid	Application group ID.

getApplicationGroupVMCount

Use this API to get the details of the application group which is not included in the recovery plan.

Method: GET

Request

<https://{HOST}/armnew/api/applicationgroup/getappgroupvmcount>

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "",
  "responseData": [
    {
      "appGroupName": "appGroup1",
      "appGroupVMCount": 3
    },
    {
      "appGroupName": "appGroup2",
      "appGroupVMCount": 1
    }
  ]
}
```

Parameters	Description
sessionId	Session id as a header parameter.

getVMNICDetails

Use this API to get the network interface card details of a Virtual Machine.

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/vmnicdetails

Body Param

```
{
  "vmserverid": "5209",
  "vmname": "Win2012r2-2",
  "uuid": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",
  "username": "",
  "password": "",
  "usercredentialsid": "5288"
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "ipType": "dhcp",
      "deviceConfigId": "4000",
      "interfaceLabel": "Network adapter 1",
      "enforcedStaticIp": false,
      "enforcedStaticDnsIp": false,
      "ipAddressToConfigure": "172.29.10.145",
      "gateway": "172.29.10.100",
      "subnetMask": "255.255.255.0",
      "macAddress": "00:50:56:AA:E1:80",
      "nicType": "VirtualVmxnet3",
      "pciSlotNumber": 192
    }
  ]
}
```

Parameters	Description
sessionId	Session id as a header parameter.

createAppGroup

Use this API to create an application group with VM details in Resiliency Director database.

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/
add?uniqueKey=1549272668990&appgroupname=AGVM&suppressWarningNicValidation=true

Body Param (For VM)

```
{
  "cdsId": 5125,
  "description": "",
  "serverId": 5127,
  "name": "AGVM",
  "sequenceOrder": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "vmMembers": [
        {
          "memberUuId": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",
          "memberName": "Win2012r2-2",
          "vm": true,
          "executionStatus": "",
          "cdsAppId": "231511",
          "cdsClusterId": "1415039054",
          "targetClusterId": "0",
          "validForAppGroup": true,
          "displayAppName": "Win2012r2-2",
          "displayAppType": "VM",
          "apptype": "VM",
          "recoveryType": {
            "recoveryTypeId": 1,
            "name": "Recover VM From VMWare VM Backup",
            "recoveryTypeOrder": "1"
          },
          "infoRequired": false,
          "id": 0,
          "vappName": "",
          "numberOfCpus": 4,
          "actualNumberOfNics": 1,
          "testNumberOfNics": 1,
          "testMemoryInMbs": "4096",
          "actualMemoryInMbs": 4096,
          "storageLocation": "[datastore1tb] Win2012r2-2/Win2012r2-2.vmx",
          "shutdownOrder": 0,
          "keyValue": "vm-21579",
          "numberOfVirtualDisks": 1,
          "hddInGbs": 40,
          "hostName": "winvmm",
          "reasonForInvalidVM": "",
          "cupResourceAllocation": {
            "reservation": 0,
            "limit": -1,
            "shares": 4000,
            "sharesLevel": 1,
            "numberOfCpus": 0,

```

```

        "corePerSocket": 0
    },
    "memoryResourceAllocation": {
        "reservation": 0,
        "limit": -1,
        "shares": 40960,
        "sharesLevel": 1,
        "numberOfCpus": 0,
        "corePerSocket": 0
    },
    "testCPUALlocation": 0,
    "actualCPUALlocation": 0,
    "prescript": "",
    "postscript": "",
    "username": "",
    "passwd": "",
    "nicNames": [
        {
            "deviceConfigId": 4000,
            "macAddress": "00:50:56:aa:e1:80",
            "nicType": "VirtualVmxnet3",
            "interfaceLabel": "Network adapter 1",
            "pciSlotNumber": 192
        }
    ],
    "vmAllNicDataList": [
        {
            "deviceConfigId": 4000,
            "macAddress": "00:50:56:aa:e1:80",
            "nicType": "VirtualVmxnet3",
            "interfaceLabel": "Network adapter 1",
            "pciSlotNumber": 192
        }
    ],
    "sourceIp": "172.29.10.145",
    "sourceSubnet": "",
    "sourceGateway": "172.29.10.100",
    "osName": "windows",
    "coresPerSocket": 1,
    "testCoresPerSocket": "1",
    "virtualSocket": 4,
    "testVirtualSocket": "4",
    "sequenceNumber": 0,
    "critical": false,
    "vappStartupOrder": 0,
    "vmAlreadyPresentInRecPlan": false,
    "sequence": "1",
    "originalDisplayAppName": "Win2012r2-2",
    "ip": ""
    }
],
    "appMembers": []
}
],
    "organizationToList": [],
    "id": ""
}

```

Body Param (For File system)

```
{
  "cdsId": 5125,
  "description": "",
  "serverId": "",
  "name": "AGFS",
  "sequenceOrder": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "vmMembers": [],
      "appMembers": [
        {
          "memberUuId": "C:\\",
          "memberName": "C:\\",
          "vm": false,
          "executionStatus": "",
          "cdsAppId": "6328",
          "cdsClusterId": "1415039054",
          "targetClusterId": "0",
          "validForAppGroup": false,
          "displayAppName": "C:\\ (win2012_sql_new) ",
          "displayAppType": "File System",
          "apptype": "FileSystem",
          "sourceHostName": "win2012_sql_new",
          "recoveryType": {
            "recoveryTypeId": 5,
            "name": "Recover Filesystem",
            "recoveryTypeOrder": "2"
          },
          "infoRequired": false,
          "id": 0,
          "appName": "C:\\",
          "originalAppId": 0,
          "sourceHostId": 6205,
          "appClass": "",
          "friendlyType": "FileSystem",
          "networkName": "",
          "targetHostId": 0,
          "depth": 0,
          "isGroup": false,
          "isClustered": false,
          "fromMount": false,
          "failoverState": 0,
          "isTbd": true,
          "sequenceNumber": 0,
          "critical": false,
          "vmalreadyPresentInRecPlan": false,
          "ip": "",
          "targetHostName": "TargetHost",
          "volumesToList": [
            {
              "sourceVolume": "C:\\",
              "targetVolume": "N:\\\"
            }
          ],
          "scriptParameterTOList": [
            {
              "scriptName": "",
              "phase": "PRE",
```



```

        "timeout": "",
        "args": "arg1,arg2"
    },
    {
        "scriptName": "",
        "phase": "POST",
        "timeout": "",
        "args": "arg3,arg4"
    }
]
}
]
}
],
"organizationToList": [],
"id": ""
}

```

Body Param (For File system, Database, and VM)

```

{
    "cdsId": 5125,
    "description": "",
    "serverId": 5127,
    "name": "AGAll",
    "sequenceOrder": [
        {
            "memberOrderIndex": 1,
            "delay": 0,
            "vmMembers": [
                {
                    "memberUuId": "502a2cbc-245f-078b-81cc-592eaf59f175",
                    "memberName": "Win2012r2-3_1549273300949",
                    "vm": true,
                    "executionStatus": "",
                    "cdsAppId": "233265",
                    "cdsClusterId": "1415039054",
                    "targetClusterId": "0",
                    "validForAppGroup": true,
                    "displayAppName": "Win2012r2-3_1549273300949",
                    "displayAppType": "VM",
                    "apptype": "VM",
                    "recoveryType": {
                        "recoveryTypeId": 2,
                        "name": "Create new VM from VMware VM",
                        "recoveryTypeOrder": "1"
                    },
                    "infoRequired": false,
                    "id": 0,
                    "vappName": "",
                    "numberOfCpus": 4,
                    "actualNumberOfNics": 1,
                    "testNumberOfNics": 1,
                    "testMemoryInMbs": "4096",
                    "actualMemoryInMbs": 4096,
                    "storageLocation": "[datastore1tb] Win2012r2-3/Win2012r2-3.vmx",
                    "shutdownOrder": 0,
                    "keyValue": "vm-21580",
                    "numberOfVirtualDisks": 1,
                    "hddInGbs": 40,
                    "hostName": "winvwm",

```

```

"reasonForInvalidVM": "",
"cupResourceAllocation": {
  "reservation": 0,
  "limit": -1,
  "shares": 4000,
  "sharesLevel": 1,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"memoryResourceAllocation": {
  "reservation": 0,
  "limit": -1,
  "shares": 40960,
  "sharesLevel": 1,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"testCPUALlocation": 0,
"actualCPUALlocation": 0,
"prescript": "",
"postscript": "",
"username": "",
"passwd": "",
"nicNames": [
  {
    "deviceConfigId": 4000,
    "macAddress": "00:50:56:aa:f5:e3",
    "nicType": "VirtualVmxnet3",
    "interfaceLabel": "Network adapter 1",
    "pciSlotNumber": 192
  }
],
"vmAllNicDataList": [
  {
    "deviceConfigId": 4000,
    "macAddress": "00:50:56:aa:f5:e3",
    "nicType": "VirtualVmxnet3",
    "interfaceLabel": "Network adapter 1",
    "pciSlotNumber": 192
  }
],
"sourceIp": "169.254.147.30",
"sourceSubnet": "",
"sourceGateway": "172.29.10.100",
"osName": "windows",
"coresPerSocket": 1,
"testCoresPerSocket": "1",
"virtualSocket": 4,
"testVirtualSocket": "4",
"sequenceNumber": 0,
"critical": false,
"vappStartupOrder": 0,
"vmAlreadyPresentInRecPlan": false,
"sequence": "1",
"originalDisplayAppName": "Win2012r2-3_1549273300949",
"isCreateNewVmScenario": true,
"ip": "",
"preScriptType": "",
"postScriptType": "",
"vmUserCredentials": {

```

```

        "id": 7172,
        "name": "Win",
        "username": "Administrator",
        "password": "",
        "systemid": "8a10f297-a384-4a73-85fc-c257f6ea94ed",
        "peercredentialsid": 7173
    },
    "defaultDnsIpToconfigure": [],
    "domainname": "",
    "vmOsHostname": "Host1",
    "domainCredentials": null
}
],
"appMembers": []
},
{
    "memberOrderIndex": 2,
    "delay": 0,
    "vmMembers": [],
    "appMembers": [
        {
            "memberUuId": "C:\\",
            "memberName": "C:\\",
            "vm": false,
            "executionStatus": "",
            "cdsAppId": "6328",
            "cdsClusterId": "1415039054",
            "targetClusterId": "0",
            "validForAppGroup": false,
            "displayAppName": "C:\\ (win2012_sql_new) ",
            "displayAppType": "File System",
            "apptype": "FileSystem",
            "sourceHostName": "win2012_sql_new",
            "recoveryType": {
                "recoveryTypeId": 5,
                "name": "Recover Filesystem",
                "recoveryTypeOrder": "2"
            },
            "infoRequired": false,
            "id": 0,
            "appName": "C:\\",
            "originalAppId": 0,
            "sourceHostId": 6205,
            "appClass": "",
            "friendlyType": "FileSystem",
            "networkName": "",
            "targetHostId": 0,
            "depth": 0,
            "isGroup": false,
            "isClustered": false,
            "fromMount": false,
            "failoverState": 0,
            "isTbd": false,
            "sequenceNumber": 0,
            "critical": false,
            "vmAlreadyPresentInRecPlan": false,
            "ip": "",
            "targetHostName": "Win2012r2-3_1549273300949",
            "volumesToList": [
                {

```

```

        "sourceVolume": "C:\\",
        "targetVolume": "N:\\",
    }
],
"scriptParameterTOList": [
    {
        "scriptName": "",
        "phase": "PRE",
        "timeout": "",
        "args": "arg1,arg2"
    },
    {
        "scriptName": "",
        "phase": "POST",
        "timeout": "",
        "args": "arg3,arg4"
    }
]
}
]
},
{
    "memberOrderIndex": 3,
    "delay": 0,
    "vmMembers": [],
    "appMembers": [
        {
            "memberUuId": "WINVVM",
            "memberName": "WINVVM",
            "vm": false,
            "executionStatus": "",
            "cdsAppId": "231634",
            "cdsClusterId": "1415039054",
            "targetClusterId": "0",
            "validForAppGroup": false,
            "displayAppName": "WINVVM (win2012r2-1) ",
            "displayAppType": "SQL Instance",
            "apptype": "SqlInstance",
            "sourceHostName": "win2012r2-1",
            "recoveryType": {
                "recoveryTypeId": 4,
                "name": "Recover SQL Server Database",
                "recoveryTypeOrder": "2"
            },
        },
        "infoRequired": false,
        "id": 0,
        "appName": "WINVVM",
        "originalAppId": 0,
        "sourceHostId": 231508,
        "appClass": "SQLServerGroup",
        "friendlyType": "SqlInstance",
        "networkName": "",
        "targetHostId": 0,
        "depth": 0,
        "isGroup": false,
        "isClustered": false,
        "fromMount": false,
        "failoverState": 0,
        "isTbd": false,
        "sequenceNumber": 0,
    ]
}

```

```

    "critical": false,
    "vmAlreadyPresentInRecPlan": false,
    "ip": "",
    "targetHostName": "Win2012r2-3_1549273300949",
    "volumesToList": [],
    "scriptParameterTOList": [],
    "appClassMetaDataTOList": [
      {
        "appid": 0,
        "parameterId": 7186,
        "value": "ConGrp1"
      },
      {
        "appid": 0,
        "parameterId": 7187,
        "value": "WINVVM"
      },
      {
        "appid": 0,
        "parameterId": 7188,
        "value": "pre"
      },
      {
        "appid": 0,
        "parameterId": 7189,
        "value": "post"
      },
      {
        "appid": 0,
        "parameterId": 7190,
        "value": true
      },
      {
        "appid": 0,
        "parameterId": 7191,
        "value": false
      },
      {
        "appid": 0,
        "parameterId": 7192
      },
      {
        "appid": 0,
        "parameterId": 7193
      }
    ],
    "inclusionList": "rdtest2,rdtest3",
    "exclusionList": ""
  }
]
}
],
"organizationToList": [],
"id": ""
}

```

Body Param (For VM in sequence 1 and CAF database – MySQL instance in sequence 2)

```

{
  "cdsId": 6251,
  "description": "",

```

```

    "id":23771,
    "name":"92258_AG1",
    "organizationToList":[

],
    "postScript":null,
    "postScriptType":null,
    "postScriptVMName":null,
    "postScriptVMUUID":null,
    "preScript":null,
    "preScriptType":null,
    "preScriptVMName":null,
    "preScriptVMUUID":null,
    "sequenceOrder":[
        {
            "appMembers":[

],
            "delay":0,
            "memberOrderIndex":1,
            "vmMembers":[
                {
                    "actualCPUAllocation":0,
                    "actualMemoryInMbs":6144,
                    "actualNumberOfNics":1,
                    "appliancegroupid":0,
                    "apptype":"VM",
                    "bootdisksize":0,
                    "cdsAppId":"1147444",
                    "cdsClusterId":"145649790597",
                    "coresPerSocket":1,
                    "critical":false,
                    "cupResourceAllocation":{
                        "corePerSocket":0,
                        "limit":-1,
                        "numberOfCpus":0,
                        "reservation":0,
                        "shares":4000,
                        "sharesLevel":1
                    },
                    "defaultDnsIpToconfigure":[

],
                    "displayAppName":"GTC_Mysql_Host",
                    "displayAppType":"VM",
                    "domainname":"",
                    "executionStatus":"",
                    "hddInGbs":312,
                    "hostName":"GTC-Mysql",
                    "id":0,
                    "infoRequired":false,
                    "ip":"",
                    "isAsm":false,
                    "keyValue":"vm-120419",
                    "memberName":"GTC_Mysql_Host",
                    "memberUuid":"5004e9ef-2f08-87da-d215-962ea995fe93",
                    "memoryResourceAllocation":{
                        "corePerSocket":0,
                        "limit":-1,
                        "numberOfCpus":0,

```

```

        "reservation":0,
        "shares":61440,
        "sharesLevel":1
    },
    "nicForDns":"","
    "nicForStaticIp":"","
    "nicNames":[
        {
            "deviceConfigId":4000,
            "interfaceLabel":"Network adapter 1",
            "macAddress":"00:50:56:84:7e:f0",
            "nicType":"VirtualVmxnet3",
            "pciSlotNumber":0
        }
    ],
    "numberOfCpus":4,
    "numberOfVirtualDisks":5,
    "originalBootdiskSize":0,
    "originalDisplayAppName":"GTC_Mysql_Host",
    "osName":"linux",
    "passwd":"","
    "policyOpCode":0,
    "postscript":"vm_reset.sh",
    "postScriptType":"shell",
    "prescript":"vm_failover.sh",
    "preScriptType":"shell",
    "reasonForInvalidVM":"","
    "recoveryType":{
        "description":"Add VMs you wish to recover at the DR site. Each VM may be
recovered only once and will retain its original host name.",
        "name":"Recover VM From VMWare VM Backup",
        "recoveryTypeId":1,
        "recoveryTypeOrder":"1"
    },
    "sequence":1,
    "sequenceNumber":1,
    "shutdownOrder":0,
    "sourceGateway":"172.27.44.1",
    "sourceIp":"172.27.44.123",
    "sourceSubnet":"","
    "storageLocation":"[datastore22] GTC_Mysql_Host/GTC_Mysql_Host.vmx",
    "targetClusterId":"142452487387",
    "testCoresPerSocket":"1",
    "testCPUAllocation":0,
    "testMemoryInMbs":6144,
    "testNumberOfNics":1,
    "testVirtualSocket":"4",
    "username":"","
    "validForAppGroup":true,
    "vappName":"","
    "vappStartupOrder":0,
    "virtualSocket":4,
    "vm":true,
    "vmAllNicDataList":[
        {
            "deviceConfigId":"4000",
            "enforcedStaticDnsIp":false,
            "enforcedStaticIp":false,
            "gateway":"","
            "interfaceLabel":"Network adapter 1",

```

```

        "macAddress": "00:50:56:84:7e:f0",
        "nicType": "VirtualVmxnet3",
        "pciSlotNumber": 192
    }
},
"vmAlreadyPresentInRecPlan": false,
"vmOsHostname": "",
"vmUserCredentials": {
    "id": 15534,
    "name": "mysqlsandy",
    "password": "12!pass345",
    "peerCredentialsId": 24986,
    "systemId": "0a0c2102-9fa9-442e-ae53-84ef9476bf4c",
    "username": "root"
}
}
]
},
{
    "appMembers": [
        {
            "appClass": "MYSQLGroup",
            "applicationChildMembers": [
                ],
            "appName": "mysqld_3066",
            "apptype": "MYSQLInstance",
            "cdsAppId": "1147379",
            "cdsClusterId": "145649790597",
            "clustername": "sky104source",
            "critical": false,
            "depth": 0,
            "displayAppName": "mysqld_3066(GTC-Mysql-RootDB-Host)",
            "displayAppType": "MYSQLInstance",
            "executionStatus": "",
            "failoverState": 0,
            "friendlyType": "MYSQLInstance",
            "fromMount": false,
            "infoRequired": false,
            "ip": "",
            "isAsm": false,
            "isClustered": false,
            "iscustomapp": true,
            "isGroup": false,
            "isTbd": false,
            "memberName": "mysqld_3066",
            "memberUuId": "mysqld_3066",
            "networkName": "",
            "originalAppId": 0,
            "policyOpCode": 0,
            "provisioningParameters": "<provisioningoptions><PORT>3066</PORT><OSUSER>mysql</OSUSER><BASEDIR>/usr/local/mysql-8.0.22-el7-x86_64</BASEDIR></provisioningoptions>",
            "recoveryType": {
                "description": "Select MySQL applications to recover at the DR site",
                "name": "Recover MySQL Instance",
                "recoveryTypeId": 6263,
                "recoveryTypeOrder": "1"
            },
            "scriptParameterTOList": [
                {

```



```

        "args": "",
        "phase": "PRE",
        "scriptName": "/act/scripts/pre.sh",
        "timeout": "0"
    },
    {
        "args": "",
        "phase": "POST",
        "scriptName": "/act/scripts/post.sh",
        "timeout": "0"
    }
],
"sequenceNumber": 2,
"sourceHostId": 456582,
"sourceHostName": "GTC-Mysql-RootDB-Host",
"targetClusterId": "142452487387",
"targetHostId": 0,
"targetHostName": "GTC_Mysql_Host",
"validForAppGroup": false,
"vm": false,
"vmAlreadyPresentInRecPlan": false
}
],
"delay": 0,
"memberOrderIndex": 2,
"vmMembers": [
]
}
],
"serverId": 9221
}

```

Response

```

{
  "status": "success",
  "statusCode": "1009",
  "statusMessage": "Application group added successfully with name: ",
  "responseData": ""
}

```

Parameters	Description
appGroupTOAdd	Application group details for creating application object in Resiliency Director database.
uniqueKey	Value which is passed to fetch new member request.

getAppGroups

Use this API to fetch Application Group(s) managed by the Resiliency Director from application database. If the method parameter is null, the method returns all the application groups from the database. If the method parameter is not null, the method returns only the application group(s) whose id(s) have been provided as input.

Method: GET

Request

https://{HOST}/armnew/api/applicationgroup/fetch

https://{HOST}/armnew/api/applicationgroup/fetch?ids=1,2,3

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 7243,
      "name": "AGAll",
      "description": "",
      "cdsId": 0,
      "cdsHostName": "RD_HYD_SRC",
      "serverId": 5127,
      "serverHostName": "172.29.11.185",
      "createdOn": "02/04/2019 09:44:21",
      "editedOn": "02/04/2019 09:44:21",
      "noOfVms": 1,
      "noOfNonVms": 2,
      "recoveryPlan": "",
      "noOfVApps": 0,
      "sequenceOrder": [
        {
          "memberOrderIndex": 1,
          "delay": 0,
          "vmMembers": [
            {
              "memberUuId": "502a2cbc-245f-078b-81cc-592eaf59f175",
              "memberName": "Win2012r2-3_1549273300949",
              "uniqueGridIndex": "1549273691295",
              "vm": true,
              "executionStatus": "",
              "cdsAppId": "233265",
              "cdsClusterId": "1415039054",
              "targetClusterId": "1415040293",
              "clustername": "RD_HYD_SRC",
              "validForAppGroup": false,
              "displayAppName": "Win2012r2-3_1549273300949",
              "displayAppType": "VM",
              "apptype": "VM",
              "recoveryType": {
                "recoveryTypeId": 2,
                "name": "Create new VM from VMware VM",
                "recoveryTypeOrder": "1"
              },
              "infoRequired": false,
              "id": 7245,
              "vappName": "",
              "numberOfCpus": 0,
              "actualNumberOfNics": 1,
              "testNumberOfNics": 1,
              "testMemoryInMbs": 4096,
              "actualMemoryInMbs": 4096,
              "shutdownOrder": 0,
              "numberOfVirtualDisks": 0,
              "hddInGbs": 0,
              "hostName": "winvwm",
              "ip": "",

```

```

"cupResourceAllocation": {
  "reservation": 0,
  "limit": -1,
  "shares": 4000,
  "sharesLevel": 1,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"memoryResourceAllocation": {
  "reservation": 0,
  "limit": -1,
  "shares": 40960,
  "sharesLevel": 1,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"testCPUALlocation": 0,
"actualCPUALlocation": 0,
"prescript": "",
"postscript": "",
"username": "",
"passwd": "",
"nicNames": [
  {
    "deviceConfigId": 4000,
    "macAddress": "00:50:56:aa:f5:e3",
    "nicType": "VirtualVmxnet3",
    "interfaceLabel": "Network adapter 1",
    "pciSlotNumber": 0
  }
],
"vmAllNicDataList": [
  {
    "deviceConfigId": "4000",
    "interfaceLabel": "Network adapter 1",
    "enforcedStaticIp": false,
    "enforcedStaticDnsIp": false,
    "gateway": "",
    "macAddress": "00:50:56:aa:f5:e3",
    "nicType": "VirtualVmxnet3",
    "pciSlotNumber": 192
  }
],
"nicForStaticIp": "",
"nicForDns": "",
"defaultDnsIpToconfigure": [],
"sourceIp": "",
"sourceSubnet": "",
"sourceGateway": "",
"portGroup": [],
"osName": "windows",
"preScriptType": "",
"postScriptType": "",
"preExternalScript": [],
"postExternalScript": [],
"coresPerSocket": 1,
"testCoresPerSocket": 1,
"virtualSocket": 4,
"testVirtualSocket": 4,
"domainname": "",

```

```

    "vmOsHostname": "Host1",
    "vmUserCredentials": {
      "id": 7172,
      "name": "Win",
      "username": "Administrator",
      "password": "maheshwar",
      "systemid": "8a10f297-a384-4a73-85fc-c257feea94ed",
      "peercredentialsid": 7173
    },
    "sequenceNumber": 1,
    "critical": false,
    "vappStartupOrder": 0,
    "vmalreadyPresentInRecPlan": false
  }
],
"appMembers": [],
"preExternalScript": [],
"postExternalScript": []
},
{
  "memberOrderIndex": 2,
  "delay": 0,
  "vmMembers": [],
  "appMembers": [
    {
      "memberName": "C:\\",
      "vm": false,
      "executionStatus": "",
      "cdsAppId": "6328",
      "cdsClusterId": "1415039054",
      "targetClusterId": "1415040293",
      "clustername": "RD_HYD_SRC",
      "validForAppGroup": false,
      "displayAppName": "C:\\ (win2012_sql_new) ",
      "displayAppType": "File System",
      "applicationChildMembers": [],
      "apptype": "FileSystem",
      "sourceHostName": "win2012_sql_new",
      "recoveryType": {
        "recoveryTypeId": 5,
        "name": "Recover Filesystem",
        "recoveryTypeOrder": "2"
      },
      "infoRequired": false,
      "id": 7249,
      "appName": "C:\\",
      "originalAppId": 0,
      "sourceHostId": 0,
      "appClass": "",
      "friendlyType": "FileSystem",
      "networkName": "",
      "targetHostName": "Win2012r2-3_1549273300949",
      "targetHostId": 0,
      "depth": 0,
      "isGroup": false,
      "isClustered": false,
      "fromMount": false,
      "failoverState": 0,
      "isTbd": false,
      "restoreOptions": "volume#C:\\,targetvolume#N:\\",

```

```

        "scriptParameters":
"name=:phase=PRE:timeout=:args=;name=:phase=POST:timeout=:args=",
        "volumesToList": [
            {
                "sourceVolume": "C:\\",
                "targetVolume": "N:\\",
            }
        ],
        "scriptParameterTOList": [
            {
                "scriptName": "",
                "phase": "PRE",
                "timeout": "",
                "args": ""
            },
            {
                "scriptName": "",
                "phase": "POST",
                "timeout": "",
                "args": ""
            }
        ],
        "appClassMetaDataTOList": [],
        "sequenceNumber": 2,
        "critical": false,
        "vmAlreadyPresentInRecPlan": false
    }
],
"preExternalScript": [],
"postExternalScript": []
},
{
    "memberOrderIndex": 3,
    "delay": 0,
    "vmMembers": [],
    "appMembers": [
        {
            "memberName": "WINVVM",
            "vm": false,
            "executionStatus": "",
            "cdsAppId": "231634",
            "cdsClusterId": "1415039054",
            "targetClusterId": "1415040293",
            "clustername": "RD_HYD_SRC",
            "validForAppGroup": false,
            "displayAppName": "WINVVM (win2012r2-1) ",
            "displayAppType": "SQL Instance",
            "applicationChildMembers": [],
            "apptype": "SqlInstance",
            "sourceHostName": "win2012r2-1",
            "recoveryType": {
                "recoveryTypeId": 4,
                "name": "Recover SQL Server Database",
                "recoveryTypeOrder": "2"
            },
            "infoRequired": false,
            "id": 7251,
            "appName": "WINVVM",
            "originalAppId": 0,
            "sourceHostId": 0,

```

```

    "appClass": "SQLServerGroup",
    "friendlyType": "SqlInstance",
    "networkName": "",
    "targetHostName": "Win2012r2-3_1549273300949",
    "targetHostId": 0,
    "depth": 0,
    "isGroup": false,
    "isClustered": false,
    "fromMount": false,
    "failoverState": 0,
    "provisioningParameters": "<provisioningoptions><ConsistencyGroupName>ConGrp1</ConsistencyGroupName><sqlinstance>WINVVM</sqlinstance><dbnameprefix>pre</dbnameprefix><dbnamesuffix>post</dbnamesuffix><recover>true</recover><userlogins>>false</userlogins></provisioningoptions>",
    "isTbd": false,
    "restoreOptions": "",
    "scriptParameters": "",
    "appClassMetaDataTOList": [
        {
            "appClassPropertiesT0": {
                "propertyId": 7186,
                "name": "SQLServerGroup",
                "propertyName": "ConsistencyGroupName",
                "propertyType": "STRING",
                "propertyLabel": "Name of Consistency Group",
                "isRequired": true,
                "description": "Specify name for the new Consistency Group that will be created as a result of app aware mount. The name must be unique across the appliance."
            },
            "appid": 7251,
            "parameterId": 7186,
            "value": "ConGrp1"
        },
        {
            "appClassPropertiesT0": {
                "propertyId": 7187,
                "name": "SQLServerGroup",
                "propertyName": "sqlinstance",
                "propertyType": "EDITABLE_SELECT",
                "propertyLabel": "SQL Server Instance Name",
                "isRequired": true,
                "description": "Select the target SQL Server Instance. The new databases will be managed by the selected SQL Server instance."
            },
            "appid": 7251,
            "parameterId": 7187,
            "value": "WINVVM"
        },
        {
            "appClassPropertiesT0": {
                "propertyId": 7188,
                "name": "SQLServerGroup",
                "propertyName": "dbnameprefix",
                "propertyType": "STRING",
                "propertyLabel": "Prefix for SQL Server Database Name",
                "isRequired": false,
                "description": "Specify prefix to be used for the newly provisioned database names. This string will be prefixed to the original database name to create new database name. Valid characters include letters, numbers, @, #, -, , and a trailing space character."
            }
        }
    ]

```

```

    "appid": 7251,
    "parameterId": 7188,
    "value": "pre"
  },
  {
    "appClassPropertiesT0": {
      "propertyId": 7189,
      "name": "SQLServerGroup",
      "propertyName": "dbnamesuffix",
      "propertyType": "STRING",
      "propertyLabel": "Suffix for SQL Server Database Name",
      "isRequired": false,
      "description": "Specify suffix to be used for the newly provisioned database
names. This string will be appended to the original database name to create new database name.
Valid characters include letters, numbers, @, #, -, , and a leading space character."
    },
    "appid": 7251,
    "parameterId": 7189,
    "value": "post"
  },
  {
    "appClassPropertiesT0": {
      "propertyId": 7190,
      "name": "SQLServerGroup",
      "propertyName": "recover",
      "propertyType": "BOOLEAN",
      "propertyLabel": "Recover Database After Restore",
      "isRequired": false,
      "description": "If selected, brings the newly created database online."
    },
    "appid": 7251,
    "parameterId": 7190,
    "value": "true"
  },
  {
    "appClassPropertiesT0": {
      "propertyId": 7191,
      "name": "SQLServerGroup",
      "propertyName": "userlogins",
      "propertyType": "BOOLEAN",
      "propertyLabel": "Recover User Logins",
      "isRequired": false,
      "description": "If selected, recover the user logins of the database. This
may require user credentials to be entered."
    },
    "appid": 7251,
    "parameterId": 7191,
    "value": "false"
  },
  {
    "appClassPropertiesT0": {
      "propertyId": 7192,
      "name": "SQLServerGroup",
      "propertyName": "username",
      "propertyType": "STRING",
      "propertyLabel": "User Name",
      "isRequired": false,
      "description": "Specify user name for database provisioning. Only required
when the account running the connector (typically 'Local System') does not have privileges to
apply transaction logs, or to detach a database (required during subsequent unmount)."
    }
  }
}

```

```

    },
    "appid": 7251,
    "parameterId": 7192
  },
  {
    "appClassPropertiesT0": {
      "propertyId": 7193,
      "name": "SQLServerGroup",
      "propertyName": "password",
      "propertyType": "ENCRYPT",
      "propertyLabel": "Password",
      "isRequired": false,
      "description": "Password for the specified user."
    },
    "appid": 7251,
    "parameterId": 7193
  }
],
"inclusionList": "rdtest2,rdtest3",
"sequenceNumber": 3,
"critical": false,
"vmAlreadyPresentInRecPlan": false
}
],
"preExternalScript": [],
"postExternalScript": []
}
],
"armServerId": 0,
"recoveryPlanId": 0,
"executionStatus": "",
"peerRecoveryPlanId": 0,
"cpuRequired": 0,
"memoryRequired": 0,
"customerName": "",
"peerAppGrpId": 0,
"preExternalScript": [],
"postExternalScript": [],
"organizationToList": [],
"startupSequence": 0,
"critical": false
},
{
  "id": 7261,
  "name": "AGVM",
  "description": "",
  "cdsId": 0,
  "cdsHostName": "RD_HYD_SRC",
  "serverId": 5127,
  "serverHostName": "172.29.11.185",
  "createdOn": "02/04/2019 09:47:57",
  "editedOn": "02/04/2019 09:47:57",
  "noOfVms": 1,
  "noOfNonVms": 0,
  "recoveryPlan": "",
  "noOfVApps": 0,
  "sequenceOrder": [
    {
      "memberOrderIndex": 1,
      "delay": 0,

```



```

"vmMembers": [
{
  "memberUuId": "502ab92d-7e57-8f43-891c-07a07d85e429",
  "memberName": "Rhel1",
  "uniqueGridIndex": "1549273712407",
  "vm": true,
  "executionStatus": "",
  "cdsAppId": "106809",
  "cdsClusterId": "1415039054",
  "targetClusterId": "1415040293",
  "clustername": "RD_HYD_SRC",
  "validForAppGroup": false,
  "displayAppName": "Rhel1",
  "displayAppType": "VM",
  "apptype": "VM",
  "recoveryType": {
    "recoveryTypeId": 1,
    "name": "Recover VM From VMWare VM Backup",
    "recoveryTypeOrder": "1"
  },
  "infoRequired": false,
  "id": 7263,
  "vappName": "",
  "numberOfCpus": 0,
  "actualNumberOfNics": 1,
  "testNumberOfNics": 1,
  "testMemoryInMbs": 2048,
  "actualMemoryInMbs": 2048,
  "shutdownOrder": 0,
  "numberOfVirtualDisks": 0,
  "hddInGbs": 0,
  "hostName": "localhost.localdomain",
  "ip": "",
  "cupResourceAllocation": {
    "reservation": 0,
    "limit": -1,
    "shares": 1000,
    "sharesLevel": 1,
    "numberOfCpus": 0,
    "corePerSocket": 0
  },
  "memoryResourceAllocation": {
    "reservation": 0,
    "limit": -1,
    "shares": 20480,
    "sharesLevel": 1,
    "numberOfCpus": 0,
    "corePerSocket": 0
  },
  "testCPUALlocation": 0,
  "actualCPUALlocation": 0,
  "prescript": "",
  "postscript": "",
  "username": "",
  "passwd": "",
  "nicNames": [
    {
      "deviceConfigId": 4000,
      "macAddress": "00:50:56:aa:b4:2c",
      "nicType": "VirtualVmxnet3",

```

```

        "interfaceLabel": "Network adapter 1",
        "pciSlotNumber": 0
    }
],
"vmAllNicDataList": [
    {
        "deviceConfigId": "4000",
        "interfaceLabel": "Network adapter 1",
        "enforcedStaticIp": false,
        "enforcedStaticDnsIp": false,
        "gateway": "",
        "macAddress": "00:50:56:aa:b4:2c",
        "nicType": "VirtualVmxnet3",
        "pciSlotNumber": 192
    }
],
"nicForStaticIp": "",
"nicForDns": "",
"defaultDnsIpToconfigure": [],
"sourceIp": "",
"sourceSubnet": "",
"sourceGateway": "",
"portGroup": [],
"osName": "linux",
"preExternalScript": [],
"postExternalScript": [],
"coresPerSocket": 1,
"testCoresPerSocket": 1,
"virtualSocket": 1,
"testVirtualSocket": 1,
"sequenceNumber": 1,
"critical": false,
"vappStartupOrder": 0,
"vmalreadyPresentInRecPlan": false
}
],
"appMembers": [],
"preExternalScript": [],
"postExternalScript": []
}
],
"armServerId": 0,
"recoveryPlanId": 0,
"executionStatus": "",
"peerRecoveryPlanId": 0,
"cpuRequired": 0,
"memoryRequired": 0,
"customerName": "",
"peerAppGrpId": 0,
"preExternalScript": [],
"postExternalScript": [],
"organizationToList": [],
"startupSequence": 0,
"critical": false
}
]
}

```

Parameters	Description
sessionid	Session id as a header parameter.
ids	Comma separated application group id(s) to be fetched.

removeAppGroup

Use this API to delete application Group(s) managed by Resiliency Director with the parameters.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/applicationgroup/remove?ids=1,2,3 (HTTPS)

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}
```

Parameters	Description
appGroupIdsToDelete	Comma separated id(s) of the application group(s) to be deleted.

getVirtualMachines

Use this API to fetch virtual machines for the given application group id managed by the Resiliency Director from application database.

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/
fetchvmdetails?serverid=5207&uniqueKey=1548932689394&appgroupname=AG

Body Param

```
{
  "memberOrderIndex": 1,
  "delay": 0,
  "vmMembers": [
    {
      "memberUuId": "502ab92d-7e57-8f43-891c-07a07d85e429",
      "memberName": "Rhel1_1549271228018",
      "uniqueGridIndex": "1549270918148_1549271228018",
      "vm": true,
      "executionStatus": "",
      "cdsAppId": "106809",
      "cdsClusterId": "1415039054",
      "targetClusterId": "0",
      "clustername": "RD_HYD_SRC",
      "validForAppGroup": false,
    }
  ]
}
```

```

        "displayAppName": "Rhel1_1549271228018",
        "displayAppType": "VM",
        "apptype": "VM",
        "sourceHostName": "rhel1",
        "recoveryType": {
            "recoveryTypeId": 2,
            "name": "Create new VM from VMware VM",
            "recoveryTypeOrder": "1"
        },
        "infoRequired": false,
        "vmalreadyPresentInRecPlan": false,
        "sequence": "1"
    }
],
"appMembers": []
}

```

Response

```

{
    "status": "success",
    "statusCode": "1000",
    "statusMessage": "Operation performed successfully.",
    "responseData": [
        {
            "memberOrderIndex": 1,
            "delay": 0,
            "vmMembers": [
                {
                    "memberUuId": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",
                    "memberName": "Win2012r2-2_1548932687404",
                    "vm": true,
                    "executionStatus": "",
                    "cdsAppId": "231511",
                    "cdsClusterId": "1415039054",
                    "targetClusterId": "0",
                    "validForAppGroup": true,
                    "displayAppName": "Win2012r2-2_1548932687404",
                    "displayAppType": "VM",
                    "apptype": "VM",
                    "recoveryType": {
                        "recoveryTypeId": 2,
                        "name": "Create new VM from VMware VM",
                        "recoveryTypeOrder": "1"
                    },
                    "infoRequired": false,
                    "id": 0,
                    "vappName": "",
                    "numberOfCpus": 4,
                    "actualNumberOfNics": 1,
                    "testNumberOfNics": 1,
                    "testMemoryInMbs": 4096,
                    "actualMemoryInMbs": 4096,
                    "storageLocation": "[datastore1tb] Win2012r2-2/Win2012r2-2.vmx",
                    "shutdownOrder": 0,
                    "keyValue": "vm-21579",
                    "numberOfVirtualDisks": 1,
                    "hddInGbs": 40,
                    "hostName": "winvwm",
                    "reasonForInvalidVM": "",
                    "cupResourceAllocation": {

```

```

        "reservation": 0,
        "limit": -1,
        "shares": 4000,
        "sharesLevel": 1,
        "numberOfCpus": 0,
        "corePerSocket": 0
    },
    "memoryResourceAllocation": {
        "reservation": 0,
        "limit": -1,
        "shares": 40960,
        "sharesLevel": 1,
        "numberOfCpus": 0,
        "corePerSocket": 0
    },
    "testCPUAllocation": 0,
    "actualCPUAllocation": 0,
    "prescript": "",
    "postscript": "",
    "username": "",
    "passwd": "",
    "nicNames": [
        {
            "deviceConfigId": 4000,
            "macAddress": "00:50:56:aa:e1:80",
            "nicType": "VirtualVmxnet3",
            "interfaceLabel": "Network adapter 1",
            "pciSlotNumber": 192
        }
    ],
    "vmAllNicDataList": [],
    "sourceIp": "172.29.10.145",
    "sourceSubnet": "",
    "sourceGateway": "172.29.10.100",
    "osName": "windows",
    "coresPerSocket": 1,
    "testCoresPerSocket": 1,
    "virtualSocket": 4,
    "testVirtualSocket": 4,
    "sequenceNumber": 0,
    "critical": false,
    "vappStartupOrder": 0,
    "vmaIreadyPresentInRecPlan": false
    }
    ],
    "appMembers": []
    }
    ]
}

```

Parameters	Description
applicationGroupIid	An application group id for which you want to fetch the virtual machines.

fetchVMAndVappsForAppGroup

Use this API to fetch virtual machines, UUID, vApps, and recovery order, for the given application group id managed by Resiliency Director from application database.

Method: GET

Request

```
https://{HOST}/armnewv/api/applicationgroup/  
fetchvmvappdetails?applicationgroupid=8311&selectedcds=5205&uniqueKey=1548932800886&appgroupna  
me=AG&_=1548931664109
```

Response

```
{  
  "status": "success",  
  "statusCode": "1000",  
  "statusMessage": "Operation performed successfully.",  
  "responseData": [  
    {  
      "memberOrderIndex": 1,  
      "delay": 0,  
      "vmMembers": [  
        {  
          "memberUuid": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",  
          "memberName": "Win2012r2-2_1548932687404",  
          "uniqueGridIndex": "1548932802354",  
          "vm": true,  
          "executionStatus": "",  
          "cdsAppId": "231511",  
          "cdsClusterId": "1415039054",  
          "targetClusterId": "1415040293",  
          "clustername": "RD_HYD_SRC",  
          "validForAppGroup": false,  
          "displayAppName": "Win2012r2-2_1548932687404",  
          "displayAppType": "VM",  
          "apptype": "VM",  
          "recoveryType": {  
            "recoveryTypeId": 2,  
            "name": "Create new VM from VMware VM",  
            "recoveryTypeOrder": "1"  
          },  
          "infoRequired": true,  
          "id": 0,  
          "vappName": "",  
          "numberOfCpus": 0,  
          "actualNumberOfNics": 0,  
          "testNumberOfNics": 0,  
          "testMemoryInMbs": 0,  
          "actualMemoryInMbs": 0,  
          "shutdownOrder": 0,  
          "numberOfVirtualDisks": 0,  
          "hddInGbs": 0,  
          "hostName": "",  
          "ip": "",  
          "cupResourceAllocation": {  
            "reservation": 0,  
            "limit": 0,  
            "shares": 0,  
            "sharesLevel": 0,  
            "numberOfCpus": 0,  
            "corePerSocket": 0  
          }  
        }  
      ]  
    }  
  ]  
}
```

```

    },
    "memoryResourceAllocation": {
      "reservation": 0,
      "limit": 0,
      "shares": 0,
      "sharesLevel": 0,
      "numberOfCpus": 0,
      "corePerSocket": 0
    },
    "testCPUAllocation": 0,
    "actualCPUAllocation": 0,
    "prescript": "",
    "postscript": "",
    "username": "",
    "passwd": "",
    "nicNames": [],
    "nicForStaticIp": "",
    "nicForDns": "",
    "defaultDnsIpToconfigure": [],
    "sourceIp": "",
    "portGroup": [],
    "osName": "",
    "preExternalScript": [],
    "postExternalScript": [],
    "coresPerSocket": 0,
    "testCoresPerSocket": 0,
    "virtualSocket": 0,
    "testVirtualSocket": 0,
    "sequenceNumber": 0,
    "critical": false,
    "vappStartupOrder": 0,
    "vmaAlreadyPresentInRecPlan": false
  }
],
"appMembers": []
}
]
}

```

Parameters	Description
sessionid	Session id as a header parameter.
applicationGroupId	Returns ID application group of the Virtual machines and vApps data.
selectedcds	Ids of the selected CDS

updateAppGroup

Use this API to update an application group with VM details into the Resiliency Director database.

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/
update?uniqueKey=1549274090163&appgroupname=AGVM&suppressWarningNicValidation=true

Body Param

```
{
  "cdsId": 5125,
  "description": "",
  "serverId": 5127,
  "name": "AGVM",
  "preScriptVMUUID": "",
  "postScriptVMUUID": "",
  "sequenceOrder": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "vmMembers": [
        {
          "memberUuid": "502ab92d-7e57-8f43-891c-07a07d85e429",
          "memberName": "Rhel1",
          "vm": true,
          "executionStatus": "",
          "cdsAppId": "106809",
          "cdsClusterId": "1415039054",
          "targetClusterId": "1415040293",
          "validForAppGroup": true,
          "displayAppName": "Rhel1",
          "displayAppType": "VM",
          "apptype": "VM",
          "recoveryType": {
            "recoveryTypeId": 1,
            "name": "Recover VM From VMWare VM Backup",
            "recoveryTypeOrder": "1"
          },
          "infoRequired": false,
          "id": 0,
          "vappName": "",
          "numberOfCpus": 1,
          "actualNumberOfNics": 1,
          "testNumberOfNics": 1,
          "testMemoryInMbs": "2048",
          "actualMemoryInMbs": 2048,
          "storageLocation": "[datastore1tb] Rhel1/Rhel1.vmx",
          "shutdownOrder": 0,
          "keyValue": "vm-16635",
          "numberOfVirtualDisks": 2,
          "hddInGbs": 22,
          "hostName": "localhost.localdomain",
          "ip": "",
          "reasonForInvalidVM": "",
          "cupResourceAllocation": {
            "reservation": 0,
            "limit": -1,
            "shares": 1000,
            "sharesLevel": 1,

```



```

        "numberOfCpus": 0,
        "corePerSocket": 0
    },
    "memoryResourceAllocation": {
        "reservation": 0,
        "limit": -1,
        "shares": 20480,
        "sharesLevel": 1,
        "numberOfCpus": 0,
        "corePerSocket": 0
    },
    "testCPUALlocation": 0,
    "actualCPUALlocation": 0,
    "prescript": "",
    "postscript": "",
    "username": "",
    "passwd": "",
    "nicNames": [
        {
            "deviceConfigId": 4000,
            "macAddress": "00:50:56:aa:b4:2c",
            "nicType": "VirtualVmxnet3",
            "interfaceLabel": "Network adapter 1",
            "pciSlotNumber": 192
        }
    ],
    "vmAllNicDataList": [
        {
            "deviceConfigId": "4000",
            "interfaceLabel": "Network adapter 1",
            "enforcedStaticIp": false,
            "enforcedStaticDnsIp": false,
            "gateway": "",
            "macAddress": "00:50:56:aa:b4:2c",
            "nicType": "VirtualVmxnet3",
            "pciSlotNumber": 192
        }
    ],
    "nicForStaticIp": "",
    "nicForDns": "",
    "defaultDnsIpToconfigure": [],
    "sourceIp": "172.29.10.210",
    "sourceSubnet": "",
    "sourceGateway": "172.29.10.100",
    "osName": "linux",
    "coresPerSocket": 1,
    "testCoresPerSocket": "1",
    "virtualSocket": 1,
    "testVirtualSocket": "1",
    "sequenceNumber": 0,
    "critical": false,
    "vappStartupOrder": 0,
    "vmalreadyPresentInRecPlan": false,
    "sequence": 1,
    "originalDisplayAppName": "Rhel1"
    }
    ],
    "appMembers": []
    }
    ],

```

```

    "organizationToList": [],
    "id": 7261
}

```

Response

```

{
  "status": "success",
  "statusCode": "1017",
  "statusMessage": "Application group updated successfully. ",
  "responseData": ""
}

```

Parameters	Description
sessionId	Session id as a header parameter.
uniqueKey	Value which is passed in fetch new member request.
AppGroupIdsToUpdate	Comma separated application group details for updating application object in Resiliency Director database.

fetchAppGroupDetailsWithVms

Use this API to fetch Application Group(s) with virtual machines details managed by the Resiliency Director from application database. If the method parameter is null, the method returns all the application groups with list of virtual machines from database. If the method parameter is not null, the method returns only the application group(s) whose id(s) have been provided as an input.

Method: GET

Request

<https://{HOST}/armnew/api/applicationgroup/fetchdetails> - To fetch all application groups from the database with virtual machines details (HTTPS).

<https://{HOST}/armnew/api/applicationgroup/fetchdetails?ids=1,2,3> - To fetch application group by id(s) with virtual machines details (HTTPS).

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 4102,
      "name": "A123",
      "description": "",
      "cdsId": 4097,
      "cdsHostName": "172.16.10.10",
      "serverId": 4099,
      "serverHostName": "172.16.10.134",
      "createdOn": 1421116451209,
      "editedOn": 1421116907013,
      "noOfVms": 1,
      "recoveryPlan": "",
      "noOfVApps": 0,
      "sequenceOrder": [
        {

```

```

"memberOrderIndex": 1,
"delay": 0,
"appGroupMembers": [
  {
    "memberUuId": "5008d31b-6f73-0098-9c1f-f8dc303b828f",
    "memberName": "Sky-VM3",
    "vm": true,
    "executionStatus": "",
    "cdsAppId": "5801",
    "cdsClusterId": "590021132584",
    "vmNewName": null,
    "id": 4108,
    "vappName": "",
    "numberOfCpus": 0,
    "actualNumberOfNics": 1,
    "testNumberOfNics": 1,
    "testMemoryInMbs": 1024,
    "actualMemoryInMbs": 1024,
    "storageLocation": null,
    "shutdownOrder": 0,
    "keyValue": null,
    "numberOfVirtualDisks": 0,
    "hddInGbs": 0,
    "hostName": "test-pri-arm",
    "ip": "null",
    "validForAppGroup": false,
    "cupResourceAllocation": {
      "reservation": 100,
      "limit": -1,
      "shares": 1000,
      "sharesLevel": 1
    },
    "memoryResourceAllocation": {
      "reservation": 100,
      "limit": -1,
      "shares": 10240,
      "sharesLevel": 1
    },
    "testCPUAllocation": 100,
    "actualCPUAllocation": 100,
    "prescript": "",
    "postscript": "",
    "username": "",
    "passwd": "",
    "nicNames": [
      {
        "deciveConfigId": 4000,
        "macAddress": "00:50:56:88:56:ba"
      }
    ],
    "selectedNIC": "",
    "deDupeAsyncStatus": null,
    "ipType": "dynamic",
    "gateway": null,
    "subnetMask": null,
    "sourceIp": "172.16.10.72",
    "sourceSubnet": "",
    "sourceGateway": "172.16.10.100",
    "portGroup": null,
    "osName": null,

```

```

        "preScriptType": "",
        "postScriptType": "",
        "appGroupLevelScript": null,
        "preExternalScript": [],
        "postExternalScript": [],
        "critical": false,
        "vappStartupOrder": 0
    }
},
"preExternalScript": [],
"postExternalScript": []
}
],
"armServerId": 0,
"armServerName": null,
"recoveryPlanId": 0,
"executionStatus": "",
"peerRecoveryPlanId": 0,
"cpuRequired": 100,
"memoryRequired": 1024,
"preScript": null,
"postScript": null,
"status": null,
"customerName": "",
"peerAppGrpId": 0,
"preScriptType": null,
"postScriptType": null,
"preScriptVMUUID": null,
"postScriptVMUUID": null,
"preExternalScript": [],
"postExternalScript": [],
"critical": false
}
]
}

```

Parameters	Description
ids	Comma separated application group id(s) that you want to fetch.

fetchCount

Use this API to fetch counts of application group(s) included and excluded in a recovery plan managed by the Resiliency Director from application database.

Method: GET

Request

https://{HOST}/armnew/api/applicationgroup/fetchcount - To fetch count for the application group from database (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "includeInRecovery": 2,
    "excludeFromRecovery": 0,
    "total": 2,
    "peerid": 10
  }
}
```

Parameters	Description
session	Session id as a header parameter.

getPreDefinedDelay

Use this API to return predefined delay for a sequence.

Method: GET

Request

https://{HOST}/armnew/api/applicationgroup/predefinedelay

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "",
  "responseData": {
    "preDefineDelay": [
      0,
      10,
      20,
      30,
      40,
      50
    ]
  }
}
```

Parameters	Description
sessionid	Session id as a header parameter.

updatePeerAppGroupCount

Use this API to update the application group count on peer Resiliency Director.

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/updatePeerAppGroupCount

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Application group count updated successfully on peer"
}
```

Parameters	Description
sessionId	Session id as a header parameter.

cancelAppGrpCreation

Use this API to cancel the cache while creating an application group.

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/cancelappgrpcreation

Body Param

```
{
  "includeInRecovery": 2,
  "excludeFromRecovery": 3,
  "total": 5,
  "peerId": 2201
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Application group creation canceled successfully."
}
```

Parameters	Description
sessionId	Session id as a header parameter.

updatePeerRecoveryPlanId

Use this API to update the application group on peer RD.

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/
updatePeerRecoveryPlanId?appnames=app_test&recoveryplanid=12

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Application group updated successfully on peer"
}
```

Parameters	Description
sessionid	Session id as header parameter.
appnames	Application group names as query parameter.
recoveryplanid	Recovery plan id, which is to be updated, as query parameter.

updatetargetclusterid

Use this API to update the target cluster id of the VM or application on RD Server based on target type

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/
updatetargetclusterid?uuid=34539535348957&targetclusterid=548444&target=VM&sourceclusterid=124
343

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": "Updated the targetclusterid successfully"
}
```

Parameters	Description
sessionid	Session id as header parameter.
UUID	UUID of the VM.
targetclusterid	Target cluster ID to be updated on the VM.
sourceclusterid	Source cluster ID on which the application or VM exists

Parameters	Description
target	VM or Application specified as a Target

fetchmappedcds

Use this API to retrieve the details of CDS mapped to an application group.

Method: GET

Request

https://{HOST}/armnew/api/applicationgroup/fetchmappedcds?appgroupid=21616

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 21607,
      "hostOrIp": "172.29.10.60",
      "username": "admin",
      "password": "",
      "port": 0,
      "useHttps": false,
      "firmwareVersion": "",
      "model": "",
      "vendorkey": "vendorKey",
      "clustername": "cluster_2",
      "listOrganizations": [],
      "selected": true,
      "emailConfigured": false,
      "authenticationSource": false,
      "clusterId": 590023229746
    },
    {
      "id": 21614,
      "hostOrIp": "172.29.11.150",
      "username": "admin",
      "password": "",
      "port": 0,
      "useHttps": false,
      "firmwareVersion": "",
      "model": "",
      "vendorkey": "vendorKey",
      "clustername": "Sky2",
      "listOrganizations": [],
      "selected": false,
      "emailConfigured": false,
      "authenticationSource": false,
      "clusterId": 1415062914
    }
  ]
}
```


Parameters	Description
sessionid	Session id as header parameter.
appgroupid	Application Group ID.

fetchapplicationchildmembers

Use this API to retrieve the child members of a given application (SQL instance and consistency group).

Method: GET

Request

`https://{HOST}/armnew/api/
fetchapplicationchildmembers?cdsappid=231634&appId=&cdsclusterid=1415039054&apptype=SqlInstanc
e&limit=25&offset=0`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "appgroupmemberid": 0,
      "groupid": 0,
      "appName": "C:\\",
      "apptype": "FileSystem",
      "isselectedforrecovery": false,
      "networkname": "",
      "pathname": "",
      "appClass": ""
    },
    {
      "appgroupmemberid": 0,
      "groupid": 0,
      "appName": "E:\\",
      "apptype": "FileSystem",
      "isselectedforrecovery": false,
      "networkname": "",
      "pathname": "",
      "appClass": ""
    },
    {
      "appgroupmemberid": 0,
      "groupid": 0,
      "appName": "msdb",
      "apptype": "SqlServerWriter",
      "isselectedforrecovery": false,
      "networkname": "",
      "pathname": "WIN-27I29HF0VIO",
      "appClass": "SQLServer"
    },
    {
      "appgroupmemberid": 0,
      "groupid": 0,
      "appName": "model",
      "apptype": "SqlServerWriter",

```

```

        "isselectedforrecovery": false,
        "networkname": "",
        "pathname": "WIN-27I29HF0VIO",
        "appClass": "SQLServer"
    }
]
}

```

Parameters	Description
sessionid	Session id as header parameter.
cdsappid	Actifio Appliance ID.
appid	Application ID.
cdsclusterid	Cluster ID.
apptype	Type of application.
limit	Number of records to fetch.
offset	Starting point to fetch the records.

fetchbackupdetails

Use this API to retrieve the backup details for a given application.

Method: GET

Request

<https://{HOST}/armnew/api/applicationgroup/fetchbackupdetails?cdsappid=231634&sourceclusterid=1415039054&backupdbdetails=master&action=exclude>

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "msdb": "C:\\\\",
    "model": "C:\\\\",
  }
}

```

Parameters	Description
sessionid	Session id as header parameter.
cdsappid	Actifio Appliance ID.
sourceclusterid	Cluster ID

Parameters	Description
action	Include or exclude options: • Include: Use this option to pull the backups mentioned in the backup details. • exclude: Use this option to pull the backups for all except the details mentioned in the backup details.
backupdbdetails	Name of the database.

Managing Organizations - IRestOrganization

This interface and its corresponding APIs allows you to execute REST requests to Actifio Resiliency Director Server for managing organizations. This interface contains the following RESTful APIs:

[createOrganization](#)
[deleteOrganizations](#)
[getCdsOrgs](#)
[getOrganizations](#)
[updateOrganization](#)

createOrganization

Use this API to create an organization in Resiliency Director database with the parameters provided in request.

Method: POST

Request

https://{HOST}/armnew/api/organization/add (HTTPS)

Body Param

```
{
  "name": "Org1 ",
  "description": "",
  "listCDSOrgs": [
    {
      "cdsOrgDescription": "New organization description",
      "cdsOrgId": "2942070",
      "cdsOrgName": "RD_QA_org",
      "childCdsOrgMapping": "false",
      "childOrgs": null,
      "createdate": "2015-10-23 01:56:51.032",
      "parentOrgId": 0
    }
  ],
  "listResource": [],
  "parentOrgId": "8231",
  "scheduleEnable": "true",
  "email": "user@testemail.com"
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}
```

Parameters	Description
sessionId	Session id as a header parameter.
orgToAdd	Organization to add in the Resiliency Director database.

deleteOrganizations

Use this API to delete organization(s) in the Resiliency Director database.

Method: GET

Request

<https://{HOST}/armnew/api/organization/remove?ids=1101> -(HTTPS)

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
ids	List of organization id(s) for which you want to fetch the data.
sessionId	Session id as a header parameter.

getCdsOrgs

Use this API to fetch CDS Orgs from Registered Actifio Appliance selected as authentication source. If the method parameter is null, the method will return all the cds orgs. If the method parameter is not null, the method will return only the cds-orgs(s) whose id(s) have been provided as an input.

Method - GET

Request

<https://{HOST}/armnew/api/organization/fetchcdsorgs?ids=4101,4109> - To fetch *cdsorgs* by id(s) HTTPS)

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
```

```

{
  "cdsOrgDescription": "New organization description",
  "cdsOrgId": "2942070",
  "cdsOrgName": "RD_QA_org",
  "childCdsOrgMapping": "false",
  "childOrgs": null,
  "createdate": "2015-10-23 01:56:51.032",
  "parentOrgId": 0
}
]
}

```

Parameters	Description
sessionid	Session id as a header parameter.
ids	List of organization id(s) for which you want to fetch the data.

getOrganizations

Use this API to fetch organizations from the Actifio Resiliency Director database. If the method parameter is null, the method will return all the organization from database. If the method parameter is not null, the method will return only the organization(s) whose id(s) have been provided as an input.

Method: GET

Request

<https://{HOST}/armnew/api/organization/fetch?> (HTTPS)

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "organizationId": 1101,
      "name": "org1",
      "description": "",
      "listCDSOrgs": [
        {
          "cdsOrgDescription": "New organization description",
          "cdsOrgId": "2942070",
          "cdsOrgName": "RD_QA_org",
          "childCdsOrgMapping": "false",
          "childOrgs": null,
          "createdate": "2015-10-23 01:56:51.032",
          "parentOrgId": 0
        }
      ],
      "createDate": "06/16/2015 03:51:19",
      "modifyDate": "06/16/2015 04:41:51",
      "childOrgs": [],
      "parentOrgId": 3,
      "scheduleEnable": true,
      "email": "user@testemail.com"
    }
  ],
}

```

```

{
  "organizationId": 1102,
  "name": "PUBLIC",
  "description": "PUBLIC organization every object in PUBLIC is accessible by everyone.",
  "listCDSOrgs": [
    {
      "cdsOrgDescription": "New organization description",
      "cdsOrgId": "2942070",
      "cdsOrgName": "RD_QA_org",
      "childCdsOrgMapping": "false",
      "childOrgs": null,
      "createdate": "2015-10-23 01:56:51.032",
      "parentOrgId": 0
    }
  ],
  "createDate": "06/16/2015 03:51:19",
  "modifyDate": "06/16/2015 04:41:51",
  "childOrgs": [
    {
      "organizationId": 1106,
      "name": "org1",
      "description": "organization1",
      "listCDSOrgs": [
        {
          "cdsOrgDescription": "New organization description",
          "cdsOrgId": "2942070",
          "cdsOrgName": "RD_QA_org",
          "childCdsOrgMapping": "false",
          "childOrgs": null,
          "createdate": "2015-10-23 01:56:51.032",
          "parentOrgId": 0
        }
      ],
      "createDate": "06/16/2015 03:51:19",
      "modifyDate": "06/16/2015 04:41:51",
      "childOrgs": [],
      "parentOrgId": 3,
      "scheduleEnable": true,
      "email": "user1@testemail.com"
    }
  ],
  "parentOrgId": 12,
  "scheduleEnable": true,
  "email": "user2@testemail.com"
}
]
}

```

Parameters	Description
sessionid	Session id as a header parameter.
ids	List of customer id(s) for which you want to fetch the data.

updateOrganization

Use this API to update an organization in the Resiliency Director database.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/organization/update (HTTPS)

Body Param

```
{
  "organizationId": 1101,
  "name": "Org1",
  "description": "test org",
  "listCDSOrgs": [
    {
      "cdsOrgDescription": "New organization description",
      "cdsOrgId": "2942070",
      "cdsOrgName": "RD_QA_org",
      "childCdsOrgMapping": "false",
      "childOrgs": null,
      "createdate": "2015-10-23 01:56:51.032",
      "parentOrgId": 0
    }
  ],
  "listResopurce": [],
  "parentOrgId": "8231",
  "scheduleEnable": "true",
  "email": "weqw@dfd.com"
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionId	Session id as a header parameter.
orgToUpdate	Organization to update in the Resiliency Director database.

Managing Recovery Plans – IRestRecoveryPlan

This interface and its corresponding REST APIs allows you to execute REST requests to the Resiliency Director Server for Recovery Plans. This interface contains the following RESTful APIs:

- getRecoveryPlans
- createRecoveryPlan
- updateRecoveryPlan
- deleteRecoveryPlans
- getRecoveryPlanAppGroup
- getAppGroupFromPrimary
- getAppGroupWithNewSequence
- getRecoveryPlanResourcePools
- executeRecoveryPlan
- getRecoveryplanState
- getAppGroupForEdit
- getAllStatesForRecoveryPlan
- resetRecoveryPlanState
- getRecoveryPlanCountGroupByState
- getRecoveryPlanSchedule
- getRecoveryPlanStatesForGivenIds
- invalidateRecoveryPlan
- getPortGroupsForRecoveryPlan
- fetchDedupeAsyncStatus
- fetchRecoveryPlansByDeDupeAsyncStatus
- cancelRecoveryPlanExecution
- disableRecoveryPlan
- getFetchInvalidRecoveryPlanNotification
- executeSingleVMRecovery
- executeSingleApplicationRecovery
- fetchVmExecutionStatus
- executeSingleScript
- fetchVMForInvalidRecoveryPlan
- fetchCompleteRecoveryPlanData
- executeonce

getRecoveryPlans

Use this API to fetch recovery plan(s) managed by the Actifio Resiliency Director Server from application database. If the method parameter is null, the method returns all the available recovery plans from database. If method parameter is not null, the method returns only the recovery plan(s) whose id(s) have been provided as an input.

Method: GET

Request

<https://{HOST}/armnew/api/recoveryplan/fetch?ids=1,2,3> - To fetch recovery plans by id(s) (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "recoveryPlanName": "RP",
      "recoveryPlanId": 13440,
      "managementServer": "172.29.11.105",
      "cdsIP": "RD_HYD_DEST",
      "resourcePoolName": "allVMs",
      "organizationList": [],
      "state": "Ready (info complete)",
      "editedOn": "2019-01-31 7:01:10",
      "editedBy": "",
      "cdsId": 0,
      "scheduleId": 13453,
      "serverid": 5207,
      "serverName": "root",
      "armServerId": 5209,
      "armServerName": "172.29.11.103",
      "portGroup": {
        "portGroupId": 13451,
        "portGroupName": "win2003-nw",
        "portGroupType": "Virtual Portgroup",
        "vSwitchName": "vSwitch0",
        "vSwitchUUID": "",
        "vLanId": 0,
        "hostName": "172.29.11.31",
        "portgroupDisplayName": "win2003-nw_vSwitch0",
        "orgId": 4,
        "virtualMachineId": 0,
        "inherited": true,
        "recoveryplanid": 13440
      },
      "resourcePoolId": "resgroup-11673",
      "nextFireTime": "",
      "scheduleStatus": "DISABLED",
      "disabled": false,
      "applicationsCount": "1 in 1 groups",
      "vmpprefixName": "Recovered_"
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
ids	Comma separated recovery plan id(s) for which you want to fetch the data.
state	State of the recovery plan to be fetched. The values for this parameter are as follows: • Ready (info complete) • Ready (info required) • In Progress • Failed • Success • Partial Success • Canceling • Canceled • Resetting • Reset Failed
mode	Mode in which recovery plan has been executed.

createRecoveryPlan

Use this API to create a recovery plan in the Resiliency Director Server database with the parameters provided in request. Request has to be in JSON format.

Method: POST

Request

`https://{HOST}/armnew/api/recoveryplan/add?suppresswarningforresourceaccessibility=true -- (HTTPS)`

Body Param

```
{
  "recoveryPlanMember": {
    "serverId": "5207",
    "vmPrefixName": "Recovered_",
    "organizationIds": [],
    "armServerId": "5209",
    "portGroup": {
      "portGroupId": 0,
      "portGroupName": "win2003-nw",
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vLanId": 0,
      "hostName": "172.29.11.31",
      "portgroupDisplayName": "win2003-nw_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 0,

```

```

    "clusterName": "Rd_Cluster",
    "inherited": true,
    "recoveryplanid": 0
  },
  "name": "RP",
  "storageServerId": null,
  "state": "",
  "editedOn": "3456789",
  "editedBy": "",
  "externalNetworkName": "",
  "routerLocation": "",
  "ipAddressPool": "",
  "memoryRequired": "",
  "cpuRequired": "",
  "nicsRequired": "",
  "resourcePools": [
    {
      "resourcePoolName": "allVMs",
      "uuid": "resgroup-11673"
    }
  ]
},
"appGroups": [
  {
    "startupSequence": 1,
    "appGroup": {
      "id": 13420,
      "name": "AG",
      "description": "",
      "cdsId": 0,
      "cdsHostName": "",
      "serverHostName": "",
      "createdOn": "12/31/1969 19:00:00",
      "editedOn": "01/31/2019 06:06:33",
      "noOfVms": 1,
      "noOfNonVms": 0,
      "recoveryPlan": "",
      "noOfVApps": 0,
      "sequenceOrder": [
        {
          "memberOrderIndex": 1,
          "delay": 0,
          "vmMembers": [
            {
              "memberUuId": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",
              "memberName": "Win2012r2-2_1548932687404",
              "vm": true,
              "executionStatus": "",
              "cdsAppId": "231511",
              "cdsClusterId": "1415039054",
              "targetClusterId": "1415040293",
              "clustername": "RD_HYD_DEST",
              "validForAppGroup": false,
              "displayAppName": "Win2012r2-2_1548932687404",
              "displayAppType": "VM",
              "apptype": "VM",
              "recoveryType": {
                "recoveryTypeId": 2,
                "name": "Create new VM from VMware VM",
                "recoveryTypeOrder": "1"
              }
            }
          ]
        }
      ]
    }
  }
]

```

```

    },
    "infoRequired": true,
    "id": 13422,
    "vappName": "",
    "numberOfCpus": 0,
    "actualNumberOfNics": 1,
    "testNumberOfNics": 1,
    "testMemoryInMbs": 4096,
    "actualMemoryInMbs": 4096,
    "shutdownOrder": 0,
    "numberOfVirtualDisks": 0,
    "hddInGbs": 0,
    "hostName": "winvwm",
    "ip": "",
    "targetClusterIp": "172.29.11.165",
    "cupResourceAllocation": {
        "reservation": 0,
        "limit": -1,
        "shares": 4000,
        "sharesLevel": 1,
        "numberOfCpus": 0,
        "corePerSocket": 0
    },
    "memoryResourceAllocation": {
        "reservation": 0,
        "limit": -1,
        "shares": 40960,
        "sharesLevel": 1,
        "numberOfCpus": 0,
        "corePerSocket": 0
    },
    "testCPUAllocation": 0,
    "actualCPUAllocation": 0,
    "prescript": "",
    "postscript": "",
    "username": "",
    "passwd": "",
    "nicNames": [
        {
            "deviceConfigId": 4000,
            "macAddress": "00:50:56:aa:e1:80",
            "nicType": "VirtualVmxnet3",
            "interfaceLabel": "Network adapter 1",
            "pciSlotNumber": 0
        }
    ],
    "vmAllNicDataList": [
        {
            "deviceConfigId": 4000,
            "macAddress": "00:50:56:aa:e1:80",
            "nicType": "VirtualVmxnet3",
            "interfaceLabel": "Network adapter 1",
            "pciSlotNumber": 0
        }
    ],
    "nicForStaticIp": "",
    "nicForDns": "",
    "defaultDnsIpToconfigure": [],
    "sourceIp": "",
    "sourceSubnet": "",

```

```

        "sourceGateway": "",
        "portGroup": [],
        "osName": "windows",
        "preExternalScript": [],
        "postExternalScript": [],
        "coresPerSocket": 1,
        "testCoresPerSocket": 1,
        "virtualSocket": 4,
        "testVirtualSocket": 4,
        "sequenceNumber": 1,
        "critical": false,
        "vappStartupOrder": 0,
        "vmAlreadyPresentInRecPlan": false,
        "resourcePool": null,
        "vmNewName": "Recovered_Win2012r2-2_1548932687404",
        "preScriptType": "",
        "postScriptType": "",
        "vmUserCredentials": {
            "id": 5240,
            "name": "Windows",
            "username": "Administrator",
            "password": "",
            "systemid": "31c73701-6b95-4f7d-9c7d-e79b69b930e0",
            "peercredentialsid": 0
        },
        "domainname": "",
        "vmOsHostname": "MyHost123",
        "domainCredentials": null
    }
],
    "appMembers": [],
    "preExternalScript": [],
    "postExternalScript": [],
    "resourcePool": null,
    "sequencePortGroup": null,
    "id": "id_169"
}
],
    "armServerId": 5209,
    "armServerName": "172.29.11.103",
    "recoveryPlanId": 0,
    "executionStatus": "",
    "peerRecoveryPlanId": 0,
    "cpuRequired": 0,
    "memoryRequired": 0,
    "customerName": "",
    "peerAppGrpId": 13420,
    "preExternalScript": [],
    "postExternalScript": [],
    "organizationToList": [],
    "startupSequence": 0,
    "critical": false
}
}
],
    "schedule": {
        "scheduleName": "asd",
        "scheduleStatus": "DISABLED",
        "scheduleType": "RECOVERY_PLAN",
        "startTime": "",

```

```

        "endTime": "",
        "repeatIntervalType": "",
        "repeatIntervalValue": "",
        "executeOnTO": null,
        "exceptOnTO": null
    }
}

```

Body Param (For VM in sequence 1 and CAF database – MySQL instance in sequence 2)

```

{
  "appGroups":[
    {
      "appGroup":{
        "agmId":0,
        "applianceGroupRequired":false,
        "armServerId":6246,
        "armServerName":"172.29.11.217",
        "cdsId":0,
        "cpuRequired":0,
        "createdOn":"12/31/1969 19:00:00",
        "critical":false,
        "customerName":"dummy_cust",
        "description":"",
        "editedOn":"05/04/2021 03:28:33",
        "executionStatus":"",
        "id":46144,
        "memoryRequired":0,
        "name":"92258_AG1",
        "noOfNonVms":1,
        "noOfVApps":0,
        "noOfVms":1,
        "organizationToList":[

        ],
        "peerAppGrpId":46144,
        "peerRecoveryPlanId":33899,
        "postExternalScript":[

        ],
        "preExternalScript":[

        ],
        "recoveryPlan":"92258_RP1",
        "recoveryPlanId":33899,
        "sequenceOrder":[
          {
            "appMembers":[

            ],
            "delay":0,
            "id":"id_6",
            "memberOrderIndex":1,
            "postExternalScript":[

            ],
            "preExternalScript":[

            ],
            "resourcePool":null,
            "sequencePortGroup":[

            ],

```

```

],
"vmMembers":[
  {
    "actualCPUAllocation":0,
    "actualMemoryInMbs":6144,
    "actualNumberOfNics":1,
    "agmAppId":"0",
    "appliancegroupid":0,
    "apptype":"VM",
    "bootdisksize":0,
    "cdsAppId":"1147444",
    "cdsClusterId":"145649790597",
    "clustername":"localhost.localdom",
    "consumedCloudTOList":[

    ],
    "consumedMigrationTOList":[

    ],
    "consumedNetworkTOList":[

    ],
    "coresPerSocket":1,
    "critical":false,
    "cupResourceAllocation":{
      "corePerSocket":0,
      "limit":-1,
      "numberOfCpus":0,
      "reservation":0,
      "shares":4000,
      "sharesLevel":1
    },
    "deDupeAsyncStatus":"GREEN",
    "defaultDnsIpToconfigure":[

    ],
    "displayAppName":"GTC_Mysql_Host",
    "displayAppType":"VM",
    "domainname":"",
    "executionStatus":"",
    "hddInGbs":0,
    "hostName":"GTC-Mysql",
    "id":52485,
    "infoRequired":false,
    "ip":"",
    "isAsm":false,
    "memberName":"GTC_Mysql_Host",
    "memberUuId":"5004e9ef-2f08-87da-d215-962ea995fe93",
    "memoryResourceAllocation":{
      "corePerSocket":0,
      "limit":-1,
      "numberOfCpus":0,
      "reservation":0,
      "shares":61440,
      "sharesLevel":1
    },
    "nicForDns":"",
    "nicForStaticIp":"",
    "nicNames":[

```

```

        {
            "deviceConfigId":4000,
            "interfaceLabel":"Network adapter 1",
            "macAddress":"00:50:56:84:7e:f0",
            "nicType":"VirtualVmxnet3",
            "pciSlotNumber":0
        }
    ],
    "numberOfCpus":4,
    "numberOfVirtualDisks":0,
    "originalBootdisksize":0,
    "osName":"linux",
    "passwd":"",
    "policyOpCode":0,
    "portGroup":[

    ],
    "postExternalScript":[

    ],
    "postscript":"vm_reset.sh",
    "postScriptType":"shell",
    "preExternalScript":[

    ],
    "prescript":"vm_failover.sh",
    "preScriptType":"shell",
    "recoveryType":{
        "description":"Add VMs you wish to recover at the DR site. Each VM
may be recovered only once and will retain its original host name.",
        "name":"Recover VM From VMWare VM Backup",
        "recoveryTypeId":1,
        "recoveryTypeOrder":"1"
    },
    "resourcePool":null,
    "sequenceNumber":1,
    "shutdownOrder":0,
    "sourceGateway":"",
    "sourceHostName":"GTC-Mysql",
    "sourceIp":"",
    "sourceSubnet":"",
    "targetClusterId":"142452487387",
    "targetClusterIp":"172.27.44.126",
    "testCoresPerSocket":1,
    "testCPUAllocation":0,
    "testMemoryInMbs":6144,
    "testNumberOfNics":1,
    "testVirtualSocket":4,
    "username":"",
    "validForAppGroup":false,
    "vappName":"",
    "vappStartupOrder":0,
    "virtualSocket":4,
    "vm":true,
    "vmAllNicDataList":[
        {
            "deviceConfigId":"4000",
            "enforcedStaticDnsIp":false,
            "enforcedStaticIp":false,
            "gateway":"",

```



```

        "interfaceLabel": "Network adapter 1",
        "macAddress": "00:50:56:84:7e:f0",
        "nicType": "VirtualVmxnet3",
        "pciSlotNumber": 192
    }
},
"vmAlreadyPresentInRecPlan": false,
"vmNewName": "recovered_92258_vm1GTC_Mysql_Host",
"vmOsHostname": "",
"vmUserCredentials": {
    "id": 24986,
    "name": "mysqlsandy",
    "password": "12!pass345",
    "peerCredentialsId": 0,
    "systemId": "0a0c2102-9fa9-442e-ae53-84ef9476bf4c",
    "username": "root"
}
}
},
{
    "appMembers": [
        {
            "appClass": "MYSQLGroup",
            "appClassMetaDataTOList": [

            ],
            "applicationChildMembers": [

            ],
            "appName": "mysqld_3066",
            "apptype": "MYSQLInstance",
            "cdsAppId": "1147379",
            "cdsClusterId": "145649790597",
            "clustername": "localhost.localdom",
            "critical": false,
            "depth": 0,
            "displayAppName": "mysqld_3066(GTC-Mysql-RootDB-Host)",
            "displayAppType": "MYSQLInstance",
            "executionStatus": "",
            "failoverState": 0,
            "friendlyType": "MYSQLInstance",
            "fromMount": false,
            "id": 52492,
            "infoRequired": false,
            "ip": "",
            "isAsm": false,
            "isClustered": false,
            "iscustomapp": true,
            "isGroup": false,
            "isTbd": false,
            "memberName": "mysqld_3066",
            "networkName": "",
            "originalAppId": 0,
            "policyOpCode": 0,
            "provisioningParameters": "<provisioningoptions><PORT>3066</
PORT><OSUSER>mysql</OSUSER><BASEDIR>/usr/local/mysql-8.0.22-el7-x86_64</BASEDIR></
provisioningoptions>",
            "recoveryType": {
                "description": "Select MySQL applications to recover at the DR site",

```

```

        "name": "Recover MYSQL Instance",
        "recoveryTypeId": 6263,
        "recoveryTypeOrder": "1"
    },
    "scriptParameters": "name=/act/scripts/pre.sh:phase=PRE:timeout=0;name=/
act/scripts/post.sh:phase=POST:timeout=0",
    "scriptParameterTOList": [
        {
            "phase": "PRE",
            "scriptName": "/act/scripts/pre.sh",
            "timeout": "0"
        },
        {
            "phase": "POST",
            "scriptName": "/act/scripts/post.sh",
            "timeout": "0"
        }
    ],
    "sequenceNumber": 2,
    "sourceHostId": 0,
    "sourceHostName": "GTC-Mysql-RootDB-Host",
    "targetClusterId": "142452487387",
    "targetHostId": 0,
    "targetHostName": "GTC_Mysql_Host",
    "validForAppGroup": false,
    "vm": false,
    "vmAlreadyPresentInRecPlan": false
    }
],
"delay": 0,
"id": "id_7",
"memberOrderIndex": 2,
"postExternalScript": [

],
"preExternalScript": [

],
"vmMembers": [

]
}
],
"serverHostName": "systst2-vc.sqa.aktifio.com",
"serverId": 10713,
"startupSequence": 1
},
"startupSequence": 1
}
],
"recoveryPlanMember": {
    "armServerId": 6246,
    "cpuRequired": "",
    "editedBy": "",
    "editedOn": "3456789",
    "externalNetworkName": "",
    "ipAddressPool": "",
    "memoryRequired": "",
    "name": "92258_RP1",
    "nicsRequired": "",

```

```

"organizationIds":[
],
"portGroup":[
{
  "clusterName":"RD_Cluster_Dswitches",
  "hostName":"172.27.44.21",
  "inherited":true,
  "mode":"both",
  "orgId":0,
  "portgroupDisplayName":"DPortGroup_MGMTSwitch",
  "portGroupId":0,
  "portGroupKey":"dvportgroup-424",
  "portGroupName":"DPortGroup",
  "portGroupType":"Distributed Virtual Portgroup",
  "recoveryplanid":0,
  "virtualMachineId":0,
  "vLanId":0,
  "vSwitchName":"MGMTSwitch",
  "vSwitchUUID":"50 04 c7 63 f8 11 4a aa-b4 63 97 fc 0c f5 d9 5e"
}
],
"recoveryPlanId":33899,
"resourcePools":[
{
  "resourcePoolName":"RD_Pool",
  "uuid":"resgroup-71906"
}
],
"routerLocation":"","
"serverId":10713,
"state":"","
"storageServerId":null,
"vmprefixName":"recovered_92258_vm1"
},
"schedule":{
  "endTime":"","
  "exceptOnTO":null,
  "executeOnTO":null,
  "repeatIntervalType":"","
  "resetAfterExecution":false,
  "scheduleName":"asd",
  "scheduleStatus":"DISABLED",
  "scheduleType":"RECOVERY_PLAN",
  "startTime":""
}
}

```

Response

```

{
  "status": "success",
  "statusCode": "1021",
  "statusMessage": "Recovery plan added successfully. ",
  "responseData": []
}

```

Parameters	Description
sessionid	Session id passed as header parameter.
recoveryplanToAdd	Recovery Plan details to be added in Resiliency Director database.
suppressWarningForResourceAccessibility	Parameter passed as a query parameter to suppress warning for the resource accessibility.

updateRecoveryPlan

Use this API to update a recovery plan in the Resiliency Director Server database with the parameters provided in request.

Method: POST

Request

`https://{HOST}:{PORT}/armnew/api/recoveryplan/
update?suppresswarningforresourceaccessibility=true -(HTTPS)`

Body Param

```
{
  "recoveryPlanMember": {
    "serverId": 5207,
    "vmprefixName": "Recovered_",
    "organizationIds": [],
    "armServerId": 5209,
    "portGroup": {
      "portGroupId": 0,
      "portGroupName": "win2003-nw",
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vLanId": 0,
      "hostName": "172.29.11.31",
      "portgroupDisplayName": "win2003-nw_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 0,
      "clusterName": "Rd_Cluster",
      "inherited": true,
      "recoveryplanid": 0
    },
    "name": "RP",
    "storageServerId": null,
    "state": "",
    "editedOn": "3456789",
    "editedBy": "",
    "externalNetworkName": "",
    "routerLocation": "",
    "ipAddressPool": "",
    "memoryRequired": "",
    "cpuRequired": "",
    "nicsRequired": "",
    "resourcePools": [
      {
        "resourcePoolName": "allVMs",
        "uuid": "resgroup-11673"
      }
    ]
  }
}
```

```

    ],
    "recoveryPlanId": 13440
  },
  "appGroups": [
    {
      "startupSequence": 1,
      "appGroup": {
        "id": 13441,
        "name": "AG",
        "description": "",
        "cdsId": 0,
        "cdsHostName": "",
        "serverId": 5207,
        "serverHostName": "172.29.11.105",
        "createdOn": "12/31/1969 19:00:00",
        "editedOn": "01/31/2019 06:06:33",
        "noOfVms": 1,
        "noOfNonVms": 0,
        "recoveryPlan": "RP",
        "noOfVApps": 0,
        "sequenceOrder": [
          {
            "memberOrderIndex": 1,
            "delay": 0,
            "vmMembers": [
              {
                "memberUuid": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",
                "memberName": "Win2012r2-2_1548932687404",
                "vm": true,
                "executionStatus": "",
                "cdsAppId": "231511",
                "cdsClusterId": "1415039054",
                "vmNewName": "Recovered_Win2012r2-2_1548932687404",
                "targetClusterId": "1415040293",
                "clustername": "RD_HYD_DEST",
                "validForAppGroup": false,
                "displayAppName": "Win2012r2-2_1548932687404",
                "displayAppType": "VM",
                "apptype": "VM",
                "recoveryType": {
                  "recoveryTypeId": 2,
                  "name": "Create new VM from VMware VM",
                  "recoveryTypeOrder": "1"
                }
              },
              {
                "infoRequired": false,
                "id": 13445,
                "vappName": "",
                "numberOfCpus": 0,
                "actualNumberOfNics": 1,
                "testNumberOfNics": 1,
                "testMemoryInMbs": 4096,
                "actualMemoryInMbs": 4096,
                "shutdownOrder": 0,
                "numberOfVirtualDisks": 0,
                "hddInGbs": 0,
                "hostName": "winvwm",
                "ip": "",
                "targetClusterIp": "172.29.11.165",
                "cupResourceAllocation": {
                  "reservation": 0,

```

```

    "limit": -1,
    "shares": 4000,
    "sharesLevel": 1,
    "numberOfCpus": 0,
    "corePerSocket": 0
  },
  "memoryResourceAllocation": {
    "reservation": 0,
    "limit": -1,
    "shares": 40960,
    "sharesLevel": 1,
    "numberOfCpus": 0,
    "corePerSocket": 0
  },
  "testCPUAllocation": 0,
  "actualCPUAllocation": 0,
  "prescript": "",
  "postscript": "",
  "username": "",
  "passwd": "",
  "nicNames": [
    {
      "deviceConfigId": 4000,
      "macAddress": "00:50:56:aa:e1:80",
      "nicType": "VirtualVmxnet3",
      "interfaceLabel": "Network adapter 1",
      "pciSlotNumber": 0
    }
  ],
  "vmAllNicDataList": [
    {
      "deviceConfigId": "4000",
      "interfaceLabel": "Network adapter 1",
      "enforcedStaticIp": false,
      "enforcedStaticDnsIp": false,
      "gateway": "",
      "macAddress": "00:50:56:aa:e1:80",
      "nicType": "VirtualVmxnet3",
      "pciSlotNumber": 0
    }
  ],
  "nicForStaticIp": "",
  "nicForDns": "",
  "defaultDnsIpToconfigure": [],
  "deDupeAsyncStatus": "GREEN",
  "sourceIp": "",
  "sourceSubnet": "",
  "sourceGateway": "",
  "portGroup": [],
  "osName": "windows",
  "preScriptType": "",
  "postScriptType": "",
  "preExternalScript": [],
  "postExternalScript": [],
  "coresPerSocket": 1,
  "testCoresPerSocket": 1,
  "virtualSocket": 4,
  "testVirtualSocket": 4,
  "domainname": "",
  "vmOsHostname": "MyHost123",

```

```

        "vmUserCredentials": {
            "id": 5240,
            "name": "Windows",
            "username": "Administrator",
            "password": "maheshwar",
            "systemid": "31c73701-6b95-4f7d-9c7d-e79b69b930e0",
            "peercredentialsid": 0
        },
        "sequenceNumber": 1,
        "critical": false,
        "vappStartupOrder": 0,
        "vmalreadyPresentInRecPlan": false,
        "resourcePool": null
    }
},
"appMembers": [],
"preExternalScript": [],
"postExternalScript": [],
"resourcePool": null,
"sequencePortGroup": null,
"id": "id_257"
}
],
"armServerId": 5209,
"armServerName": "172.29.11.103",
"recoveryPlanId": 13440,
"executionStatus": "",
"peerRecoveryPlanId": 0,
"cpuRequired": 0,
"memoryRequired": 0,
"customerName": "dummy_cust",
"peerAppGrpId": 13441,
"preExternalScript": [],
"postExternalScript": [],
"organizationToList": [],
"startupSequence": 1,
"critical": false
}
}
],
"schedule": {
    "scheduleName": "asd",
    "scheduleStatus": "DISABLED",
    "scheduleType": "RECOVERY_PLAN",
    "startTime": "",
    "endTime": "",
    "repeatIntervalType": "",
    "repeatIntervalValue": "",
    "executeOnT0": null,
    "exceptOnT0": null
}
}
}

```

Response:

```

{
    "status": "success",
    "statusCode": "1000",
    "statusMessage": "Operation performed successfully.",
    "responseData": []
}

```

Parameters	Description
sessionid	Session id passed as header parameter.
recoveryplansToUpdate	Recovery plan details that you want to update in Resiliency Director database.
suppressWarningForResourceAccessibility	Parameter is passed as a query parameter to suppress warning for the resource accessibility

deleteRecoveryPlans

Use this API to delete a recovery plan(s) in the Actifio Resiliency Director database with the parameters provided in request.

Method: GET

Request:

`https://{HOST}/armnew/api/recoveryplan/remove?ids=1,2,3 (HTTPS)`

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully. Number of deleted Recovery plan(s): 1",
  "responseData": ""
}
```

Parameters	Description
sessionid	Session id as a header parameter.
ids	Comma separated id(s) of the recovery plan that you want to delete.

getRecoveryPlanAppGroup

Use this API to fetch recovery plan(s) managed by Resiliency Director from the application database. If the method parameter is null, the method returns all the available application groups from database. If the method parameter is not null, the method returns only the application group(s) whose recovery plan id have been provided as an input.

Method: GET

Request:

`https://{HOST}:{PORT}/armnew/api/recoveryplan/fetchappgroup?id=1.`

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "startupSequence": 1,

```



```

"appGroup": {
  "id": 8235,
  "name": "test",
  "description": "",
  "cdsId": 5121,
  "cdsHostName": "172.29.10.70",
  "serverId": 5126,
  "serverHostName": "172.29.11.105",
  "createdOn": "11/09/2015 03:05:27",
  "editedOn": "11/09/2015 03:05:27",
  "noOfVms": 0,
  "recoveryPlan": "soma1",
  "noOfVApps": 0,
  "sequenceOrder": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "appGroupMembers": [
        {
          "memberUuId": "502a79d4-c8c1-6adf-0350-4fed92ebe351",
          "memberName": "RD-QA-10-152",
          "vm": true,
          "executionStatus": "",
          "cdsAppId": "7221",
          "cdsClusterId": "590027424048",
          "vmNewName": "soma1_RD-QA-10-152",
          "id": 8239,
          "vappName": "",
          "numberOfCpus": 0,
          "actualNumberOfNics": 1,
          "testNumberOfNics": 1,
          "testMemoryInMbs": 2048,
          "actualMemoryInMbs": 2048,
          "storageLocation": null,
          "shutdownOrder": 0,
          "keyValue": null,
          "numberOfVirtualDisks": 0,
          "hddInGbs": 0,
          "hostName": "localhost.localdomain",
          "ip": "",
          "validForAppGroup": false,
          "cupResourceAllocation": {
            "reservation": 200,
            "limit": -1,
            "shares": 1000,
            "sharesLevel": 1
          },
          "memoryResourceAllocation": {
            "reservation": 0,
            "limit": -1,
            "shares": 10240,
            "sharesLevel": 1
          },
          "testCPUALlocation": 200,
          "actualCPUALlocation": 200,
          "prescript": "",
          "postscript": "",
          "username": "",
          "passwd": "",
          "nicNames": [

```

```

    {
      "deviceConfigId": 4000,
      "macAddress": "00:50:56:aa:c5:6b",
      "nicType": "VirtualVmxnet3",
      "interfaceLabel": "Network adapter 1"
    }
  ],
  "vmAllNicDataList": [
    {
      "ipType": null,
      "deviceConfigId": "4000",
      "enforcedStaticIp": false,
      "enforcedStaticDnsIp": false,
      "ipAddressToConfigure": null,
      "gateway": "",
      "subnetMask": null,
      "macAddress": "00:50:56:aa:c5:6b",
      "nicType": "VirtualVmxnet3"
    }
  ],
  "nicForStaticIp": "",
  "nicForDns": "",
  "defaultDnsIpToconfigure": [],
  "deDupeAsyncStatus": "GREEN",
  "ipType": "",
  "defaultGateway": "",
  "subnetMask": "",
  "sourceIp": "172.29.10.137",
  "sourceSubnet": "",
  "sourceGateway": "172.29.10.100",
  "portGroup": [
    {
      "portGroupId": 8240,
      "portGroupName": "soma1",
      "portGroupKey": "null",
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vSwitchUuid": "null",
      "vLanId": 4095,
      "hostName": "172.29.11.33",
      "portgroupDisplayName": "soma1_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 0,
      "macAddress": null,
      "nicType": null,
      "clusterName": null
    }
  ],
  "osName": null,
  "preScriptType": "",
  "postScriptType": "",
  "appGroupLevelScript": null,
  "preExternalScript": [],
  "postExternalScript": [],
  "resourcePool": null,
  "critical": false,
  "vappStartupOrder": 0,
  "vmAlreadyPresentInRecPlan": false
}
],

```

```

        "preExternalScript": [],
        "postExternalScript": [],
        "resourcePool": null,
        "sequencePortGroup": {
            "portGroupId": 8237,
            "portGroupName": "soma1",
            "portGroupKey": "null",
            "portGroupType": "Virtual Portgroup",
            "vSwitchName": "vSwitch0",
            "vSwitchUUID": "null",
            "vLanId": 4095,
            "hostName": "172.29.11.33",
            "portgroupDisplayName": "soma1_vSwitch0",
            "orgId": 0,
            "virtualMachineId": 0,
            "macAddress": null,
            "nicType": null,
            "clusterName": null
        }
    },
    ],
    "armServerId": 6156,
    "armServerName": "172.29.11.110",
    "recoveryPlanId": 8234,
    "executionStatus": "",
    "peerRecoveryPlanId": 0,
    "cpuRequired": 200,
    "memoryRequired": 2048,
    "preScript": null,
    "postScript": null,
    "status": null,
    "customerName": "dummy_cust",
    "peerAppGrpId": 6166,
    "preScriptType": null,
    "postScriptType": null,
    "preScriptVMUUID": null,
    "postScriptVMUUID": null,
    "preExternalScript": [],
    "postExternalScript": [],
    "organizationToList": [],
    "critical": false
}
}
]
}

```

Parameters	Description
ids	Recovery plan id(s) whose application group data you want to fetch.
sessionid	Session id passed as header parameter.

getAppGroupFromPrimary

Use this API to fetch application group(s) managed by the Actifio Resiliency Director Collector from application database. If the method parameter is null, method does not return any application group data. If the method parameter is not null, the method returns only the application group(s) whose Resiliency Director Collector id(s) have been provided as an input.

Method: GET

Request:

https://{HOST}/armnew/api/recoveryplan/getappgroupfromprimary?armserverid=1.

Response:

```
{
  "recoveryPlanMember": {
    "serverId": 5207,
    "vmprefixName": "Recovered_",
    "organizationIds": [],
    "armServerId": 5209,
    "portGroup": {
      "portGroupId": 0,
      "portGroupName": "win2003-nw",
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vLanId": 0,
      "hostName": "172.29.11.31",
      "portgroupDisplayName": "win2003-nw_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 0,
      "clusterName": "Rd_Cluster",
      "inherited": true,
      "recoveryplanid": 0
    },
    "name": "RP",
    "storageServerId": null,
    "state": "",
    "editedOn": "3456789",
    "editedBy": "",
    "externalNetworkName": "",
    "routerLocation": "",
    "ipAddressPool": "",
    "memoryRequired": "",
    "cpuRequired": "",
    "nicsRequired": "",
    "resourcePools": [
      {
        "resourcePoolName": "allVMs",
        "uuid": "resgroup-11673"
      }
    ],
    "recoveryPlanId": 13440
  },
  "appGroups": [
    {
      "startupSequence": 1,
      "appGroup": {
        "id": 13441,
        "name": "AG",
        "description": "",
        "cdsId": 0,
        "cdsHostName": "",

```

```

"serverId": 5207,
"serverHostName": "172.29.11.105",
"createdOn": "12/31/1969 19:00:00",
"editedOn": "01/31/2019 06:06:33",
"noOfVms": 1,
"noOfNonVms": 0,
"recoveryPlan": "RP",
"noOfVApps": 0,
"sequenceOrder": [
  {
    "memberOrderIndex": 1,
    "delay": 0,
    "vmMembers": [
      {
        "memberUuid": "502a1d8c-fc41-7f24-e147-ee38ff20e39b",
        "memberName": "Win2012r2-2_1548932687404",
        "vm": true,
        "executionStatus": "",
        "cdaAppId": "231511",
        "cdaClusterId": "1415039054",
        "vmNewName": "Recovered_Win2012r2-2_1548932687404",
        "targetClusterId": "1415040293",
        "clustername": "RD_HYD_DEST",
        "validForAppGroup": false,
        "displayAppName": "Win2012r2-2_1548932687404",
        "displayAppType": "VM",
        "apptype": "VM",
        "recoveryType": {
          "recoveryTypeId": 2,
          "name": "Create new VM from VMware VM",
          "recoveryTypeOrder": "1"
        },
        "infoRequired": false,
        "id": 13445,
        "vappName": "",
        "numberOfCpus": 0,
        "actualNumberOfNics": 1,
        "testNumberOfNics": 1,
        "testMemoryInMbs": 4096,
        "actualMemoryInMbs": 4096,
        "shutdownOrder": 0,
        "numberOfVirtualDisks": 0,
        "hddInGbs": 0,
        "hostName": "winvwm",
        "ip": "",
        "targetClusterIp": "172.29.11.165",
        "cupResourceAllocation": {
          "reservation": 0,
          "limit": -1,
          "shares": 4000,
          "sharesLevel": 1,
          "numberOfCpus": 0,
          "corePerSocket": 0
        },
        "memoryResourceAllocation": {
          "reservation": 0,
          "limit": -1,
          "shares": 40960,
          "sharesLevel": 1,
          "numberOfCpus": 0,

```

```

    "corePerSocket": 0
  },
  "testCPUAllocation": 0,
  "actualCPUAllocation": 0,
  "prescript": "",
  "postscript": "",
  "username": "",
  "passwd": "",
  "nicNames": [
    {
      "deviceConfigId": 4000,
      "macAddress": "00:50:56:aa:e1:80",
      "nicType": "VirtualVmxnet3",
      "interfaceLabel": "Network adapter 1",
      "pciSlotNumber": 0
    }
  ],
  "vmAllNicDataList": [
    {
      "deviceConfigId": "4000",
      "interfaceLabel": "Network adapter 1",
      "enforcedStaticIp": false,
      "enforcedStaticDnsIp": false,
      "gateway": "",
      "macAddress": "00:50:56:aa:e1:80",
      "nicType": "VirtualVmxnet3",
      "pciSlotNumber": 0
    }
  ],
  "nicForStaticIp": "",
  "nicForDns": "",
  "defaultDnsIpToconfigure": [],
  "deDupeAsyncStatus": "GREEN",
  "sourceIp": "",
  "sourceSubnet": "",
  "sourceGateway": "",
  "portGroup": [],
  "osName": "windows",
  "preScriptType": "",
  "postScriptType": "",
  "preExternalScript": [],
  "postExternalScript": [],
  "coresPerSocket": 1,
  "testCoresPerSocket": 1,
  "virtualSocket": 4,
  "testVirtualSocket": 4,
  "domainname": "",
  "vmOsHostname": "MyHost123",
  "vmUserCredentials": {
    "id": 5240,
    "name": "Windows",
    "username": "Administrator",
    "password": "maheshwar",
    "systemid": "31c73701-6b95-4f7d-9c7d-e79b69b930e0",
    "peercredentialsid": 0
  },
  "sequenceNumber": 1,
  "critical": false,
  "vappStartupOrder": 0,
  "vmalreadyPresentInRecPlan": false,

```

```

        "resourcePool": null
    }
],
"appMembers": [],
"preExternalScript": [],
"postExternalScript": [],
"resourcePool": null,
"sequencePortGroup": null,
"id": "id_257"
}
],
"armServerId": 5209,
"armServerName": "172.29.11.103",
"recoveryPlanId": 13440,
"executionStatus": "",
"peerRecoveryPlanId": 0,
"cpuRequired": 0,
"memoryRequired": 0,
"customerName": "dummy_cust",
"peerAppGrpId": 13441,
"preExternalScript": [],
"postExternalScript": [],
"organizationToList": [],
"startupSequence": 1,
"critical": false
}
}
],
"schedule": {
    "scheduleName": "asd",
    "scheduleStatus": "DISABLED",
    "scheduleType": "RECOVERY_PLAN",
    "startTime": "",
    "endTime": "",
    "repeatIntervalType": "",
    "repeatIntervalValue": "",
    "executeOnT0": null,
    "exceptOnT0": null
}
}
}

```

Parameters	Description
sessionid	Session id as a header parameter.
armserverid	Resiliency Director Collector id whose data you want to fetch.

getAppGroupWithNewSequence

Use this API to fetch application group(s) managed by the Actifio Resiliency Director Collector from application database. If the method parameter is null, method does not return any application group data. If the method parameter is not null, the method returns only the application group(s) whose Resiliency Director Collector id(s) have been provided as an input.

Method: POST

Request

https://{HOST}/armnew/api/recoveryplan/
getappgroupwithvmdetailsfromprimary?armserverid=5131&isedit=true&recoveryPlanId=7400

Body Param

```
[
  {
    "appGroup": {
      "id": 7401,
      "armServerId": 5131,
      "armServerName": "172.29.11.135"
    },
    "startupSequence": 1
  }
]
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "appGroupsList": [
      {
        "startupSequence": 1,
        "appGroup": {
          "id": 7365,
          "name": "AGVM",
          "description": "",
          "cdsId": 0,
          "cdsHostName": "",
          "serverId": 5124,
          "serverHostName": "172.29.11.105",
          "createdOn": "01/01/1970 00:00:00",
          "editedOn": "02/05/2019 01:19:26",
          "noOfVms": 1,
          "noOfNonVms": 0,
          "recoveryPlan": "",
          "noOfVApps": 0,
          "sequenceOrder": [
            {
              "memberOrderIndex": 1,
              "delay": 0,
              "vmMembers": [
                {
                  "memberUuId": "502ab92d-7e57-8f43-891c-07a07d85e429",
                  "memberName": "Rhel1",
                  "vm": true,
                  "executionStatus": "",
                  "cdsAppId": "106809",
                  "cdsClusterId": "1415039054",
                  "targetClusterId": "1415040293",
                  "clustername": "RD_HYD_DEST",
                  "validForAppGroup": false,
                  "displayAppName": "Rhel1",
                  "displayAppType": "VM",
                  "apptype": "VM",
                  "recoveryType": {
                    "recoveryTypeId": 1,
                    "name": "Recover VM From VMWare VM Backup",

```



```

    "recoveryTypeOrder": "1"
  },
  "infoRequired": false,
  "id": 7396,
  "vappName": "",
  "numberOfCpus": 0,
  "actualNumberOfNics": 1,
  "testNumberOfNics": 1,
  "testMemoryInMbs": 2048,
  "actualMemoryInMbs": 2048,
  "shutdownOrder": 0,
  "numberOfVirtualDisks": 0,
  "hddInGbs": 0,
  "hostName": "localhost.localdomain",
  "ip": "",
  "targetClusterIp": "172.29.11.165",
  "cupResourceAllocation": {
    "reservation": 0,
    "limit": -1,
    "shares": 1000,
    "sharesLevel": 1,
    "numberOfCpus": 0,
    "corePerSocket": 0
  },
  "memoryResourceAllocation": {
    "reservation": 0,
    "limit": -1,
    "shares": 20480,
    "sharesLevel": 1,
    "numberOfCpus": 0,
    "corePerSocket": 0
  },
  "testCPUAllocation": 0,
  "actualCPUAllocation": 0,
  "prescript": "",
  "postscript": "",
  "username": "",
  "passwd": "",
  "nicNames": [
    {
      "deviceConfigId": 4000,
      "macAddress": "00:50:56:aa:b4:2c",
      "nicType": "VirtualVmxnet3",
      "interfaceLabel": "Network adapter 1",
      "pciSlotNumber": 0
    }
  ],
  "vmAllNicDataList": [
    {
      "deviceConfigId": "4000",
      "interfaceLabel": "Network adapter 1",
      "enforcedStaticIp": false,
      "enforcedStaticDnsIp": false,
      "gateway": "",
      "macAddress": "00:50:56:aa:b4:2c",
      "nicType": "VirtualVmxnet3",
      "pciSlotNumber": 0
    }
  ],
  "nicForStaticIp": "",

```

```

        "nicForDns": "",
        "defaultDnsIpToconfigure": [],
        "sourceIp": "",
        "sourceSubnet": "",
        "sourceGateway": "",
        "portGroup": [],
        "osName": "linux",
        "preExternalScript": [],
        "postExternalScript": [],
        "coresPerSocket": 1,
        "testCoresPerSocket": 1,
        "virtualSocket": 1,
        "testVirtualSocket": 1,
        "sequenceNumber": 1,
        "critical": false,
        "vappStartupOrder": 0,
        "vmalreadyPresentInRecPlan": false
    }
},
    "appMembers": [],
    "preExternalScript": [],
    "postExternalScript": []
}
],
    "armServerId": 5131,
    "armServerName": "172.29.11.135",
    "recoveryPlanId": 0,
    "executionStatus": "",
    "peerRecoveryPlanId": 0,
    "cpuRequired": 0,
    "memoryRequired": 0,
    "customerName": "",
    "peerAppGrpId": 7365,
    "preExternalScript": [],
    "postExternalScript": [],
    "organizationToList": [],
    "startupSequence": 1,
    "critical": false
}
}
],
    "maxDNSEntries": 10
}
}

```

Parameters	Description
id	Recovery plan id whose data you want to fetch.
armServerId	Arm server id selected in recovery plan.
sessionId	Session id for user.
recoveryPlanAppGroupWithSequence	Application group with sequence.

Parameters	Description
isCallForEdit	Boolean for deciding call for edit recovery plan.
recoveryPlanId	Recovery plan id.

getRecoveryPlanResourcePools

Use this API to fetch Resource Pool(s) of hypervisor. If the method parameter is null, the method returns all the available resource pools from database. If the method parameter is not null, the method returns only the resource pool(s) whose name(s) have been provided as an input.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/getresourcepool?serverid=2

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "resourcePoolId": "resgroup-1538",
      "resourcePoolName": "DEv_RP2",
      "resourcePoolMemory": "0",
      "resourcePoolCPU": "0",
      "resourcePoolParentName": "Resources",
      "resourcePoolOwnerName": "RD",
      "resourcePoolUuid": null,
      "resourcePoolIsInvalid": false
    },
    {
      "resourcePoolId": "resgroup-2634",
      "resourcePoolName": "RP-ND",
      "resourcePoolMemory": "17475",
      "resourcePoolCPU": "3913",
      "resourcePoolParentName": "Resources",
      "resourcePoolOwnerName": "RD",
      "resourcePoolUuid": null,
      "resourcePoolIsInvalid": false
    }
  ]
}
```

Parameters	Description
sessionId	Session id as header parameter.
serverId	Server id whose data is to be fetched.
resourcePoolName	Specific name for resource pool. If this name is not null, resource pool will get listed even it is a part of other recovery plan.

executeRecoveryPlan

Use this API to execute a recovery plan with the provided id.

Method: POST

Request

https://{HOST}/armnew/api/recoveryplan/execute?id=4148&mode=TEST (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionId	Session id as header parameter.
id	Id of the recovery plan to be execute.
mode	Recovery mode used to execute a recovery plan.

getRecoveryplanState

Use this API to fetch the state for a recovery plan.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/fetchstate?id=4148.

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "postExternalScript": "null",
    "status": "Ready(info complete)",
    "preExternalScript": "null",
    "statusMessage": "Recovery plan : 'test_recovery' completed execution with status : Success
on : Mon Feb 20 00:43:47 EST 2014"
  }
}
```

Parameters	Description
sessionId	Session id as a header parameter.
id	Recovery plan id for which you want to fetch recovery state.

getAppGroupForEdit

Use this API to fetch application group(s) managed by the Actifio Resiliency Director Collector to edit a recovery plan. If the method parameter is null, method does not return any application group data. If the method parameter is not null, the method returns only the application group(s) whose Resiliency Director Collector id(s) have been provided as an input.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/
getappgroupfromprimaryforedit?armserverid=1,recoveryplanid=2.

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "startupSequence": 0,
      "appGroup": {
        "id": 7317,
        "name": "AGAll",
        "description": "",
        "cdsId": 0,
        "cdsHostName": "",
        "serverHostName": "",
        "createdOn": "01/01/1970 00:00:00",
        "editedOn": "02/05/2019 01:19:26",
        "noOfVms": 0,
        "noOfNonVms": 0,
        "recoveryPlan": "",
        "noOfVApps": 0,
        "sequenceOrder": [
          {
            "memberOrderIndex": 1,
            "delay": 0,
            "vmMembers": [
              {
                "memberUuId": "502a2cbc-245f-078b-81cc-592eaf59f175",
                "memberName": "Win2012r2-3_1549273300949",
                "vm": true,
                "executionStatus": "",
                "cdsAppId": "233265",
                "cdsClusterId": "1415039054",
                "targetClusterId": "1415040293",
                "clustername": "RD_HYD_DEST",
                "validForAppGroup": false,
                "displayAppName": "Win2012r2-3_1549273300949",
                "displayAppType": "VM",
                "apptype": "VM",
                "recoveryType": {
                  "recoveryTypeId": 2,
                  "name": "Create new VM from VMware VM",
                  "recoveryTypeOrder": "1"
                },
                "infoRequired": false,
                "id": 7380,
                "vappName": "",
                "numberOfCpus": 0,

```

```

"actualNumberOfNics": 1,
"testNumberOfNics": 1,
"testMemoryInMbs": 4096,
"actualMemoryInMbs": 4096,
"shutdownOrder": 0,
"numberOfVirtualDisks": 0,
"hddInGbs": 0,
"hostName": "winvwm",
"ip": "",
"cupResourceAllocation": {
  "reservation": 0,
  "limit": -1,
  "shares": 4000,
  "sharesLevel": 1,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"memoryResourceAllocation": {
  "reservation": 0,
  "limit": -1,
  "shares": 40960,
  "sharesLevel": 1,
  "numberOfCpus": 0,
  "corePerSocket": 0
},
"testCPUAllocation": 0,
"actualCPUAllocation": 0,
"prescript": "",
"postscript": "",
"username": "",
"passwd": "",
"nicNames": [
  {
    "deviceConfigId": 4000,
    "macAddress": "00:50:56:aa:f5:e3",
    "nicType": "VirtualVmxnet3",
    "interfaceLabel": "Network adapter 1",
    "pciSlotNumber": 0
  }
],
"vmAllNicDataList": [
  {
    "deviceConfigId": "4000",
    "interfaceLabel": "Network adapter 1",
    "enforcedStaticIp": false,
    "enforcedStaticDnsIp": false,
    "gateway": "",
    "macAddress": "00:50:56:aa:f5:e3",
    "nicType": "VirtualVmxnet3",
    "pciSlotNumber": 0
  }
],
"nicForStaticIp": "",
"nicForDns": "",
"defaultDnsIpToconfigure": [],
"sourceIp": "",
"sourceSubnet": "",
"sourceGateway": "",
"portGroup": [],
"osName": "windows",

```

```

        "preScriptType": "",
        "postScriptType": "",
        "preExternalScript": [],
        "postExternalScript": [],
        "coresPerSocket": 1,
        "testCoresPerSocket": 1,
        "virtualSocket": 4,
        "testVirtualSocket": 4,
        "domainname": "",
        "vmOsHostname": "Host1",
        "vmUserCredentials": {
            "id": 7173,
            "name": "Win",
            "username": "Administrator",
            "password": "maheshwar",
            "systemid": "8a10f297-a384-4a73-85fc-c257feea94ed",
            "peercredentialsid": 0
        },
        "sequenceNumber": 1,
        "critical": false,
        "vappStartupOrder": 0,
        "vmAlreadyPresentInRecPlan": false
    }
],
"appMembers": [],
"preExternalScript": [],
"postExternalScript": []
},
{
    "memberOrderIndex": 2,
    "delay": 0,
    "vmMembers": [],
    "appMembers": [
        {
            "memberName": "C:\\",
            "vm": false,
            "executionStatus": "",
            "cdsAppId": "6328",
            "cdsClusterId": "1415039054",
            "targetClusterId": "1415040293",
            "clustername": "RD_HYD_DEST",
            "validForAppGroup": false,
            "displayAppName": "C:\\ (win2012_sql_new) ",
            "displayAppType": "File System",
            "applicationChildMembers": [],
            "apptype": "FileSystem",
            "sourceHostName": "win2012_sql_new",
            "recoveryType": {
                "recoveryTypeId": 5,
                "name": "Recover Filesystem",
                "recoveryTypeOrder": "2"
            },
            "infoRequired": false,
            "id": 7384,
            "appName": "C:\\",
            "originalAppId": 0,
            "sourceHostId": 0,
            "appClass": "",
            "friendlyType": "FileSystem",
            "networkName": ""
        }
    ]
}

```

```

        "targetHostName": "Win2012r2-3_1549273300949",
        "targetHostId": 0,
        "depth": 0,
        "isGroup": false,
        "isClustered": false,
        "fromMount": false,
        "failoverState": 0,
        "isTbd": false,
        "restoreOptions": "volume#C:\\,targetvolume#N:\\",
        "scriptParameters": "name=:phase=PRE:timeout=;name=:phase=POST:timeout=",
        "volumesToList": [
            {
                "sourceVolume": "C:\\",
                "targetVolume": "N:\\",
            }
        ],
        "scriptParameterTOList": [
            {
                "scriptName": "",
                "phase": "PRE",
                "timeout": ""
            },
            {
                "scriptName": "",
                "phase": "POST",
                "timeout": ""
            }
        ],
        "appClassMetaDataTOList": [],
        "targetHostNameList": [
            "recovwery_shiva",
            "Recovered_Rhel1"
        ],
        "sequenceNumber": 2,
        "critical": false,
        "vmAlreadyPresentInRecPlan": false
    }
},
"preExternalScript": [],
"postExternalScript": []
},
{
    "memberOrderIndex": 3,
    "delay": 0,
    "vmMembers": [],
    "appMembers": [
        {
            "memberName": "WINVVM",
            "vm": false,
            "executionStatus": "",
            "cdsAppId": "231634",
            "cdsClusterId": "1415039054",
            "targetClusterId": "1415040293",
            "clustername": "RD_HYD_DEST",
            "validForAppGroup": false,
            "displayAppName": "WINVVM (win2012r2-1) ",
            "displayAppType": "SQL Instance",
            "applicationChildMembers": [],
            "apptype": "SqlInstance",
            "sourceHostName": "win2012r2-1",

```



```

    "recoveryType": {
      "recoveryTypeId": 4,
      "name": "Recover SQL Server Database",
      "recoveryTypeOrder": "2"
    },
    "infoRequired": false,
    "id": 7386,
    "appName": "WINVVM",
    "originalAppId": 0,
    "sourceHostId": 0,
    "appClass": "SQLServerGroup",
    "friendlyType": "SqlInstance",
    "networkName": "",
    "targetHostName": "Win2012r2-3_1549273300949",
    "targetHostId": 0,
    "depth": 0,
    "isGroup": false,
    "isClustered": false,
    "fromMount": false,
    "failoverState": 0,
    "provisioningParameters": "<provisioningoptions><ConsistencyGroupName>ConGrp1</ConsistencyGroupName><sqlinstance>WINVVM</sqlinstance><dbnameprefix>pre</dbnameprefix><dbnamesuffix>post</dbnamesuffix><recover>true</recover><userlogins>false</userlogins></provisioningoptions>",
    "isTbd": false,
    "appClassMetaDataTOList": [
      {
        "appClassPropertiesTO": {
          "propertyId": 7217,
          "name": "SQLServerGroup",
          "propertyName": "ConsistencyGroupName",
          "propertyType": "STRING",
          "propertyLabel": "Name of Consistency Group",
          "isRequired": true,
          "description": "Specify name for the new Consistency Group that will be
created as a result of app aware mount. The name must be unique across the appliance."
        },
        "appid": 7386,
        "parameterId": 7217,
        "value": "ConGrp1"
      },
      {
        "appClassPropertiesTO": {
          "propertyId": 7218,
          "name": "SQLServerGroup",
          "propertyName": "sqlinstance",
          "propertyType": "EDITABLE_SELECT",
          "propertyLabel": "SQL Server Instance Name",
          "isRequired": true,
          "description": "Select the target SQL Server Instance. The new databases
will be managed by the selected SQL Server instance."
        },
        "appid": 7386,
        "parameterId": 7218,
        "value": "WINVVM"
      },
      {
        "appClassPropertiesTO": {
          "propertyId": 7219,
          "name": "SQLServerGroup",

```

```

        "propertyName": "dbnameprefix",
        "propertyType": "STRING",
        "propertyLabel": "Prefix for SQL Server Database Name",
        "isRequired": false,
        "description": "Specify prefix to be used for the newly provisioned
database names. This string will be prefixed to the original database name to create new
database name. Valid characters include letters, numbers, @, #, -, , and a trailing space
character."
    },
    "appid": 7386,
    "parameterId": 7219,
    "value": "pre"
},
{
    "appClassPropertiesT0": {
        "propertyId": 7220,
        "name": "SQLServerGroup",
        "propertyName": "dbnamesuffix",
        "propertyType": "STRING",
        "propertyLabel": "Suffix for SQL Server Database Name",
        "isRequired": false,
        "description": "Specify suffix to be used for the newly provisioned
database names. This string will be appended to the original database name to create new
database name. Valid characters include letters, numbers, @, #, -, , and a leading space
character."
    },
    "appid": 7386,
    "parameterId": 7220,
    "value": "post"
},
{
    "appClassPropertiesT0": {
        "propertyId": 7221,
        "name": "SQLServerGroup",
        "propertyName": "recover",
        "propertyType": "BOOLEAN",
        "propertyLabel": "Recover Database After Restore",
        "isRequired": false,
        "description": "If selected, brings the newly created database online."
    },
    "appid": 7386,
    "parameterId": 7221,
    "value": "true"
},
{
    "appClassPropertiesT0": {
        "propertyId": 7222,
        "name": "SQLServerGroup",
        "propertyName": "userlogins",
        "propertyType": "BOOLEAN",
        "propertyLabel": "Recover User Logins",
        "isRequired": false,
        "description": "If selected, recover the user logins of the database. This
may require user credentials to be entered."
    },
    "appid": 7386,
    "parameterId": 7222,
    "value": "false"
},
{

```

```

        "appClassPropertiesT0": {
            "propertyId": 7223,
            "name": "SQLServerGroup",
            "propertyName": "username",
            "propertyType": "STRING",
            "propertyLabel": "User Name",
            "isRequired": false,
            "description": "Specify user name for database provisioning. Only required
when the account running the connector (typically 'Local System') does not have privileges to
apply transaction logs, or to detach a database (required during subsequent unmount).",
        },
        "appid": 7386,
        "parameterId": 7223
    },
    {
        "appClassPropertiesT0": {
            "propertyId": 7224,
            "name": "SQLServerGroup",
            "propertyName": "password",
            "propertyType": "ENCRYPT",
            "propertyLabel": "Password",
            "isRequired": false,
            "description": "Password for the specified user."
        },
        "appid": 7386,
        "parameterId": 7224
    }
],
"inclusionList": "rdtest2,rdtest3",
"targetHostNameList": [
    "recovery_shiva",
    "Recovered_Rhel1"
],
"sequenceNumber": 3,
"critical": false,
"vmAlreadyPresentInRecPlan": false
}
],
"preExternalScript": [],
"postExternalScript": []
}
],
"armServerId": 5131,
"armServerName": "172.29.11.135",
"recoveryPlanId": 0,
"executionStatus": "",
"peerRecoveryPlanId": 0,
"cpuRequired": 0,
"memoryRequired": 0,
"customerName": "",
"peerAppGrpId": 7243,
"preExternalScript": [],
"postExternalScript": [],
"organizationToList": [],
"startupSequence": 0,
"critical": false
}
},
{
    "startupSequence": 1,

```

```

"appGroup": {
  "id": 7401,
  "name": "AGVM",
  "description": "",
  "cdsId": 0,
  "cdsHostName": "",
  "serverId": 5124,
  "serverHostName": "172.29.11.105",
  "createdOn": "01/01/1970 00:00:00",
  "editedOn": "02/05/2019 01:19:26",
  "noOfVms": 0,
  "noOfNonVms": 0,
  "recoveryPlan": "RP1",
  "noOfVApps": 0,
  "sequenceOrder": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "vmMembers": [
        {
          "memberUuId": "502ab92d-7e57-8f43-891c-07a07d85e429",
          "memberName": "Rhel1",
          "vm": true,
          "executionStatus": "",
          "cdsAppId": "106809",
          "cdsClusterId": "1415039054",
          "vmNewName": "Recovered_Rhel1",
          "targetClusterId": "1415040293",
          "clustername": "RD_HYD_DEST",
          "validForAppGroup": false,
          "displayAppName": "Rhel1",
          "displayAppType": "VM",
          "apptype": "VM",
          "recoveryType": {
            "recoveryTypeId": 1,
            "name": "Recover VM From VMWare VM Backup",
            "recoveryTypeOrder": "1"
          },
          "infoRequired": false,
          "id": 7405,
          "vappName": "",
          "numberOfCpus": 0,
          "actualNumberOfNics": 1,
          "testNumberOfNics": 1,
          "testMemoryInMbs": 2048,
          "actualMemoryInMbs": 2048,
          "shutdownOrder": 0,
          "numberOfVirtualDisks": 0,
          "hddInGbs": 0,
          "hostName": "localhost.localdomain",
          "ip": "",
          "cupResourceAllocation": {
            "reservation": 0,
            "limit": -1,
            "shares": 1000,
            "sharesLevel": 1,
            "numberOfCpus": 0,
            "corePerSocket": 0
          },
          "memoryResourceAllocation": {

```

```

    "reservation": 0,
    "limit": -1,
    "shares": 20480,
    "sharesLevel": 1,
    "numberOfCpus": 0,
    "corePerSocket": 0
  },
  "testCPUAllocation": 0,
  "actualCPUAllocation": 0,
  "prescript": "",
  "postscript": "",
  "username": "",
  "passwd": "",
  "nicNames": [
    {
      "deviceConfigId": 4000,
      "macAddress": "00:50:56:aa:b4:2c",
      "nicType": "VirtualVmxnet3",
      "interfaceLabel": "Network adapter 1",
      "pciSlotNumber": 0
    }
  ],
  "vmAllNicDataList": [
    {
      "deviceConfigId": "4000",
      "interfaceLabel": "Network adapter 1",
      "enforcedStaticIp": false,
      "enforcedStaticDnsIp": false,
      "gateway": "",
      "macAddress": "00:50:56:aa:b4:2c",
      "nicType": "VirtualVmxnet3",
      "pciSlotNumber": 0
    }
  ],
  "nicForStaticIp": "",
  "nicForDns": "",
  "defaultDnsIpToconfigure": [],
  "deDupeAsyncStatus": "GREEN",
  "sourceIp": "",
  "sourceSubnet": "",
  "sourceGateway": "",
  "portGroup": [
    {
      "portGroupId": 7407,
      "portGroupName": "win2003-nw",
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vSwitchUUID": "",
      "vLanId": 0,
      "hostName": "172.29.11.31",
      "portgroupDisplayName": "win2003-nw_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 7405,
      "macAddress": "00:50:56:aa:b4:2c",
      "nicType": "VirtualVmxnet3",
      "inherited": true,
      "recoveryplanid": 7400
    },
    {
      "portGroupId": 7406,

```

```

        "portGroupName": "win2003-nw",
        "portGroupType": "Virtual Portgroup",
        "vSwitchName": "vSwitch0",
        "vSwitchUUID": "",
        "vLanId": 0,
        "hostName": "172.29.11.31",
        "portgroupDisplayName": "win2003-nw_vSwitch0",
        "orgId": 0,
        "virtualMachineId": 7405,
        "inherited": true,
        "recoveryplanid": 7400
    }
],
"osName": "linux",
"preExternalScript": [],
"postExternalScript": [],
"coresPerSocket": 1,
"testCoresPerSocket": 1,
"virtualSocket": 1,
"testVirtualSocket": 1,
"sequenceNumber": 1,
"critical": false,
"vappStartupOrder": 0,
"vmalreadyPresentInRecPlan": false
}
],
"appMembers": [],
"preExternalScript": [],
"postExternalScript": [],
"sequencePortGroup": {
    "portGroupId": 7403,
    "portGroupName": "win2003-nw",
    "portGroupType": "Virtual Portgroup",
    "vSwitchName": "vSwitch0",
    "vSwitchUUID": "",
    "vLanId": 0,
    "hostName": "172.29.11.31",
    "portgroupDisplayName": "win2003-nw_vSwitch0",
    "orgId": 0,
    "virtualMachineId": 0,
    "inherited": true,
    "recoveryplanid": 7400
}
}
],
"armServerId": 5131,
"armServerName": "172.29.11.135",
"recoveryPlanId": 7400,
"executionStatus": "",
"peerRecoveryPlanId": 0,
"cpuRequired": 0,
"memoryRequired": 0,
"customerName": "dummy_cust",
"peerAppGrpId": 7261,
"preExternalScript": [],
"postExternalScript": [],
"organizationToList": [],
"startupSequence": 1,
"critical": false
}

```

```
}
]
}
```

Parameters	Description
recoveryplanid	Recovery plan id whose data you want to fetch.
armserverid	Resiliency Director Collector id whose data you want to fetch.
sessionid	Session id as header parameter.

getAllStatesForRecoveryPlan

Use this API to fetch recovery plan execution status details managed by CSP from application database.

Method: GET

Request

<https://{HOST}/armnew/api/recoveryplan/getexecallstates?recoveryplanid=2> - To fetch all recovery execution status from database (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 0,
      "executionStartDateTime": 456464,
      "executionEndDateTime": 454564,
      "overallExecutionStatus": "ReadyToRun",
      "detailedExecutionStatus": "DetailsEexecutionstatus",
      "typeOfRecoveryPlanExecution": "type"
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
recoveryplanid	Recovery plan id of the data to be fetched.

resetRecoveryPlanState

Use this API to reset the status of a recovery plan for the provided id.

Method: POST

Request

<https://{HOST}/armnew/api/recoveryplan/resetstate?id=2&state=Success>

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionid	Session id as a header parameter.
id	Recovery plan Id that you want to reset.
state	Recovery plan state that you want to update.

getRecoveryPlanCountGroupByState

Use this API to fetch recovery plan count by providing execution status details as an input. If the method parameter is null, method does not return invalid recovery plan input. If the method parameter is not null, the method returns only the recovery plan execution status(es) count whose recovery plan id have been provided as an input.

Method: GET

Request

`https://{HOST}:{PORT}/armnew/api/recoveryplan/fetchcount?status=ReadyToRun`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "readyToRun": 1,
      "draft": 1,
      "inValidation": 0,
      "total": 2
    }
  ]
}
```

Parameters	Description
sessionid	Session id as header parameter.
status	The recovery plan status for which you want to fetch count.

getRecoveryPlanSchedule

Use this API to fetch schedule details for a recovery plan managed by CSP from the application database. If the method parameter is null, method does not return invalid recovery plan input. If the method parameter is not null, the method returns the schedule details only for recovery plan(s) whose id(s) have been provided as an input.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/fetchschedule?recoveryplanid=1

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    "schedule":
    {
      "scheduleName": "ScheduleName",
      "scheduleStatus": "DISABLED",
      "scheduleType": "RECOVERY_PLAN",
      "startTime": 46464,
      "endTime": 6464,
      "repeatIntervalType": "234234",
      "repeatIntervalValue": "4234",
      "executeOnT0": {
        "frequencyType": "",
        "frequencyInterval": "",
        "daysOfWeek": "",
        "daysOfMonth": "",
        "monthsOfYear": "",
        "weekOfMonth": ""
      },
      "exceptOnT0": {
        "frequencyType": "",
        "frequencyInterval": "",
        "daysOfWeek": "",
        "daysOfMonth": "",
        "monthsOfYear": "",
        "weekOfMonth": ""
      }
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
recoveryplanid	Recovery plan id for which you want to fetch schedule.

getRecoveryPlanStatesForGivenIds

Use this API to fetch recovery plan state details managed by CSP from application database.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/fetchstatebyrecoveryids?ids=1,2,3 - To fetch recovery plan execution state by recovery plan id(s) from database (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 232,
      "state": "ReadyToRun"
    }
  ]
}
```

Parameters	Description
sessionId	Session id as a header parameter.
ids	Recovery plan ids whose state you want to fetch.

invalidateRecoveryPlan

Use this API to invalidate recovery plans for the given list of recovery plan ids.

Method: GET

Request

http://{HOST}/armnew/api/recoveryplan/invalidate?ids=1

Body Param

```
{
  "id": 1244,
  "message": "ApplicationGroup is invalid",
  "createdOn": 123344,
  "recoveryPlanId": 1
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Recovery Plan has been invalidated successfully",
  "responseData": ""
}
```

Parameters	Description
sessionId	Session id as a header parameter.
ids	Recovery plan ids that you want to invalidate.

getPortGroupsForRecoveryPlan

Use this API to fetch port groups for resource pools for hypervisor.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/
fetchportgroups?serverid=3&resourcepoolid=8&portgroupname=port1

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "portGroupId": 0,
      "portGroupName": "QA_Test",
      "portGroupKey": null,
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vSwitchUUID": null,
      "vLanId": 0,
      "hostName": "172.29.10.29",
      "portgroupDisplayName": "QA_Test_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 0,
      "macAddress": null,
      "nicType": null,
      "clusterName": "RD"
    },
    {
      "portGroupId": 0,
      "portGroupName": "ND-Default-Portgroup",
      "portGroupKey": null,
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vSwitchUUID": null,
      "vLanId": 0,
      "hostName": "172.29.10.29",
      "portgroupDisplayName": "ND-Default-Portgroup_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 0,
      "macAddress": null,
      "nicType": null,
      "clusterName": "RD"
    }
  ]
}
```

Parameters	Description
serverid	Server id from where you want to list port groups.
resourcepoolid	Resource pools id whose port groups you want to list.
portGroupName	Port group name that you want to execute from the list.

Parameters	Description
vSwchName	vSwitch name as query parameter.
orgId	Organization id as query parameter.
sessionId	Session id passed as header parameter.

fetchDedupeAsyncStatus

Use this API to get dedup async and stream snap status for recovery plans.

Method: GET

Request

<https://{HOST}/armnew/api/recoveryplan/fetchdedupeasyncstatus>

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "green": 3,
    "yellow": 0,
    "red": 0
  }
}
```

Parameters	Description
sessionId	Session id as a header parameter.

fetchRecoveryPlansByDeDupeAsyncStatus

Use this API to fetch recovery plans using dedup async and stream snap status.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/fetchbydedupeasyncstatus?status=GREEN

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "recoveryPlanName": "RD_1678",
      "recoveryPlanId": 5165,
      "deploymentPlatform": "platform",
      "managementServer": "172.29.10.179",
      "cdsIP": "172.29.10.80",
      "nextSchedule": "",
      "resourcePoolName": "DEV_RP1",
      "organizationList": [
        {
          "id": 5155,
          "name": "HR",
          "description": null,
          "createdate": null,
          "modifydate": null,
          "email": null,
          "scheduleEnable": false,
          "parentOrgId": 0,
          "listCDSOrgs": null,
          "childOrgs": null
        }
      ],
      "state": "Success",
      "editedOn": "11/16/2015 04:31:33",
      "editedBy": "",
      "cdsId": 5127,
      "scheduleId": 5250,
      "serverid": 5138,
      "serverName": "root",
      "armServerId": 5123,
      "armServerName": "172.29.10.161",
      "portGroup": null,
      "resourcePoolId": "resgroup-1537",
      "deDupeAsyncStatus": "GREEN",
      "nextFireTime": "",
      "scheduleStatus": null,
      "disabled": false,
      "executionMode": "TEST"
    },
    {
      "recoveryPlanName": "rr",
      "recoveryPlanId": 6178,
      "deploymentPlatform": "platform",
      "managementServer": "172.29.10.179",
      "cdsIP": "172.29.10.80",
      "nextSchedule": "",

```

```

    "resourcePoolName": "soma",
    "organizationList": [],
    "state": "ReadyToRun",
    "editedOn": "11/16/2015 05:57:59",
    "editedBy": "",
    "cdsId": 5127,
    "scheduleId": 6191,
    "serverid": 5138,
    "serverName": "root",
    "armServerId": 5123,
    "armServerName": "172.29.10.161",
    "portGroup": null,
    "resourcePoolId": "resgroup-2921",
    "deDupeAsyncStatus": "GREEN",
    "nextFireTime": "",
    "scheduleStatus": null,
    "disabled": false,
    "executionMode": null
  }
]
}

```

Parameters	Description
sessionId	Session id as a header parameter.
status	The deDupAsyncStatus of the recovery plans.

cancelRecoveryPlanExecution

Use this API to cancel the recovery plan execution managed by the Actifio Resiliency Director from application database. If the method parameter is null, the method returns invalid input exception. If the method parameter is not null, the method cancels only the recovery plan whose id is provided as an input.

Method: GET

Request

<https://{HOST}/armnew/api/recoveryplan/cancel?id=1>.

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}

```

Parameters	Description
id	Recovery plan id whose execution you want to cancel.
sessionId	Session id passed as header parameter.

disableRecoveryPlan

Use this API to disable a recovery plan managed by the Actifio Resiliency Director from application database. If the method parameter is null, the method returns invalid input exception. If the method parameter is not null, the method disables only the recovery plan whose id have been provided as an input.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/recoveryplan/disable?id=1

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionid	Session id passed as header parameter.
id	Recovery plan id that you want to disable.

getFetchInvalidRecoveryPlanNotification

Use this API to fetch notifications for a state of recovery plans.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/fetchinvalidrecoveryplannotifications?recoveryplanids=1
- To fetch recovery plans by id (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 45,
      "messege": "Recovery plan has been invalidated due to change in application group",
      "createdOn": " ",
      "recoveryPlanId": 1
    }
  ]
}
```

Parameters	Description
recoveryplanids	Recovery plan id for which you want to fetch notifications.
sessionid	Session id passed as header parameter.

executeSingleVMRecovery

Use this API to execute recovery of a single virtual machine with the provided virtual machine id and the recovery plan id.

Method: POST

Request

https://{HOST}/armnew/api/recoveryplan/executesinglevm?recoveryplanid=1&vmid=21

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionid	Session id as a header parameter.
recoveryplanid	Recovery plan id of which virtual machine is a part.
vmid	Virtual machine id that you want to recover.

executeSingleApplicationRecovery

Use this API to execute recovery of a single application with the provided application id and the recovery plan id it is a part of.

Method: POST

Request

https://{HOST}/armnew/api/recoveryplan/
executesingleapplication?recoveryplanid=86032&applicationid=86048

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionid	Session id as a header parameter.
recoveryplanid	Recovery plan id of which virtual machine is a part.
applicationid	Application id that you want to recover.

fetchVmExecutionStatus

Use this API to fetch virtual machine execution status details for the given application group id.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/fetchvmexecutionstatus?applicationgroupid=1

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "preExternalScript": [],
      "postExternalScript": [],
      "vmExecutionDetails": [
        {
          "vmId": 5134,
          "vmUuid": "502aee74-a330-3e10-eddf-d584ff387d51",
          "state": "",
          "executionMessage": "",
          "vmName": "_RD-win2012-106-152",
          "vAppName": "",
          "preExternalScript": [],
          "postExternalScript": [],
          "critical": false
        },
        {
          "vmId": 5137,
          "vmUuid": "5008d7f2-4642-23c9-ddba-beedae996970",
          "state": "",
          "executionMessage": "",
          "vmName": "_RD-Dev-22-152",
          "vAppName": "",
          "preExternalScript": [],
          "postExternalScript": [],
          "critical": false
        }
      ]
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
applicationgroupid	Application group id of which virtual machine is a part.

executeSingleScript

Use this API to execute script individually.

Method: POST

Request

https://{HOST}/armnew/api/recoveryplan/executesinglescript

Body Param

```
{
  "recoveryPlanId": 123,
  "scriptId": 126,
  "objectId": 125,
  "executionLevel": "application group",
  "scriptExecutionType": "pre"
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Single external script execution started successfully."
}
```

Parameters	Description
sessionid	Session id as a header parameter.
scriptDetail	Script details that you want to execute.

fetchVMForInvalidRecoveryPlan

Use this API to return list of VMs available in an invalid recovery plan.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/recoveryplan/fetchvmforinvalidrecoveryplan

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "peerRecoveryPlanId": "4152",
    "listVMUUIDs": [
      "5036e87c-70ab-99e4-57d7-4feba7716d89",
      "5036dd4a-295c-b772-7a37-3e52575be153"
    ]
  }
}
```

Parameters	Description
sessionid	Session id as a header parameter.

fetchCompleteRecoveryPlanData

Use this API to get the complete details of the recovery plan, and this API is used in runonce of recovery plan.

Method: GET

Request

https://{HOST}/armnew/api/recoveryplan/fetchcompleteterecoveryplandata?id=8279&_=1548599047787

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "recoveryPlanMember": {
      "recoveryPlanId": 8279,
      "storageServerId": 0,
      "armServerId": 5209,
      "name": "RPAppOnly",
      "state": "Ready (info required)",
      "editedOn": 1548599115187,
      "editedBy": "",
      "routerLocation": "",
      "ipAddressPool": "",
      "externalNetworkName": "",
      "memoryRequired": 0,
      "cpuRequired": 0,
      "nicsRequired": 0,
      "cpuReservationRequired": 0,
      "memoryReservationRequired": 0,
      "portGroup": {
        "portGroupId": 0,
        "vlanId": 0,
        "orgId": 0,
        "virtualMachineId": 0,
        "inherited": true,
        "recoveryplanid": 0
      },
      "organizationIds": [],
      "resourcePools": [
        {
          "memoryReservation": 0,
          "cpuReservation": 0,
          "cpuLimit": 0,
          "memoryLimit": 0,
          "inherited": true,
          "resourcePoolId": 0,
          "virtualmachineid": 0,
          "recoveryplanid": 0
        }
      ],
      "vmprefixName": ""
    },
    "appGroups": [
      {
        "startupSequence": 1,
        "appGroup": {
          "id": 8280,
          "name": "AG2",

```

```

"description": "",
"cdsId": 0,
"cdsHostName": "",
"serverHostName": "",
"createdOn": "12/31/1969 19:00:00",
"editedOn": "01/27/2019 09:23:29",
"noOfVms": 0,
"noOfNonVms": 0,
"recoveryPlan": "RPAAppOnly",
"noOfVApps": 0,
"sequenceOrder": [
  {
    "memberOrderIndex": 1,
    "delay": 0,
    "vmMembers": [],
    "appMembers": [
      {
        "memberName": "msdb",
        "vm": false,
        "executionStatus": "",
        "cdsAppId": "6330",
        "cdsClusterId": "1415039054",
        "targetClusterId": "1415040293",
        "clustername": "RD_HYD_DEST",
        "validForAppGroup": false,
        "displayAppName": "msdb (win2012_sql_new) ",
        "displayAppType": "SQL Database",
        "applicationChildMembers": [],
        "apptype": "SqlServerWriter",
        "sourceHostName": "win2012_sql_new",
        "recoveryType": {
          "recoveryTypeId": 4,
          "name": "Recover SQL Server Database",
          "recoveryTypeOrder": "2"
        },
        "infoRequired": true,
        "id": 8282,
        "appName": "msdb",
        "originalAppId": 0,
        "sourceHostId": 0,
        "appClass": "SQLServer",
        "friendlyType": "SQLServer",
        "networkName": "",
        "targetHostId": 0,
        "depth": 0,
        "isGroup": false,
        "isClustered": false,
        "fromMount": false,
        "failoverState": 0,
        "isTbd": false,
        "appClassMetaDataTOList": [],
        "targetHostNameList": [
          "recovery_shiva"
        ],
        "sequenceNumber": 1,
        "critical": false,
        "vmalreadyPresentInRecPlan": false
      }
    ],
    "preExternalScript": [],

```

```

        "postExternalScript": []
    },
    ],
    "armServerId": 5209,
    "armServerName": "172.29.11.103",
    "recoveryPlanId": 8279,
    "executionStatus": "",
    "peerRecoveryPlanId": 0,
    "cpuRequired": 0,
    "memoryRequired": 0,
    "customerName": "dummy_cust",
    "peerAppGrpId": 5221,
    "preExternalScript": [],
    "postExternalScript": [],
    "organizationToList": [],
    "startupSequence": 1,
    "critical": false
}
}
]
}
}
}

```

Parameters	Description
sessionid	Session id as a header parameter.
id	Id of the recovery plan.

executeonce

Use this API to execute the recovery plan one time by configuring the applications.

Method: POST

Request

<https://{HOST}/armnew/api/recoveryplan/executeonce?execmode=Test>

Body Param

```

{
  "recoveryPlanMember": {
    "recoveryPlanId": 8279,
    "storageServerId": 0,
    "armServerId": 5209,
    "name": "RPAppOnly",
    "state": "Ready (info required)",
    "editedOn": 1548599115187,
    "editedBy": "",
    "routerLocation": "",
    "ipAddressPool": "",
    "externalNetworkName": "",

```

```

"memoryRequired": 0,
"cpuRequired": 0,
"nicsRequired": 0,
"cpuReservationRequired": 0,
"memoryReservationRequired": 0,
"portGroup": {
  "portGroupId": 0,
  "vlanId": 0,
  "orgId": 0,
  "virtualMachineId": 0,
  "inherited": true,
  "recoveryplanid": 0
},
"organizationIds": [],
"resourcePools": [
  {
    "memoryReservation": 0,
    "cpuReservation": 0,
    "cpuLimit": 0,
    "memoryLimit": 0,
    "inherited": true,
    "resourcePoolId": 0,
    "virtualmachineid": 0,
    "recoveryplanid": 0
  }
],
"vmprefixName": ""
},
"appGroups": [
  {
    "startupSequence": 1,
    "appGroup": {
      "id": 8280,
      "name": "AG2",
      "description": "",
      "cdsId": 0,
      "cdsHostName": "",
      "serverHostName": "",
      "createdOn": "12/31/1969 19:00:00",

```

```

"editedOn": "01/27/2019 09:23:29",
"noOfVms": 0,
"noOfNonVms": 0,
"recoveryPlan": "RPAAppOnly",
"noOfVApps": 0,
"sequenceOrder": [
  {
    "memberOrderIndex": 1,
    "delay": 0,
    "vmMembers": [],
    "appMembers": [
      {
        "memberName": "msdb",
        "vm": false,
        "executionStatus": "",
        "cdsAppId": "6330",
        "cdsClusterId": "1415039054",
        "targetClusterId": "1415040293",
        "clustername": "RD_HYD_DEST",
        "validForAppGroup": false,
        "displayAppName": "msdb (win2012_sql_new) ",
        "displayAppType": "SQL Database",
        "applicationChildMembers": [],
        "apptype": "SqlServerWriter",
        "sourceHostName": "win2012_sql_new",
        "recoveryType": {
          "recoveryTypeId": 4,
          "name": "Recover SQL Server Database",
          "recoveryTypeOrder": "2"
        },
      },
      {
        "infoRequired": true,
        "id": 8282,
        "appName": "msdb",
        "originalAppId": 0,
        "sourceHostId": 0,
        "appClass": "SQLServer",
        "friendlyType": "SQLServer",
        "networkName": "",
        "targetHostId": 0,
      }
    ]
  }
]

```

```

"depth": 0,
"isGroup": false,
"isClustered": false,
"fromMount": false,
"failoverState": 0,
"isTbd": false,
"appClassMetaDataTOList": [
  {
    "appid": 0,
    "parameterId": 5242,
    "value": "WINVVM"
  },
  {
    "appid": 0,
    "parameterId": 5243,
    "value": "rddb1"
  },
  {
    "appid": 0,
    "parameterId": 5244,
    "value": true
  },
  {
    "appid": 0,
    "parameterId": 5245,
    "value": false
  },
  {
    "appid": 0,
    "parameterId": 5246
  },
  {
    "appid": 0,
    "parameterId": 5247
  }
],
"targetHostNameList": [
  "recovwery_shiva"
],

```



```

        "sequenceNumber": 1,
        "critical": false,
        "vmAlreadyPresentInRecPlan": false,
        "classname": "",
        "ip": "",
        "targetHostName": "recovery_shiva",
        "volumesToList": [],
        "scriptParameterToList": [],
        "inclusionList": "",
        "exclusionList": ""
    }
],
    "preExternalScript": [],
    "postExternalScript": [],
    "classname": "",
    "id": "id_219"
}
],
    "armServerId": 5209,
    "armServerName": "172.29.11.103",
    "recoveryPlanId": 8279,
    "executionStatus": "",
    "peerRecoveryPlanId": 0,
    "cpuRequired": 0,
    "memoryRequired": 0,
    "customerName": "dummy_cust",
    "peerAppGrpId": 5221,
    "preExternalScript": [],
    "postExternalScript": [],
    "organizationToList": [],
    "startupSequence": 1,
    "critical": false
},
    "classname": ""
}
]
}

```

Response

```
{
  "status": "success",
  "statusCode": "1031",
  "statusMessage": "Recovery plan execution initiated successfully.",
  "responseData": []
}
```

Parameters	Description
sessionid	Session id as a header parameter.
recoveryPlanMember	Recovery plan member details

fetchtargethostnameList

Use this API to fetch all target hostname present in appliance or appliances present in appliance group. This API will be invoked while configuring SQL apps in Recovery plan create/edit flow.

Method: GET

Request:

`https://{HOST}/armnew/api_fetchTargetHostNameList?targetclusterid=0&appliancegroupid=5245`

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "clustername": "RDTargetSky",
      "clusterid": 142966846547,
      "hostnameList": [
        "recovered-actsql02",
        "recovered-rhel71",
        "recovered_actsql06",
        "recovered_nike1-win-2012-sql2016",
        "recovered_rhel7_1"
      ]
    }
  ]
}
```

Managing Alerts – IRestRoleManager

This interface and its corresponding APIs allows you to execute REST requests to User role manager. This interface contains the following RESTful APIs:

- createRole
- deleteRoles
- getCDSRoles
- getRoles
- updateRole
- getRights

createRole

Use this API to create a new role in ARM database with the parameters provided in the request.

Method: POST

Request

https://{HOST}/armnew/api/role/add

Body Param

```
{
  "name": "RD Viewer",
  "description": "The role will have all access",
  "listRightIds": [
    1,
    2
  ],
  "listCDSIds": [
    3,
    4
  ]
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": ""
}
```

Parameters	Description
sessionId	Session id as header parameter.
roleToAdd	Role details that you want to add in the CSP database.

deleteRoles

Use this API to delete user roles in ARM database with the parameters provided in request.

Method: GET

Request

https://{HOST}/armnew/api/role/remove?ids=1101,1102,1103

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
ids	Comma separated alert id(s) that you want to fetch.
sessionid	Session id as header parameter.

getCDSRoles

Use this API to fetch CDS roles. If the method parameter is null, the method will return all CDS roles. If the method parameter is not null, the method will return only role(s) corresponding to the id(s) provided as an input.

Method: GET

Request

`https://{HOST}/armnew/api/role/fetchactifioapplianceroles?ids=1,2--` To fetch roles by id(s) -- (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "id": 1,
      "description": "The role will have all access",
      "name": "admin"
    },
    {
      "id": 2,
      "description": "The user having role will run RP",
      "name": "rdexecutor"
    }
  ]
}
```

Parameters	Description
sessionid	Session id as header parameter.
ids	Comma separated alert id(s) that you want to fetch.

getRoles

Use this API to fetch user roles from Resiliency Director database. If the method parameter is null, the method will return all roles from database. If the method parameter is not null, method will return only the role(s) corresponding to the id(s) provided as an input.

Method: GET

Request

`https://{HOST}/armnew/api/role/fetch?ids=1101,1102` -To fetch roles by id(s)

`https://{HOST}/armnew/api/role/fetch?ids=` -To fetch all roles

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "roleId": 57346,
      "name": "RD User",
      "description": "The role will have all access",
      "createDate": "06/16/2015 03:51:19",
      "modifyDate": "06/16/2015 04:41:51",
      "listRights": [
        {
          "id": 1,
          "name": "Customer View",
          "description": "View cust details."
        }
      ],
      "listCDSRoles": [
        {
          "id": 1,
          "description": "The role will have all access",
          "name": "admin"
        }
      ]
    },
    {
      "roleId": 59393,
      "name": "RD Viewer",
      "description": "The role will have all access",
      "createDate": "06/16/2015 04:45:05",
      "modifyDate": "06/16/2015 04:45:05",
      "listRights": [
        {
          "id": 1,
          "name": "Customer View",
          "description": "View cust details."
        }
      ],
      "listCDSRoles": [
        {
          "id": 1,
          "description": "The role will have all access",
          "name": "admin"
        }
      ]
    }
  ]
}
```

```
]
}
```

Parameters	Description
sessionid	Session id as header parameter.
ids	Comma separated alert id(s) that you want to fetch.

updateRole

Use this API to update a role in ARM database.

Method: GET

Request

`https://{HOST}:{PORT}/armnew/api/role/update`

Body Param

```
{
  "roleId": 59393,
  "name": "RD User",
  "description": "The role will have all access",
  "createDate": "06/16/2015 03:51:19",
  "listRightIds": [
    1,
    2
  ],
  "listCDSIds": [
    1,
    2
  ]
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionid	Session id as header parameter.
roleToUpdate	Roles that you want to update in the database.

getRights

Use this API to fetch rights from Resiliency Director database. If the method parameter is null, the method will return all the rights from database. If the method parameter is not null, the method will return only the rights(s) whose id(s) have been provided as an input.

Method: GET

Request

`https://{HOST}/armnew/api/fetchrights?ids=4101,4109` - (HTTPS)

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "id": 4101,
      "name": "Customer View",
      "description": "View customer details."
    },
    {
      "id": 4109,
      "name": "Appliance Manage",
      "description": "Create or delete appliance."
    }
  ]
}
```

Parameters	Description
sessionId	Session id as header parameter.
ids	Comma separated virtual management server id(s) for which you want to fetch the data.

Managing Virtual Management Server Resources- IRestVMSResources

This interface and its corresponding RESTful APIs allows you to add, list, edit, and delete virtual management server resources used by the Resiliency Director. This interface contains the following RESTful APIs:

- [createPortGroup](#)
- [createResourcePool](#)
- [deletePortGroup](#)
- [deleteResourcePool](#)
- [discoverPortGroupsForMapping](#)
- [discoverResourcepoolsForMapping](#)
- [fetchHostOrClusterFromServer](#)
- [fetchPortGroupOrgs](#)
- [fetchResourcePoolOrgs](#)
- [getPortGroupsForMapping](#)
- [getResourcePoolsforMapping](#)
- [refreshResourcePool](#)
- [updatePortGroup](#)
- [updateResourcePool](#)

createPortGroup

Use this API to create a new port group in the Actifio Resiliency Director database with the parameters provided in request.

Method: POST

Request

<https://{HOST}/armnew/api/vmsresources/addportgroup>

Body Param

```
[
  {
    "listOrganizations": [
      {
        "id": 13470
      }
    ],
    "portgroup": {
      "portGroupId": 0,
      "portGroupName": "Test.delete",
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vLanId": 0,
      "hostName": "172.29.11.31",
      "portgroupDisplayName": "Test.delete_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 0,
      "inherited": true,
      "recoveryplanid": 0,
      "organizationsRP": [
        "Org1",
        "Org2"
      ],
      "clusterName": "Rd_Cluster"
    },
    "server": {
```



```

        "id": 5207,
        "hostOrIp": "172.29.11.105",
        "username": "root",
        "password": "",
        "port": 443,
        "discovered": true,
        "noOfRecoverPlan": 1,
        "listOrganizations": [],
        "selected": false
    }
}
]

```

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}

```

Parameters	Description
sessionid	Session id as a header parameter.
portGroupToAdd	Actifio appliance port group to be added in the Resiliency Director database.

createResourcePool

Use this API to create a new resource pool in the Actifio Resiliency Director database with the parameters provided in request.

Method: POST

Request

<https://{HOST}/armnew/api/vmsresources/addresourcepool>

Body Param

```

[
  {
    "listOrganizations": [
      {
        "id": 13472
      }
    ],
    "resourcepool": {
      "resourcePoolName": "allVMs",
      "resourcePoolMemory": "0",
      "resourcePoolCPU": "0",
      "resourcePoolParentName": "Resources",
      "resourcePoolOwnerName": "Rd_Cluster",
      "resourcePoolUuid": "resgroup-11673",
      "resourcePoolIsInvalid": false,
      "resourcePoolMemoryLimit": "0",
      "resourcePoolCPULimit": "0",
      "resourcePoolMemoryExpandableReservation": false,
      "resourcePoolCPUExpandableReservation": true,
      "organizationsRP": [
        "Org1",

```

```

        "Org2"
    ]
},
"server": {
    "id": 5207,
    "hostOrIp": "172.29.11.105",
    "username": "root",
    "password": "",
    "port": 443,
    "discovered": true,
    "noOfRecoverPlan": 1,
    "listOrganizations": [],
    "selected": false
}
}
]

```

Response

```

{
    "status": "success",
    "statusCode": "1000",
    "statusMessage": "Operation performed successfully",
    "responseData": ""
}

```

Parameters	Description
sessionId	Session id as a header parameter.
resourcePoolToAdd	Resource pool to be added in the Resiliency Director database.

deletePortGroup

Use this API to delete port groups in the Resiliency Director database.

Method: GET

Request

<https://{HOST}/armnew/api/vmsresources/removeportgroup?ids=1111,1112>

Response

```

{
    "status": "success",
    "statusCode": "1000",
    "statusMessage": "Operation performed successfully.",
    "responseData": ""
}

```

Parameters	Description
sessionId	Session id as a header parameter.
ids	ids of port groups to delete.

deleteResourcePool

Use this API to delete resource pool in Actifio Resiliency Director database.

Method: GET

Request

https://{HOST}/armnew/api/vmsresources/removeresourcepool?ids=1111,1112

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
sessionid	Session id as a header parameter.
ids	ids of resource pools to delete.

discoverPortGroupsForMapping

Use this API to fetch port groups of resource pools of Virtual Management Server.

Method: GET

Request

https://{HOST}/armnew/api/vmsresources/
discoverportgroups?serverid=5207&clustername=Rd_Cluster&_=1549261858577

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "portGroupId": 0,
      "portGroupName": "QA-network4",
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vLanId": 0,
      "hostName": "172.29.11.31",
      "portgroupDisplayName": "QA-network4_vSwitch0",
      "orgId": 0,
      "virtualMachineId": 0,
      "inherited": true,
      "recoveryplanid": 0
    },
    {
      "portGroupId": 0,
      "portGroupName": "Sample123",
      "portGroupType": "Virtual Portgroup",
      "vSwitchName": "vSwitch0",
      "vLanId": 0,

```

```

    "hostName": "172.29.11.33",
    "portgroupDisplayName": "Sample123_vSwitch0",
    "orgId": 0,
    "virtualMachineId": 0,
    "inherited": true,
    "recoveryplanid": 0
  },
  {
    "portGroupId": 0,
    "portGroupName": "QA-network2",
    "portGroupType": "Virtual Portgroup",
    "vSwitchName": "vSwitch0",
    "vlanId": 0,
    "hostName": "172.29.11.31",
    "portgroupDisplayName": "QA-network2_vSwitch0",
    "orgId": 0,
    "virtualMachineId": 0,
    "inherited": true,
    "recoveryplanid": 0
  },
  {
    "portGroupId": 0,
    "portGroupName": "WindowsNW",
    "portGroupType": "Virtual Portgroup",
    "vSwitchName": "vSwitch0",
    "vlanId": 0,
    "hostName": "172.29.11.31",
    "portgroupDisplayName": "WindowsNW_vSwitch0",
    "orgId": 0,
    "virtualMachineId": 0,
    "inherited": true,
    "recoveryplanid": 0
  },
  {
    "portGroupId": 0,
    "portGroupName": "win2008-nw",
    "portGroupType": "Virtual Portgroup",
    "vSwitchName": "vSwitch0",
    "vlanId": 0,
    "hostName": "172.29.11.31",
    "portgroupDisplayName": "win2008-nw_vSwitch0",
    "orgId": 0,
    "virtualMachineId": 0,
    "inherited": true,
    "recoveryplanid": 0
  },
  {
    "portGroupId": 0,
    "portGroupName": "LinuxNW",
    "portGroupType": "Virtual Portgroup",
    "vSwitchName": "vSwitch0",
    "vlanId": 0,
    "hostName": "172.29.11.31",
    "portgroupDisplayName": "LinuxNW_vSwitch0",
    "orgId": 0,
    "virtualMachineId": 0,
    "inherited": true,
    "recoveryplanid": 0
  }
]

```

}

Parameters	Description
sessionid	Session id as a header parameter.
serverid	Id of the Hypervisor from where port groups are to be listed.
resourcepoolid	Id of resource pools that are mapped to organization port groups are to be listed.
clustername	Name of the cluster.

discoverResourcepoolsForMapping

Use this API to fetch resource pool from the Virtual Management Server with given server id and resource pool name.

Method: GET

Request

`https://{HOST}/armnew/api/vmsresources/discoverresourcepool?serverid=5126&resourcepool=rp1` -To fetch organizations by id

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "resourcePoolId": null,
      "resourcePoolName": "IMP** Do not delete - vCenter",
      "resourcePoolMemory": "94",
      "resourcePoolCPU": "0",
      "resourcePoolParentName": "Resources",
      "resourcePoolOwnerName": "RD",
      "resourcePoolUuid": "resgroup-1387",
      "resourcePoolIsInvalid": false
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
serverid	Server id as query parameter.

fetchHostOrClusterFromServer

Use this API to fetch host or cluster details from server in the Resiliency Director database.

Method: GET

Request

https://{HOST}/armnew/api/vmsresources/fetchhostorclusters?serverid=4555 -(HTTPS)

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "serverId": 4555,
      "clusterNames": "Cluster1"
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
serverid	Server id as header parameter.

fetchPortGroupOrgs

Use this API to fetch organization(s) of port group in the Resiliency Director database provided in request.

Method: GET

Request

https://{Host}/armnew/api/vmsresources/fetchportgrouporgs?portgroupid=182296&serverid=174105

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 183367,
      "name": "sadsa",
      "description": "sdads",
      "createdate": "04/17/2020 08:01:37",
      "modifydate": "04/17/2020 08:01:37",
      "email": "",
      "scheduleEnable": true,
      "parentOrgId": 0,
      "listCDSOrgs": [],
      "childOrgs": []
    }
  ]
}
```

}

Parameters	Description
sessionid	Session id as a header parameter.
portgroupid	portgroup id that you want to fetch.
serverid	server id of the portgroup.

fetchResourcePoolOrgs

Use this API to fetch organization(s) of resource pool in the Actifio Resiliency Director database provided in request.

Method: GET

Request

<https://{HOST}/armnew/api/vmsresources/fetchresourcepoolorgs?resourcepoolid=182293>

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 183367,
      "name": "sadsa",
      "description": "sdads",
      "createdate": "04/17/2020 08:01:37",
      "modifydate": "04/17/2020 08:01:37",
      "email": "",
      "scheduleEnable": true,
      "parentOrgId": 0,
      "listCDSOrgs": [],
      "childOrgs": []
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
resourcepoolid	Resource pool id that you want to fetch.

getPortGroupsForMapping

Use this API to fetch mapped port groups from Virtual Management Server.

Method: GET

Request

<http://{HOST}/armnew/api/vmsresources/fetchportgroups>

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "server": {
        "id": 5138,
        "hostOrIp": "172.29.10.179",
        "username": "root",
        "password": "",
        "port": 0,
        "release": "1.1",
        "discovered": true,
        "deploymentPlatform": "platform",
        "noOfRecoverPlan": 0,
        "listOrganizations": null
      },
      "resourcePoolNames": [
        "testrp1",
        "testrp2",
        "testrp3",
        "testrp4"
      ],
      "portgroup": {
        "portGroupId": 5189,
        "portGroupName": "ND-Default-Portgroup",
        "portGroupKey": "null",
        "portGroupType": "Virtual Portgroup",
        "vSwitchName": "vSwitch0",
        "vSwitchUUID": "null",
        "vLanId": 0,
        "hostName": "172.29.10.29",
        "portgroupDisplayName": "ND-Default-Portgroup",
        "orgId": 0,
        "virtualMachineId": 0,
        "macAddress": null,
        "nicType": null,
        "clusterName": "RD"
      },
      "listOrganizations": [
        {
          "organizationId": 1101,
          "name": "ALL",
          "description": "ALL organization, all objects belong to ALL organization.",
          "listCDSOrgs": null,
          "createDate": "06/16/2015 03:51:19",
          "modifyDate": "06/16/2015 04:41:51",
          "childOrgs": null,
          "parentOrgId": 3,
          "scheduleEnable": true,
          "email": "user@testemail.com"
        }
      ]
    }
  ]
}

```


}

Parameters	Description
sessionid	Session id as a header parameter.

getResourcePoolsforMapping

Use this API to fetch resource pools present on given VMS.

Method - GET

Request

https://{HOST}/armnew/api/vmsresources/fetchresourcepool - (HTTPS)

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": {
    "server": [
      {
        "deploymentPlatform": "platform",
        "discovered": true,
        "hostOrIp": "172.29.11.105",
        "id": 5126,
        "listOrganizations": [],
        "noOfRecoverPlan": 3,
        "password": "",
        "port": 0,
        "release": "1.1",
        "username": "root"
      }
    ],
    "resourcepool": [
      {
        "resourcePoolId": "1111",
        "resourcePoolName": "RP1",
        "resourcePoolMemory": "40000",
        "resourcePoolCPU": "5800",
        "resourcePoolParentName": "Resources",
        "resourcePoolOwnerName": "Rd_Cluster",
        "resourcePoolUuid": "resgroup-2844",
        "resourcePoolIsInvalid": "false"
      }
    ],
    "listOrganizations": [
      {
        "id": 6250,
        "name": "netaji",
        "description": null,
        "createdate": null,
        "modifydate": null,
        "email": null,
        "scheduleEnable": false,
        "parentOrgId": 0,
        "listCDSOrgs": null,
        "childOrgs": null
      }
    ]
  }
}
```

```

    }
  ]
}
}

```

Parameters	Description
sessionId	Session id as a header parameter.

refreshResourcePool

Use this API to refresh the resource pool state from invalid to valid. A resource pool is moved into an invalid state when the resource pool attribute on Virtual Management Server are changed. Due to this change, the mapped resource pool used in a recovery plan might fail during execution. Hence the user is notified by moving the resource pool and the corresponding recovery plan into invalid state.

This API provides a way for the user to acknowledge this change. By calling this method on a resource pool present in invalid state, the user simply acknowledges that the resource pool attributes are changed. If the user calls this API on a resource pool which is not present in an invalid state, no changes would happen to the state of the resource pool and the API would simply return.

Method: POST

Request

<https://{HOST}:{PORT}/armnew/api/vmsresources/refreshresourcepool> - (HTTPS)

Body Param

```

{
  "vmsResourceid": "1111",
  "organizationIds": [
    1113,
    1114
  ]
}

```

Response

```

{
  "vmsResourceid": "1111",
  "organizationIds": [
    1113,
    1114
  ]
}

```

Parameters	Description
sessionId	Session id as a header parameter.
resourcePoolToRefresh	Resource pool details that you want to add in the Resiliency Director database.

updatePortGroup

Use this API to update a port group in the Actifio Resiliency Director database with the parameters provided in request.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/vmsresources/updateportgroup - (HTTPS)

Body Param

```
{
  "id": null,
  "vmsResourceid": "1111",
  "organizationIds": [
    1113,
    1114
  ]
}
```

Response

```
{ "status": "success", "statusCode": "1000", "statusMessage": "Operation performed successfully.", "responseData": "" }
```

Parameters	Description
sessionid	Session id as a header parameter.
orgToUpdate	Organizations you want to update in the Resiliency Director database.

updateResourcePool

Use this API to update a resource pool in the Actifio Resiliency Director database.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/vmsresources/updateresourcepool - (HTTPS)

Body Param

```
{
  "vmsResourceid": "1111",
  "organizationIds": [
    1113,
    1114
  ]
}
```

Response

```
{ "status": "success", "statusCode": "1000", "statusMessage": "Operation performed successfully.", "responseData": "" }
```

Parameters	Description
sessionid	Session id as a header parameter.

Parameters	Description
resourcePoolToUpdate	Resource pool that you want to update in the Resiliency Director database.

Managing Alerts – IRestAlerts

This interface and corresponding APIs allow you to execute REST requests for alerts. This interface contains the following RESTful APIs:

[getAlerts](#)
[getLatestAlerts](#)

getAlerts

Use this API to fetch alerts from the Actifio Resiliency Director database. If the method parameter is null, the method returns all the alerts from database. If the method parameter is not null, the method returns only the alert(s) whose id(s) is provided as an input.

Method: GET

Request

`https://{HOST}:{PORT}/armnew/api/alerts/fetch` - To fetch all the alerts from database.

`https://{HOST}:{PORT}/armnew/api/alerts/fetch?ids=1,2,3` - To fetch alerts by id(s).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "id": 833802,
      "description": "Failed snapshot Job_0833414c for application _cl on host _cl",
      "createdOn": "11/02/2015 03:24:35",
      "entity": "Actifio Appliance",
      "action": "Add actifio appliance",
      "status": "error"
    }
  ]
}
```

Parameters	Description
ids	Comma separated alert id(s) that you want to fetch.

getLatestAlerts

Use this API to fetch the alerts by offset and limit from the Actifio Resiliency Director database and past 24 event alerts from the Actifio Appliance. If the method parameter is null, the method returns all alerts from database. If the method parameter is not null, the method returns only the limited alert(s) based on offset and limit.

Method: GET

Request

`https://{HOST}/armnew/api/alerts/fetchbyoffset?offset=1&limit=1` - To fetch alerts by offset and limit

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "id": 833802,
      "description": "Failed snapshot Job_0833414c for application _cl on host _cl",
      "createdOn": "11/02/2015 03:24:35",
      "entity": "Actifio Appliance",
      "action": "Add actifio appliance",
      "status": "error"
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
offset	Offset from which alerts needs to be read as query parameter.
limit	Number of record needs to be fetched as query parameter.

Managing Dashboard – IRestDashboard

This interface and its corresponding APIs allow you to fetch the recovery plan information, recovery plan execution statuses for the selected time period, application group included and excluded in the recovery plan, and no. of applications included and excluded in the recovery plan, events, health status, and system details.

This interface contains the following RESTful APIs:

- [fetchrecoveryplancount](#)
- [getRecoveryPlanStatesForGivendateRange](#)
- [fetchappcount](#)
- [fetchapplicationgroupcount](#)
- [fetchhealthinfo](#)
- [fetchsysteminfo](#)
- [fetchapplicationtypecount](#)

fetchrecoveryplancount

Use this API to fetch count of the recovery plan(s) in different states present in the Resiliency Director Server/ Cloud Recovery based on the selected statuses – All, Test, Actual.

When the status “**All**” is selected:

Method: GET

Request:

`https://{HOST}/armnew/api/recoveryplan/fetchcount?status=ALL`

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "infocomplete": 8,
    "cancelled": 0,
    "resetting": 0,
    "total": 9,
    "resetfailed": 0,
    "inprogress": 0,
    "partialsuccess": 0,
    "failed": 0,
    "successful": 0,
    "inforequired": 1
  }
}
```

When the status “**Test or Actual**” is selected:

Method: GET

Request:

`https://{HOST}/armnew/api/recoveryplan/fetchcount?status=TEST`

Response:

```
{
  "status": "success",
```

```

    "statusCode": "1000",
    "statusMessage": "Operation performed successfully.",
    "responseData": {
      "total": 1,
      "inProgress": 0,
      "failed": 0,
      "successful": 1
    }
  }
}

```

getRecoveryPlanStatesForGivendateRange

Use this API to fetch the details of the executed recovery plans over a period of time along with the duration and status of each.

Method: GET

Request:

<https://{HOST}/armnew/api/recoveryplan/getRecoveryPlanStatesForGivendateRange?startdate=1596884150091&enddate=1597402550091>

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "status": "Success",
      "duration": "193 sec",
      "recoverystartdate": 1597334204194,
      "recoveryenddate": 1597334397269,
      "recoveryplanid": "157756",
      "recoveryplanname": "RP_AAG"
    },
    {
      "status": "Success",
      "duration": "122 sec",
      "recoverystartdate": 1597401663060,
      "recoveryenddate": 1597401785473,
      "recoveryplanid": "144425",
      "recoveryplanname": "RP_SingleDB"
    },
    {
      "status": "Success",
      "duration": "125 sec",
      "recoverystartdate": 1597136291546,
      "recoveryenddate": 1597136416602,
      "recoveryplanid": "144425",
      "recoveryplanname": "RP_SingleDB"
    },
    {
      "status": "Failed",
      "duration": "7 sec",
      "recoverystartdate": 1597136511860,
      "recoveryenddate": 1597136518961,
      "recoveryplanid": "144425",
      "recoveryplanname": "RP_SingleDB"
    },
    {
      "status": "Success",

```

```

    "duration": "121 sec",
    "recoverystartdate": 1597136702521,
    "recoveryenddate": 1597136824273,
    "recoveryplanid": "144425",
    "recoveryplanname": "RP_SingleDB"
  },
  {
    "status": "Success",
    "duration": "127 sec",
    "recoverystartdate": 1597140725493,
    "recoveryenddate": 1597140853043,
    "recoveryplanid": "144425",
    "recoveryplanname": "RP_SingleDB"
  },
  {
    "status": "Failed",
    "duration": "0 sec",
    "recoverystartdate": 1597166304156,
    "recoveryenddate": 1597166304929,
    "recoveryplanid": "151594",
    "recoveryplanname": "RP_FS_Both"
  },
  {
    "status": "Failed",
    "duration": "8 sec",
    "recoverystartdate": 1597211200652,
    "recoveryenddate": 1597211208750,
    "recoveryplanid": "96320",
    "recoveryplanname": "RPAwsSQLInstance"
  },
  {
    "status": "Success",
    "duration": "228 sec",
    "recoverystartdate": 1597211491345,
    "recoveryenddate": 1597211720225,
    "recoveryplanid": "96320",
    "recoveryplanname": "RPAwsSQLInstance"
  },
  {
    "status": "Failed",
    "duration": "15 sec",
    "recoverystartdate": 1597212089057,
    "recoveryenddate": 1597212104475,
    "recoveryplanid": "88082",
    "recoveryplanname": "RP_ConGrp_AWS"
  },
  {
    "status": "Failed",
    "duration": "85 sec",
    "recoverystartdate": 1597228911418,
    "recoveryenddate": 1597228996456,
    "recoveryplanid": "151637",
    "recoveryplanname": "RP_DB_Both_Pol"
  },
  {
    "status": "Success",
    "duration": "123 sec",
    "recoverystartdate": 1597243986834,
    "recoveryenddate": 1597244110192,
    "recoveryplanid": "144425",

```



```

        "recoveryplanname": "RP_SingleDB"
    }
]
}

```

fetchappcount

Use this API to fetch count of the app(s) included and excluded in the recovery plan managed in the Resiliency Director.

Method: GET

Request:

https://{HOST}/armnew/api/applicationgroup/fetchappcount?

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "includeInRecovery": 9,
    "excludeFromRecovery": 0,
    "total": 9
  }
}

```

fetchapplicationgroupcount

Use this API to fetch count of application group(s) included and excluded in the recovery plan managed in the Resiliency Director.

Method: GET

Request:

https://{HOST}/armnew/api/applicationgroup/fetchcount

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "includeInRecovery": 9,
    "excludeFromRecovery": 0,
    "total": 9
  }
}

```

getreportcountbytype

Use this API to fetch the count of events.

Method: GET

Request:

https://{HOST}/armnew/api/report/getreportcountbytype

Response:

```

{
  "status": "success",
  "statusCode": "1000",

```

```

    "statusMessage": "Operation performed successfully.",
    "responseData": {
      "1": 12,
      "2": 2,
      "3": 6
    }
  }
}

```

Note:

Event type 1 = INFO

Event type 2 = ERROR

Event type 3 = WARNING

fetchhealthinfo

Use this API to fetch the health status of different appliances present in Resiliency Director.

Method: GET

Request:

<https://{HOST}/armnew/api/appliances/health/info>

For RD Cloud Recovery

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "skyAppliance": {
      "online": 3,
      "offline": 0
    },
    "agmAppliance": {
      "online": 1,
      "offline": 0
    },
    "cloudStatus": {
      "online": 7,
      "offline": 0
    }
  }
}

```

For RD Server

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "vmsAppliance": {
      "online": 1,
      "offline": 0
    },
    "skyAppliance": {
      "online": 3,

```

```

        "offline":0
    }
}
}

```

fetchsysteminfo

Use this API to fetch the system details of the Resiliency Director.

Method: GET

Request:

https://{HOST}/armnew/api/system/info

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "hostname": "cloud",
    "os": "Actifio Linux 6",
    "totalmemory": "8061188 KB",
    "netmask": "255.255.252.0",
    "freememory": "4866740 KB",
    "dnsserver": "192.168.192.10,192.168.225.2",
    "usedmemory": "3194448 KB",
    "ipaddr": "172.27.46.75",
    "gateway": "172.27.44.1",
    "freecpu": "99.6%"
  }
}

```

fetchapplicationtypecount

Use this API to fetch the count of the type of applications associated with each application group.

Method: GET

Request:

https://{HOST}/armnew/api/applicationgroup/fetchApplicationCount

Response:

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "appgrpname": "aglinux",
      "appcount": {
        "VM": 1
      }
    },
    {
      "appgrpname": "AG_SingleDB",
      "appcount": {
        "SqlServerWriter": 1
      }
    }
  ]
}

```

```

        "appgrpname": "AG_AAG",
        "appcount": {}
    },
    {
        "appgrpname": "AGConGrp_AWS",
        "appcount": {
            "ConsistGrp": 1
        }
    },
    {
        "appgrpname": "AG_FS_Both_Pol",
        "appcount": {
            "FileSystem": 1
        }
    },
    {
        "appgrpname": "AG_DB_Both_Pol",
        "appcount": {
            "SqlServerWriter": 1
        }
    },
    {
        "appgrpname": "AGSQLInstance",
        "appcount": {
            "SqlInstance": 1
        }
    }
]
}

```

Viewing events-IRestEvents

This interface and corresponding RESTful API allows you to view events that occurred on the RD Server/Cloud Recovery. This interface contains the following the RESTful API:

[getdetailreport](#)

getdetailreport

Use this API to get event details that occurred in the Resiliency Director Server/Cloud Recovery.

Method: GET

Request:

<https://{HOST}/armnew/api/report/getdetailreport?limit=25&offset=0>

Response:

```

{
    "status": "success",
    "statusCode": "1000",
    "statusMessage": "Operation performed successfully.",
    "responseData": {
        "count": "935",
        "auditLoggingReportList": [
            {
                "auditid": 182339,
                "username": "admin",
                "action": "Login",
                "status": "User :admin authenticated successfully.",
                "starttime": "08/24/2020 03:15:06",
                "endtime": "08/24/2020 03:15:06",
                "serverIdentity": "Solo",
                "notificationtype": 1
            }
        ]
    }
}

```

```

    },
    {
        "auditid": 182338,
        "username": "admin",
        "action": "Login",
        "status": "User :admin authenticated successfully.",
        "starttime": "08/24/2020 00:50:10",
        "endtime": "08/24/2020 00:50:10",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182337,
        "username": "admin",
        "action": "Login",
        "status": "User :admin authenticated successfully.",
        "starttime": "08/23/2020 06:53:09",
        "endtime": "08/23/2020 06:53:09",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182336,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:59:31",
        "endtime": "08/21/2020 07:59:31",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182335,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:59:08",
        "endtime": "08/21/2020 07:59:09",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182334,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:58:41",
        "endtime": "08/21/2020 07:58:41",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182333,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:58:17",
        "endtime": "08/21/2020 07:58:18",
        "serverIdentity": "Solo",
        "notificationtype": 1
    }

```

```

    },
    {
      "auditid": 182332,
      "username": "admin",
      "action": "Cloud Server Update",
      "status": "Cloud definition updated successfully. Cloud id: 182320",
      "starttime": "08/21/2020 07:57:08",
      "endtime": "08/21/2020 07:57:08",
      "serverIdentity": "Solo",
      "notificationtype": 1
    },
    {
      "auditid": 182331,
      "username": "admin",
      "action": "Cloud Server Update",
      "status": "Cloud definition updated successfully. Cloud id: 182320",
      "starttime": "08/21/2020 07:56:40",
      "endtime": "08/21/2020 07:56:40",
      "serverIdentity": "Solo",
      "notificationtype": 1
    },
    {
      "auditid": 182330,
      "username": "admin",
      "action": "Login",
      "status": "User :admin authenticated successfully.",
      "starttime": "08/21/2020 07:54:15",
      "endtime": "08/21/2020 07:54:15",
      "serverIdentity": "Solo",
      "notificationtype": 1
    },
    {
      "auditid": 182329,
      "username": "admin",
      "action": "Login",
      "status": "User :admin authenticated successfully.",
      "starttime": "08/21/2020 07:51:09",
      "endtime": "08/21/2020 07:51:09",
      "serverIdentity": "Solo",
      "notificationtype": 1
    },
    {
      "auditid": 182328,
      "username": "admin",
      "action": "Cloud Server Update",
      "status": "Cloud definition updated successfully. Cloud id: 182320",
      "starttime": "08/21/2020 07:47:27",
      "endtime": "08/21/2020 07:47:27",
      "serverIdentity": "Solo",
      "notificationtype": 1
    },
    {
      "auditid": 182327,
      "username": "admin",
      "action": "Cloud Server Update",
      "status": "Cloud definition updated successfully. Cloud id: 182320",
      "starttime": "08/21/2020 07:45:48",
      "endtime": "08/21/2020 07:45:48",
      "serverIdentity": "Solo",
      "notificationtype": 1
    }
  ]
}

```

```

    },
    {
        "auditid": 182326,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:44:57",
        "endtime": "08/21/2020 07:44:57",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182325,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:43:18",
        "endtime": "08/21/2020 07:43:18",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182324,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:42:39",
        "endtime": "08/21/2020 07:42:39",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182323,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:41:18",
        "endtime": "08/21/2020 07:41:18",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182322,
        "username": "admin",
        "action": "Cloud Server Update",
        "status": "Cloud definition updated successfully. Cloud id: 182320",
        "starttime": "08/21/2020 07:41:00",
        "endtime": "08/21/2020 07:41:01",
        "serverIdentity": "Solo",
        "notificationtype": 1
    },
    {
        "auditid": 182321,
        "username": "admin",
        "action": "AGM Server Add",
        "status": "Cloud definition added successfully. with name:sarathcloud and id:0",
        "starttime": "08/21/2020 07:29:55",
        "endtime": "08/21/2020 07:29:55",
        "serverIdentity": "Solo",
        "notificationtype": 1
    }

```

```

    },
    {
      "auditid": 182319,
      "username": "admin",
      "action": "Cloud Server Update",
      "status": "Cloud definition updated successfully. Cloud id: 182317",
      "starttime": "08/21/2020 07:20:53",
      "endtime": "08/21/2020 07:20:53",
      "serverIdentity": "Solo",
      "notificationtype": 1
    },
    {
      "auditid": 182314,
      "username": "admin",
      "action": "Login",
      "status": "User :admin authenticated successfully.",
      "starttime": "08/21/2020 03:43:16",
      "endtime": "08/21/2020 03:43:16",
      "serverIdentity": "Solo",
      "notificationtype": 1
    },
    {
      "auditid": 182313,
      "username": "admin",
      "action": "Login",
      "status": "User :admin authenticated successfully.",
      "starttime": "08/21/2020 02:22:51",
      "endtime": "08/21/2020 02:22:51",
      "serverIdentity": "Solo",
      "notificationtype": 1
    }
  ]
}

```

Parameter	Description
Sessionid	Session id as a header parameter.
offset	Offset from which events need to be read as query parameters.
limit	Number of records needs to be fetched as a query parameter.

Viewing Logs – IRestLogs

This interface and corresponding APIs allows you to execute REST requests to the Actifio Resiliency Director for Actifio Appliance logs. This interface contains the following RESTful APIs:

[fetchCDSLogs](#)
[getBRDLogFile](#)

fetchCDSLogs

Use this API to fetch the Actifio Appliance logs. The method returns the Actifio Appliance logs whose id(s) is provided as an input.

Method: GET

Request

https://{HOST}/armnew/api/logs/cds/fetch?id=1 - To fetch Actifio appliance logs by id (HTTPS).

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully",
  "responseData": [
    {
      "id": "625504",
      "username": "admin",
      "component": "API",
      "status": "0",
      "issuedate": "2014-03-04 02:00:08.320",
      "proxy": "",
      "command": "apilogin admin calsoft",
      "privileged": "true",
      "ipaddress": ""
    }
  ]
}
```

Parameters	Description
ids	Actifio Appliance id(s) for which you want to fetch the logs.

getBRDLogFile

Use this API to fetch the Resiliency Director log file.

Method: GET

Request

https://{HOST}/armnew/api/logs/get

Response

Returns the compressed file for all the Resiliency Director logs.

Parameters	Description
sessionid	Session id as a query parameter.

Fetching VM Count – IRestLicense

This interface and corresponding API allows you to fetch VM count for corresponding organization. This interface contains the following RESTful API:

[getOrganizationVMCount](#)

getOrganizationVMCount

Use this API to fetch VM count for each customer.

Method: GET

Request

`https://{HOST}/armnew/api/license/fetch?orgids=`
`https://{HOST}/armnew/api/license/fetch?orgids=4101`

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "",
  "responseData": [
    {
      "orgName": "build_66.1",
      "orgVMCount": 4,
      "listRecoveryPlanVMCountTo": [
        {
          "recoveryPlanName": "rec1",
          "recoveryPlanVMCount": 2,
          "listAppGroupVMCountTo": [
            {
              "appGroupName": "app4",
              "appGroupVMCount": 2
            }
          ]
        }
      ],
    },
    {
      "recoveryPlanName": "rec2",
      "recoveryPlanVMCount": 2,
      "listAppGroupVMCountTo": [
        {
          "appGroupName": "app2",
          "appGroupVMCount": 2
        }
      ]
    }
  ]
},
{
  "orgName": "Aniket",
  "orgVMCount": 1,
  "listRecoveryPlanVMCountTo": [
    {
      "recoveryPlanName": "rec3",
      "recoveryPlanVMCount": 1,
      "listAppGroupVMCountTo": [
        {
          "appGroupName": "suvAp1",
          "appGroupVMCount": 1
        }
      ]
    }
  ]
}
```

```
}  
  ]  
  }  
  ]  
  }  
  ]
```

Fetching Client Properties – IRestClientProperties

This interface and corresponding API allows you to manage information of client properties. This interface contains the following RESTful API:

[getIntervalProperties](#)

getIntervalProperties

Use this API to fetch client properties in JSON format.

Method: GET

Request:

`http://{HOST}:{PORT}/armnew/api/clientproperties/fetch?propertiesNames=max-delay`

`http://{HOST}:{PORT}/armnew/api/clientproperties/fetch?propertiesNames=max-sequence`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "max-delay": "3600",
    "max-sequence": "9999"
  }
}
```

Parameters	Description
sessionid	Session id as a header parameter.

Managing Files – IRestFileManager

This interface and corresponding APIs allow you to manage scripts. This interface contains the following RESTful APIs:

- [deleteFileOnServer](#)
- [deleteScriptsByIds](#)
- [fetchFileListOnServer](#)
- [uploadFile](#)
- [getSupportedExternalScript](#)
- [updateScriptHostData](#)

deleteFileOnServer

Use this API to delete the script file from a file storage path as well as from the Resiliency Director database.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/script/deletebyname?filename=deployLocalDB.sh

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully"
}
```

Parameters	Description
sessionid	Session id as a header parameter.
filename	Name of the file that has to be deleted from storage path.

deleteScriptsByIds

Use this API to delete scripts with the given ids. It will remove respective entries from database and delete them physically. It also returns successfully deleted script names.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/script/deletebyids?ids=6169,1200

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully"
}
```

Parameters	Description
sessionid	Session id as a header parameter.

Parameters	Description
ids	Script id(s) using which you want to delete scripts.

fetchFileListOnServer

Use this API to get the details of scripts which are currently uploaded on the server.

Method: GET

Request

`https://{HOST}:{PORT}/armnew/api/script/fetch`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": ""
}
```

Parameters	Description
sessionId	Session id as a header parameter.

uploadFile

Use this API to upload script on server.

Method: POST

Request:

`https://{HOST}:{PORT}/armnew/api/script/upload?command=python&sizeinkb=1025&hostids=2145,6688&isFileText=true&organizationIds=2156,2569`

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": ""
}
```

Parameters	Description
configParams	This should be multipart data for file needs to upload.
sessionId	This is header parameter and should have a valid sessionId.
scriptType	This parameter is used as a query command by which script should execute.
scriptSize	This query parameter should fetch the size of file in bytes.

Parameters	Description
uploadComponentName	This query parameter uploads the component name.
hostids	This query parameter is used to fetch remote host ids.
isFileText	This query parameter will fetch the file, if file is text true.
organizationIds	This parameter will fetch the organization ids.

getSupportedExternalScript

Use this API to get a supported external scripts.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/script/supportedscript

Response:

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "",
  "responseData": {
    "supportedExternalScript": [
      {
        "scriptType": "python",
        "supportingOS": [
          "Linux"
        ]
      }
    ],
    "maxScriptSize": 1024
  }
}
```

Parameters	Description
sessionId	Session id as a header parameter.

updateScriptHostData

Use this API to update script command and host data.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/script/update

Body Param

```
{
  "command": "shell",
  "scriptId": "2255",
  "hostId": "6688,5566"
}
```

Response:

```
{  
  "status": "success",  
  "statusCode": "1000",  
  "statusMessage": ""  
}
```

Parameters	Description
sessionid	Session id as a header parameter.
scriptHostDataToUpdate	Host details for which you want to update the script data.

Upgrading Resiliency Directory – IRestRDUpgrade

This interface and corresponding APIs allows you to execute REST request to upgrade the Actifio Resiliency Director. This interface contains the following RESTful APIs:

- [fetchUpdateHistory](#)
- [getReleaseNote](#)
- [listRDUpdate](#)
- [removeRDUpdates](#)
- [upgradeRDServer](#)
- [uploadUpdateFile](#)
- [checkUpdateInProgress](#)

fetchUpdateHistory

Use this API to return server update history.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/rdupgrade/fetchupdatehistory

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "_listUpgrageHistory": [
      {
        "rdversion": "6.1.3.410",
        "binaryversion": "6.1.3.410",
        "schemaversion": "6.1.3.410",
        "migrationstatus": "SUCCESS",
        "statusmessage": "SUCCESS",
        "starttime": "02/19/2015 13:15:31",
        "endtime": "02/19/2015 13:15:31",
        "baseVersion": ""
      }
    ],
    "_currentStatus": {
      "rdversion": "6.1.3.410",
      "binaryversion": "6.1.3.410",
      "schemaversion": "6.1.3.410",
      "migrationstatus": "SUCCESS",
      "statusmessage": "SUCCESS",
      "starttime": "02/19/2015 13:15:31",
      "endtime": "02/19/2015 13:15:31",
      "baseVersion": "1.0.66.0"
    }
  }
}
```

Parameters	Description
sessionid	Session id as a header parameter.

getReleaseNote

Use this API to download the help file given as query parameter.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/rdupgrade/releasenote

Response

Downloads the Release Note in PDF format.

Parameters	Description
sessionid	Session id as a header parameter.

listRDUpdate

Use this API to return valid updates present on server.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/rdupgrade/listupdate

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "rdversion": "6.1.3",
    "svnrevision": "457",
    "filename": "6.1.3.457",
    "invalid": true
  }
}
```

Parameters	Description
sessionid	Session id as a header parameter.

removeRDUpdates

Use this API will remove all valid/invalid updates present on the server.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/rdupgrade/removeupdates

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

```
}
```

Parameters	Description
sessionId	Session id as a header parameter.

upgradeRDServer

Use this API to initiate server upgrade.

Method: POST

Request

`https://{HOST}:{PORT}/armnew/api/rdupgrade/upgradeserver`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Upgrade started successfully. All services will be restarted. RD UI will be
inaccessible for 10-15 mins",
  "responseData": ""
}
```

Parameters	Description
sessionId	Session id as a header parameter.

uploadUpdateFile

Use this API to upload update file on server in update stage directory. It will extract .gpg file in the same directory and delete same GPG file. If upgradenow parameter is true then it will immediately start upgrade once upload completes.

Method: POST

Request

`https://{HOST}:{PORT}/armnew/api/rdupgrade/
uploadupdate?uploadcomponentname=uploader&upgradenow=true`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Upgrade started successfully. All services will be restarted. RD UI will be
inaccessible for 10-15 mins",
  "responseData": ""
}
```

Parameters	Description
configParams	Configuration details for Resiliency Director system installation.

checkUpdateInProgress

Use this API to check whether the upgrade process in progress or not.

Method: GET

Request

https://{HOST}:{PORT}/armnew/api/rdupgrade/checkupgradeinprogress

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": false
}
```

Parameters	Description
sessionid	Session id as a header parameter.

Additional APIs- IRestMiscellaneous

This interface provide API to execute REST request on RD server for various miscellaneous calls:

[fetchRecoveryTypes](#)

[fetchAppclassProperties](#)

[fetchappnamepresentonhost](#)

fetchRecoveryTypes

Use this API to get all the available recovery types.

Method: GET

Request

`https://{HOST}/armnew/api/fetchrecoverytypes?_=1619004063511`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": {
    "Databases": [
      {
        "recoveryTypeId": 3,
        "name": "Recover Oracle Database",
        "recoveryTypeOrder": "2"
      },
      {
        "recoveryTypeId": 4,
        "name": "Recover SQL Server Database",
        "recoveryTypeOrder": "2",
        "description": "Select SQL Server databases, instances, availability groups, or consistency groups of databases to recover at the DR site."
      },
      {
        "recoveryTypeId": 6265,
        "name": "Recover POSTGRESQL Instance",
        "recoveryTypeOrder": "1",
        "description": "Select POSTGRESQL applications to recover at the DR site"
      },
      {
        "recoveryTypeId": 6263,
        "name": "Recover MYSQL Instance",
        "recoveryTypeOrder": "1",
        "description": "Select MYSQL applications to recover at the DR site"
      },
      {
        "recoveryTypeId": 6260,
        "name": "Recover MARIADB Instance",
        "recoveryTypeOrder": "1",
        "description": "Select MARIADB applications to recover at the DR site"
      }
    ],
    "VM Servers and File Systems": [
      {
        "recoveryTypeId": 1,
        "name": "Recover VM From VMWare VM Backup",
        "recoveryTypeOrder": "1",
```

```

        "description": "Add VMs you wish to recover at the DR site. Each VM may be recovered
only once and will retain its original host name."
    },
    {
        "recoveryTypeId": 2,
        "name": "Create new VM from VMware VM",
        "recoveryTypeOrder": "1",
        "description": "Select VMs to use like a template to create a new VM. The same source
VM backup may be used to create many new VMs. Each new VM is given a new host name and IP and
may optionally be joined to a domain."
    },
    {
        "recoveryTypeId": 5,
        "name": "Recover Filesystem",
        "recoveryTypeOrder": "2",
        "description": "Select filesystems or consistency groups of filesystems to recover at
the DR site."
    }
]
}
}

```

Parameters	Description
sessionid	Session id as a header parameter.

fetchAppclassProperties

Use this API to get the appclass properties based on the appclass name passed as parameter.

Method: GET

Request

https://{HOST}/armnew/api/
fetchappclassproperties?name=SQLServer&appId=&cdsId=5125&_=1549349225240

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "appClassPropertiesT0": {
        "propertyId": 7198,
        "name": "SQLServer",
        "propertyName": "sqlinstance",
        "propertyType": "EDITABLE_SELECT",
        "propertyLabel": "SQL Server Instance Name",
        "isRequired": true,
        "description": "Select the target SQL Server Instance. The new database will be managed
by the selected SQL Serverinstance."
      },
      "appid": 0,
      "parameterId": 0
    },
    {
      "appClassPropertiesT0": {
        "propertyId": 7199,
        "name": "SQLServer",
        "propertyName": "dbname",
        "propertyType": "STRING",
        "propertyLabel": "SQL Server Database Name",
        "isRequired": true,
        "description": "Specify name for the new SQL Server database to be provisioned. Valid
characters include letters, numbers, @, #, -, _. No leading or trailing spacesallowed."
      },
      "appid": 0,
      "parameterId": 0
    },
    {
      "appClassPropertiesT0": {
        "propertyId": 7200,
        "name": "SQLServer",
        "propertyName": "recover",
        "propertyType": "BOOLEAN",
        "propertyLabel": "Recover Database After Restore",
        "isRequired": false,
        "description": "If selected, brings the newly created databaseonline."
      },
      "appid": 0,
      "parameterId": 0
    },
    {
      "appClassPropertiesT0": {
        "propertyId": 7201,
        "name": "SQLServer",
```

```

        "propertyName": "userlogins",
        "propertyType": "BOOLEAN",
        "propertyLabel": "Recover User Logins",
        "isRequired": false,
        "description": "If selected, recover the user logins of the database. This may require
user credentials to be entered."
    },
    "appid": 0,
    "parameterId": 0
},
{
    "appClassPropertiesT0": {
        "propertyId": 7202,
        "name": "SQLServer",
        "propertyName": "username",
        "propertyType": "STRING",
        "propertyLabel": "User Name",
        "isRequired": false,
        "description": "Specify user name for database provisioning. Only required when the
account running the connector (typically 'Local System') does not have privileges to apply
transaction logs, or to detach a database (required during subsequent unmount).\"
    },
    "appid": 0,
    "parameterId": 0
},
{
    "appClassPropertiesT0": {
        "propertyId": 7203,
        "name": "SQLServer",
        "propertyName": "password",
        "propertyType": "ENCRYPT",
        "propertyLabel": "Password",
        "isRequired": false,
        "description": "Password for the specified user.\"
    },
    "appid": 0,
    "parameterId": 0
}
}
]
}

```

Parameters	Description
sessionid	Session id as a header parameter.
name	Name of the Application class. <i>Note: The response will vary based on the name parameter.</i>
appid	Application ID of the application in RD database.
cdsid	ID of the Actifio Appliance in RD database.

fetchappnamepresentonhost

Use this API to get the SQL instance names available on the host.

Method: GET

Request

https://{HOST}/armnew/api/fetchappnamepresentonhost?hostname=Win2012r2-2&cdsclusterid=1415039054&_=1548925458841

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "sqlInstanceName": "WINVVM",
      "cdsappid": "233024",
      "sourceclusterid": "1415039054",
      "cdshostid": "231510"
    }
  ]
}
```

Parameters	Description
sessionid	Session id as a header parameter.
hostname	Name of the host on which you are trying to fetch the SQL instances.
cdsclusterid	Cluster ID of the Actifio Appliance on which the host resides.

Downloading Help – IRestHelpFileDownload

This interface and corresponding RESTful API allows you to download the help file. This interface contains the following the RESTful API:

[getHelpFile](#)

getHelpFile

Use this API to download the help file given as query parameter.

Method: GET

Request

`https://{HOST}:{PORT}/armnew/api/helpfile/download/ReleaseNote.pdf`

Response:

Returns the file in the PDF format.

Parameters	Description
sessionid	Session id as a header parameter.
file	File name to download.

Update Password

Use this API to update password of the Resiliency Director Server/collector with the parameters provided in the request.

Method: POST

Request

`https://{HOST}:{PORT}/armnew/api/resetuserpasswd`

Body Param

```
{
  "username": "admin",
  "oldpassword": "cGFzc3dvcmRuZXc=",
  "newpassword": "cGFzc3dvcmQ="
}
```

Managing System

This interface and its corresponding REST APIs allow you to reboot and shutdown the Resiliency Director. This interface contains the following RESTful APIs:

- Reboot
- Shutdown

Reboot

Use this API to reboot the Resiliency Director.

Method: POST

Request:

`https://{HOST}/armnew/api/system/reboot`

Shutdown

Use this API to shutdown the Resiliency Director

Method: POST

Request:

`https://{HOST}/armnew/api/system/shutdown`

Using the Resiliency Director Cloud Recovery RESTful APIs

The following interfaces and RESTful APIs are available only for the Resiliency Director Cloud Recovery.

[Managing AGM](#) on page 190

[Managing Appliance Groups](#) on page 193

[Managing Clouds](#) on page 199

[Managing Networks](#) on page 202

[Managing Application Groups](#) on page 206

[Managing Recovery Plans](#) on page 221

Managing AGM

This interface and its corresponding REST APIs allow you to add, list, edit, and delete the AGM used by the Resiliency Director. This interface contains the following RESTful APIs:

[Add AGM](#)

[List AGM](#)

[Edit AGM](#)

[Remove AGM](#)

Add AGM

Use this API to add an AGM in the Resiliency Director appliance with the parameters provided in the request.

Method: POST

Request

`https://{HOST}/armnew/api/agm/add`

Body Param

```
{
  "hostOrIp": "172.16.201.117",
  "username": "admin",
  "password": "cGFzc3dvcmQ=",
  "listOrganizations": []
}
```

Response

```
{
  "status": "success",
  "statusCode": "10505",
  "statusMessage": "agm server added successfully.",
  "responseData": ""
}
```

Parameters	Description
hostOrIp	Host address or IP address.
username	Username for the AGM
password	Desired password.

List AGM

Use this API to list the available AGM in the Resiliency Director appliance.

Method: GET

Request

https://{HOST}/armnew/api/agm/fetch?_=1572512635343

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 5216,
      "hostOrIp": "172.16.200.175",
      "username": "admin",
      "password": "",
      "port": 0,
      "listOrganizations": [
      ],
      "version": "10.0.2.4910",
      "status": "Online"
    }
  ]
}
```

Parameters	Description
id	Id of the AGM.
hostOrIp	Host address or IP address.
username	Username for the AGM

Edit AGM

Use this API to edit an AGM in the Resiliency Director appliance.

Method: POST

Request

https://{HOST}/armnew/api/agm/update

Body Param

```
{
  "hostOrIp": "172.16.201.117",
  "username": "admin",
  "password": "cGFzc3dvcmQ=",
  "listOrganizations": [],
  "id": "5216"
}
```

Response

```
{
```

```

"status": "success",
"statusCode": "1000",
"statusMessage": "Operation performed successfully.",
"responseData": ""
}

```

Parameters	Description
hostOrIp	Host address or IP address.
username	Username for the AGM

Remove AGM

Use this API to remove an AGM from the Resiliency Director appliance.

Method: GET

Request

https://{HOST}/armnew/api/agm/remove?ids=5216&_=1572512635344

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}

```

Parameters	Description
status	Status of AGM delete process.
statusCode	Code for the status.
statusMessage	Message displayed after an AGM is deleted.

Managing Appliance Groups

This interface and its corresponding REST APIs allow you to add, list, edit, and delete Appliance Group used by the Resiliency Director. This interface contains the following RESTful APIs:

- [Add Appliance Group](#)
- [Edit Appliance Group](#)
- [List Appliance Group](#)
- [Remove Appliance Group](#)

Add Appliance Group

Use this API to add an Appliance Group to the Resiliency Director appliance.

Method: POST

Request

`https://{HOST}/armnew/api/appliancegroup/add`

Body Param

```
{
  "appliancegroupname": "EngGroup",
  "storageserverToList": [
    {
      "id": 151557,
      "hostOrIp": "172.23.50.88",
      "username": "admin",
      "password": "",
      "port": 0,
      "useHttps": false,
      "firmwareVersion": "",
      "model": "",
      "vendorkey": "vendorKey",
      "clustername": "rdawstarget",
      "listApplianceGrps": [
        {
          "appliancegroupid": 123979,
          "appliancegroupname": "Test_Appliancegroup"
        },
        {
          "appliancegroupid": 151559,
          "appliancegroupname": "testappgroup"
        }
      ]
    },
    {
      "id": 87160,
      "hostOrIp": "172.23.16.36",
      "username": "admin",
      "password": "",
      "port": 0,
      "useHttps": false,
      "firmwareVersion": "",
      "model": "",
      "vendorkey": "vendorKey",

```

```

    "clustername": "rdgcponvault",
    "listApplianceGrps": [
      {
        "appliancegroupid": 64905,
        "appliancegroupname": "ApplianceGrp2"
      },
      {
        "appliancegroupid": 125222,
        "appliancegroupname": "AppGcp"
      }
    ],
    "listOrganizations": [
      {
        "id": 45633,
        "name": "o5",
        "scheduleEnable": false,
        "parentOrgId": 0
      }
    ],
    "selected": false,
    "emailConfigured": false,
    "authenticationSource": false,
    "clusterId": 1415048803
  }
]
}

```

Response

```

{
  "status": "success",
  "statusCode": "10524",
  "statusMessage": "appliancegroup added successfully.",
  "responseData": ""
}

```

Parameters	Description
hostOrIp	Host address or IP address.
username	Username for the Appliance group.
password	Desired password.

Edit Appliance Group

Use this API to edit an appliance group in the Resiliency Director appliance.

Method: POST

Request

<https://{HOST}/armnew/api/appliancegroup/update>

Body Param

```

{
  "appliancegroupname": "EngGroupNew",
  "storageserverToList": [
    {

```



```

    "id": 151557,
    "hostOrIp": "172.23.50.88",
    "username": "admin",
    "password": "",
    "port": 0,
    "useHttps": false,
    "firmwareVersion": "",
    "model": "",
    "vendorkey": "vendorKey",
    "clustername": "rdawstarget",
    "listApplianceGrps": [
      {
        "appliancegroupid": 123979,
        "appliancegroupname": "Test_Appliancegroup"
      },
      {
        "appliancegroupid": 151559,
        "appliancegroupname": "ag_user"
      },
      {
        "appliancegroupid": 157773,
        "appliancegroupname": "EngGroup"
      }
    ],
    "listOrganizations": [],
    "selected": false,
    "emailConfigured": false,
    "authenticationSource": true,
    "clusterId": 1415043690
  },
  {
    "id": 87160,
    "hostOrIp": "172.23.16.36",
    "username": "admin",
    "password": "",
    "port": 0,
    "useHttps": false,
    "firmwareVersion": "",
    "model": "",
    "vendorkey": "vendorKey",
    "clustername": "rdgcpnvault",
    "listApplianceGrps": [
      {
        "appliancegroupid": 64905,
        "appliancegroupname": "ApplianceGrp2"
      },
      {
        "appliancegroupid": 125222,
        "appliancegroupname": "AppGcp"
      },
      {
        "appliancegroupid": 157773,
        "appliancegroupname": "EngGroup"
      }
    ],
    "listOrganizations": [
      {
        "id": 45633,
        "name": "o5",
        "scheduleEnable": false,

```

```

        "parentOrgId": 0
    }
],
    "selected": false,
    "emailConfigured": false,
    "authenticationSource": false,
    "clusterId": 1415048803
}
],
    "appliancegroupid": 157773
}

```

Response

```

{
    "status": "success",
    "statusCode": 1000,
    "statusMessage": "Operation performed successfully.",
    "responseData": ""
}

```

List Appliance Group

Use this API to list the Appliance Group available in the Resiliency Director appliance.

Method: GET

Request

https://{HOST}/armnew/api/appliancegroup/fetch?_=1572936313528

Response

```

{
    "status": "success",
    "statusCode": "1000",
    "statusMessage": "Operation performed successfully.",
    "responseData": [
        {
            "appliancegroupid": 64802,
            "appliancegroupname": "testAG",
            "storageserverToList": [
                {
                    "id": 64899,
                    "hostOrIp": "172.16.201.187",
                    "username": "admin",
                    "password": "",
                    "port": 0,
                    "useHttps": false,
                    "vendorkey": "VendorKey",
                    "clustername": "sourcesky",
                    "selected": false,
                    "emailConfigured": false,
                    "authenticationSource": false,
                    "clusterId": 141807825785
                }
            ]
        },
        {
            "appliancegroupid": 123979,
            "appliancegroupname": "Test_Appliancegroup",
            "storageserverToList": [

```

```

    {
      "id": 151557,
      "hostOrIp": "172.23.50.88",
      "username": "admin",
      "password": "",
      "port": 0,
      "useHttps": false,
      "firmwareVersion": "",
      "model": "",
      "vendorkey": "vendorKey",
      "clustername": "rdawstarget",
      "selected": false,
      "emailConfigured": false,
      "authenticationSource": true,
      "clusterId": 1415043690
    }
  ]
},
{
  "appliancegroupid": 157773,
  "appliancegroupname": "EngGroupNew",
  "storageserverToList": [
    {
      "id": 87160,
      "hostOrIp": "172.23.16.36",
      "username": "admin",
      "password": "",
      "port": 0,
      "useHttps": false,
      "firmwareVersion": "",
      "model": "",
      "vendorkey": "vendorKey",
      "clustername": "rdgcponvault",
      "selected": false,
      "emailConfigured": false,
      "authenticationSource": false,
      "clusterId": 1415048803
    },
    {
      "id": 151557,
      "hostOrIp": "172.23.50.88",
      "username": "admin",
      "password": "",
      "port": 0,
      "useHttps": false,
      "firmwareVersion": "",
      "model": "",
      "vendorkey": "vendorKey",
      "clustername": "rdawstarget",
      "selected": false,
      "emailConfigured": false,
      "authenticationSource": true,
      "clusterId": 1415043690
    }
  ]
}
]
}
}

```

Remove Appliance Group

Use this API to remove the Appliance Group from the Resiliency Director appliance.

Method: GET

Request

`https://{HOST}/armnew/api/appliancegroup/remove?ids=157773&_=1572936313529`

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Managing Clouds

This interface and its corresponding REST APIs allow you to add, list, edit, and delete Cloud used by the Resiliency Director. This interface contains the following RESTful APIs:

[Add Cloud](#)
[Edit Cloud](#)
[List Cloud](#)
[Delete Cloud](#)

Add Cloud

Use this API to add a Cloud to the Resiliency Director appliance.

Method: POST

Request

https://{HOST}/armnew/api/cloud/add

Body Param

```
{
  "listOrganizations": [],
  "name": "EngCloud",
  "cloudtype": "AWS",
  "volumetype": "Magnetic",
  "iops": "200",
  "regioncode": "us-east-1",
  "networkid": "admin",
  "accesskeyid": "password",
  "secretkey": "Test1"
}
```

Response

```
{
  "status": "success",
  "statusCode": "10515",
  "statusMessage": "Cloud definition added successfully.",
  "responseData": ""
}
```

Parameters	Description
name	Name for the Cloud connection.
cloudtype	Type of Cloud connection registered.
networkid	Network id for the cloud connection.

Edit Cloud

Use this API to edit a Cloud to the Resiliency Director appliance.

Method: POST

Request

https://{HOST}/armnew/api/cloud/update

Body Param

```
{
  "listOrganizations": [],
  "name": "EngCloud2",
  "cloudtype": "AWS",
  "volumetype": "Magnetic",
  "iops": "200",
  "regioncode": "us-east-1",
  "networkid": "admin",
  "accesskeyid": "password",
  "secretkey": "",
  "id": 157895
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
name	Name for the Cloud connection.
cloudtype	Type of Cloud connection registered.
networkid	Network id for the cloud connection.

List Cloud

Use this API to list the available Clouds to the Resiliency Director appliance.

Method: GET

Request

https://{HOST}/armnew/api/cloud/fetch?_=1572936313538

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 5218,
      "name": "AWS",
      "cloudtype": "AWS",
      "diskvolumetype": "Magnetic",
      "volumetype": "Magnetic",
      "iops": "100",
      "regioncode": "us-east-1",
      "networkid": "vpc-0600f8ea46862480b",
      "accesskeyid": "AKIARU34P7QOF3NHFP2I",
      "secretkey": ""
    }
  ]
}
```

```

        "tags": "",
        "status": "Online",
        "encryption": false,
        "authKey": "",
        "listOrganizations": [
    ]
},
{
    "id": 5220,
    "name": "GCP",
    "cloudtype": "GCP",
    "diskvolumetype": "Standard persistent disk",
    "volumetype": "Standard persistent disk",
    "regioncode": "us-east1",
    "secretkey": "",
    "zone": "us-east1-b",
    "gcpkeys": "",
    "keyfilename": "GCP.txt",
    "tags": "",
    "status": "Online",
    "encryption": false,
    "authKey": "",
    "listOrganizations": [
    ]
}
]
}

```

Delete Cloud

Use this API to delete a Cloud from the Resiliency Director appliance.

Method: GET

Request

https://{HOST}/armnew/api/cloud/remove?ids=157895&_=1572937277284

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}

```

Managing Networks

This interface and its corresponding REST APIs allow you to add, list, edit, and delete Network used by the Resiliency Director. This interface contains the following RESTful APIs:

- [Add Network](#)
- [Edit Network](#)
- [List Network](#)
- [Delete Network](#)
- [Add Appliance to an Appliance Group](#)

Add Network

Use this API to add a Cloud to the Resiliency Director appliance.

Method: POST

Request

`https://{HOST}/armnew/api/network/add`

Body Param

```
{
  "networkname": "Sample_AWS_Network",
  "cloudtype": "AWS",
  "listOrganizations": [],
  "subnetid": "subnetid1",
  "securitygroupid": "test11"
}
```

Response :

```
{
  "status": "success",
  "statusCode": "10508",
  "statusMessage": "network added successfully.",
  "responseData": ""
}
```

Parameters	Description
networkname	Name for the Network connection.
cloudtype	Type of Cloud connection registered.
subnetid	Subnet id for the Network connection.

Edit Network

Use this API to edit a Cloud in the Resiliency Director appliance.

Method: POST

Request

https://{HOST}/armnew/api/network/update

Body Param

```
{
  "networkname": "Test_NetworkNew",
  "cloudtype": "AWS",
  "listOrganizations": [],
  "subnetid": "admin",
  "securitygroupid": "admin,admin_test",
  "id": 125820
}
```

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}
```

Parameters	Description
networkname	Name for the Network connection.
cloudtype	Type of Cloud connection registered.
subnetid	Subnet id for the Network connection.

List Network

Use this API to list the available Network in the Resiliency Director appliance.

Method: GET

Request

https://{HOST}/armnew/api/network/fetch?_=1572943410746

Response

```
{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "id": 57351,
      "networkname": "GCP_network",
      "cloudtype": "GCP",
      "networkid": "default",
      "subnetid": "engineering-east-1",
      "listOrganizations": []
    }
  ]
}
```

```

    },
    {
      "id": 125820,
      "networkname": "Test_Network",
      "cloudtype": "AWS",
      "subnetid": "admin",
      "securitygroupid": "admin,admin_test",
      "listOrganizations": []
    },
    {
      "id": 125959,
      "networkname": "Network_AWS",
      "cloudtype": "AWS",
      "subnetid": "admin",
      "securitygroupid": "root_demo",
      "listOrganizations": []
    },
    {
      "id": 125961,
      "networkname": "Network_gcp",
      "cloudtype": "GCP",
      "networkid": "test_user",
      "subnetid": "admin",
      "listOrganizations": []
    },
    {
      "id": 57357,
      "networkname": "AWS_Network",
      "cloudtype": "AWS",
      "subnetid": "subnet-0fa435f6b51b25f77",
      "securitygroupid": "sg-01dd906a26159d705",
      "listOrganizations": []
    }
  ]
}

```

Delete Network

Use this API to delete the Network from the Resiliency Director appliance.

Method: GET

Request

https://{HOST}/armnew/api/network/remove?ids=125820&_=1572943410753

Response

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": ""
}

```

Add Appliance to an Appliance Group

Use this API to add appliances to an Appliance Group in the Resiliency Director.

Method: POST

Request

https://{HOST}/armnew/api/cds/add

Body Param

```
{
  "hostOrIp": "172.16.201.119",
  "username": "admin",
  "password": "cGFzc3dvcmQ=",
  "listOrganizations": [],
  "listApplianceGrps": [
    {
      "appliancegroupid": 50178,
      "appliancegroupname": "Grp1"
    }
  ]
}
```

Response :

```
{
  "status": "success",
  "statusCode": "1005",
  "statusMessage": "Actifio Appliance added successfully.",
  "responseData": ""
}
```

Parameters	Description
hostOrIp	Host address or IP address.
username	Username for the AGM
password	Desired password.

Managing Application Groups

This interface and its corresponding REST APIs allows you to fetch the details of a VM or system state, or add an application group to a VM or system state in the Resiliency Director Cloud Recovery. This interface contains the following RESTful APIs:

- [FetchVmDetails](#)
- [Add Application Group to a VM or System state](#)
- [Add Application Group to a VM and CAF Database](#)
- [Update Application Group](#)
- [Remove Application Group](#)

FetchVmDetails

Use this API to fetch the details of a VM or system state from an AGM.

Method: POST

Request

`https://{HOST}/armnew/api/applicationgroup/
fetchvmdetails?agmid=160779&uniqueKey=1573798452394&appgroupname=AGCloud`

Body Param

```
[  
  {  
    "memberOrderIndex": 1,  
    "delay": 0,  
    "vmMembers": [  
      {  
        "memberUuId": "Win2012_100",  
        "memberName": "Win2012_100",  
        "uniqueGridIndex": "9733",  
        "vm": true,  
        "executionStatus": "",  
        "cdsAppId": "556199",  
        "agmAppId": "400160",  
        "cdsClusterId": "141807825785",  
        "targetClusterId": "1415057143",  
        "clustername": "sourcesky",  
        "targetClusterName": "rdgcptarget",  
        "validForAppGroup": false,  
        "displayAppName": "Win2012_100",  
        "displayAppType": "VM",  
        "apptype": "VM",  
        "sourceHostName": "win2012_100",  
        "recoveryType": {  
          "recoveryTypeId": 6,  
          "name": "Recover VMs or Physical Servers to Cloud",  
          "recoveryTypeOrder": "1"  
        },  
        "infoRequired": false,  
        "policyOpCode": 1,  
        "vmalreadyPresentInRecPlan": false,  
        "sequence": "1",  
        "originalDisplayAppName": "Win2012_100"  
      }  
    ],  
    "appMembers": []  
  }  
]
```

Response :

```

{
  "status": "success",
  "statusCode": "1000",
  "statusMessage": "Operation performed successfully.",
  "responseData": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "vmMembers": [
        {
          "memberUuId": "Win2012_100",
          "memberName": "Win2012_100",
          "uniqueGridIndex": "9733",
          "vm": true,
          "executionStatus": "",
          "cdsAppId": "556199",
          "agmAppId": "400160",
          "cdsClusterId": "141807825785",
          "targetClusterId": "1415057143",
          "clustername": "sourcesky",
          "targetClusterName": "rdgcptarget",
          "validForAppGroup": false,
          "displayAppName": "Win2012_100",
          "displayAppType": "VM",
          "apptype": "VM",
          "sourceHostName": "win2012_100",
          "recoveryType": {
            "recoveryTypeId": 6,
            "name": "Recover VMs or Physical Servers to Cloud",
            "recoveryTypeOrder": "1"
          },
          "infoRequired": false,
          "policyOpCode": 1,
          "id": 0,
          "vappName": "",
          "numberOfCpus": 4,
          "actualNumberOfNics": 1,
          "testNumberOfNics": 0,
          "testMemoryInMbs": 4096,
          "actualMemoryInMbs": 4096,
          "shutdownOrder": 0,
          "numberOfVirtualDisks": 0,
          "hddInGbs": 0,
          "hostName": "",
          "testCPUAllocation": 4,
          "actualCPUAllocation": 4,
          "prescript": "",
          "postscript": "",
          "username": "",
          "passwd": "",
          "nicNames": [
            {
              "deviceConfigId": 0,
              "macAddress": "",
              "interfaceLabel": "Network adapter 1",
              "pciSlotNumber": 0
            }
          ],
          "vmAllNicDataList": [
            {

```

```

        "interfaceLabel": "Network adapter 1",
        "enforcedStaticIp": true,
        "enforcedStaticDnsIp": false,
        "ipAddressToConfigure": "",
        "gateway": "",
        "macAddress": "",
        "pciSlotNumber": 0
    }
],
"sourceIp": "",
"osName": "windows",
"coresPerSocket": 0,
"testCoresPerSocket": 0,
"virtualSocket": 0,
"testVirtualSocket": 0,
"sequenceNumber": 0,
"bootdisksize": 40,
"originalBootdisksize": 40,
"critical": false,
"vappStartupOrder": 0,
"vmalreadyPresentInRecPlan": false
}
],
"appMembers": []
}
]
}
}

```

Parameters	Description
memberOrderIndex	Index of the order in which the application group is fetched
delay	Sequence delay of the application group recovery

Add Application Group to a VM or System state

Use this API to add Application Group to a VM or system state in the Resiliency Director Cloud Recovery.

Method: POST

Request

https://{HOST}/armnew/api/applicationgroup/
add?uniqueKey=1573798694934&appgroupname=AGCloud&suppressWarningNicValidation=true

Body Param

```

{
  "description": "",
  "name": "AGCloud",
  "sequenceOrder": [
    {
      "memberOrderIndex": 1,
      "delay": 0,
      "vmMembers": [
        {
          "memberUuId": "Win2012_100",
          "memberName": "Win2012_100",
          "uniqueGridIndex": "9733",

```

```

"vm": true,
"executionStatus": "",
"cdsAppId": "556199",
"agmAppId": "400160",
"cdsClusterId": "141807825785",
"targetClusterId": "1415057143",
"clustername": "sourcesky",
"targetClusterName": "rdgcptarget",
"validForAppGroup": false,
"displayAppName": "Win2012_100",
"displayAppType": "VM",
"apptype": "VM",
"sourceHostName": "win2012_100",
"recoveryType": {
  "recoveryTypeId": 6,
  "name": "Recover VMs or Physical Servers to Cloud",
  "recoveryTypeOrder": "1"
},
"infoRequired": false,
"policyOpCode": 1,
"id": 0,
"vappName": "",
"numberOfCpus": 4,
"actualNumberOfNics": 1,
"testNumberOfNics": 0,
"testMemoryInMbs": "4096",
"actualMemoryInMbs": "4096",
"shutdownOrder": 0,
"numberOfVirtualDisks": 0,
"hddInGbs": 0,
"hostname": "",
"testCPUAllocation": "4",
"actualCPUAllocation": "4",
"prescript": "",
"postscript": "",
"username": "",
"passwd": "",
"nicNames": [
  {
    "deviceConfigId": 0,
    "macAddress": "",
    "interfaceLabel": "Network adapter 1",
    "pciSlotNumber": 0
  }
],
"vmAllNicDataList": [
  {
    "interfaceLabel": "Network adapter 1",
    "enforcedStaticIp": true,
    "enforcedStaticDnsIp": false,
    "ipAddressToConfigure": "",
    "gateway": "",
    "macAddress": "",
    "pciSlotNumber": 0
  }
],
"sourceIp": "",
"osName": "windows",
"coresPerSocket": 0,
"testCoresPerSocket": 0,

```

```

        "virtualSocket": 0,
        "testVirtualSocket": 0,
        "sequenceNumber": 0,
        "bootdisksize": 40,
        "originalBootdisksize": 40,
        "critical": false,
        "vappStartupOrder": 0,
        "vmalreadyPresentInRecPlan": false,
        "sequence": "1",
        "originalDisplayAppName": "Win2012_100",
        "ip": ""
    }
],
"appMembers": []
}
],
"organizationToList": [],
"id": "",
"agmId": 160779
}

```

Response

```

{
  "status": "success",
  "statusCode": "1009",
  "statusMessage": "Application group added successfully with name: ",
  "responseData": ""
}

```

Parameters	Description
name	Name of the Application group.
sequence	Sequence order of the application group recovery

Add Application Group to a VM and CAF Database

Use this API to create an application group with VM in sequence 1 and CAF database – PostgreSQL in sequence 2 and Db2 in sequence 3.

Method: POST

Request: https://{host}/armnew/api/applicationgroup/add?uniqueKey=1618980254335&appgroupname=AG_Apps2&suppressWarningNicValidation=true

Body Param:

```

{
  "agmId": 6147,
  "description": "",
  "id": "",
  "name": "new_ag_VM_DB",
  "organizationToList": [
  ],
  "postScript": null,
  "postScriptType": null,

```



```

"postScriptVMName":null,
"postScriptVMUUID":null,
"preScript":null,
"preScriptType":null,
"preScriptVMName":null,
"preScriptVMUUID":null,
"sequenceOrder":[
  {
    "appMembers":[
    ],
    "delay":0,
    "memberOrderIndex":1,
    "vmMembers":[
      {
        "actualCPUAllocation":"2",
        "actualMemoryInMbs":8192,
        "actualNumberOfNics":1,
        "agmAppId":"387576",
        "appliancegroupid":0,
        "apptype":"VM",
        "bootdisksize":50,
        "cdsAppId":"9821",
        "cdsClusterId":"143428285568",
        "clustername":"SKY5SP4",
        "coresPerSocket":0,
        "critical":false,
        "displayAppName":"Cent7.5_MaxDB1_CAF1",
        "displayAppType":"VM",
        "executionStatus":"",
        "hddInGbs":0,
        "hostName":"",
        "id":0,
        "infoRequired":false,
        "ip":"",
        "isAsm":false,
        "memberName":"Cent7.5_MaxDB1_CAF1",
        "memberUuId":"Cent7.5_MaxDB1_CAF1",
        "nicNames":[
          {
            "deviceConfigId":0,
            "interfaceLabel":"Network adapter 1",
            "macAddress":"",
            "pciSlotNumber":0
          }
        ],
        "numberOfCpus":2,
        "numberOfVirtualDisks":0,
        "onVaultPoolHash":"-171848852,1917492342,-171848852,1917492342,-
171848852,1917492342",
        "originalBootdisksize":50,
        "originalDisplayAppName":"Cent7.5_MaxDB1_CAF1",
        "osName":"linux",
        "passwd":"",
        "policyOpCode":3,
        "postscript":"",
        "prescript":"",
        "recoveryType":{

```

"description":"Select VMware VMs or System State applications (connector backups) to convert into cloud instances for DR recovery to the cloud. Each source VM or server may be recovered only once and will retain its original host name.",

```

    "name":"Recover VMs or Physical Servers to Cloud",
    "recoveryTypeId":6,
    "recoveryTypeOrder":"1"
  },
  "sequence":"1",
  "sequenceNumber":0,
  "shutdownOrder":0,
  "sourceClusterName":"SKY5SP4",
  "sourceHostName":"cent7.5_maxdb1_caf1",
  "sourceIp":"",
  "targetClusterId":"144740187595",
  "targetClusterName":"awssky",
  "testCoresPerSocket":0,
  "testCPUALlocation":"2",
  "testMemoryInMbs":8192,
  "testNumberOfNics":0,
  "testVirtualSocket":0,
  "uniqueGridIndex":"109115490670860",
  "username":"",
  "validForAppGroup":false,
  "vappName":"",
  "vappStartupOrder":0,
  "virtualSocket":0,
  "vm":true,
  "vmAllNicDataList":[
    {
      "enforcedStaticDnsIp":false,
      "enforcedStaticIp":true,
      "gateway":"",
      "interfaceLabel":"Network adapter 1",
      "ipAddressToConfigure":"",
      "macAddress":"",
      "pciSlotNumber":0
    }
  ],
  "vmAlreadyPresentInRecPlan":false
}
],
},
{
  "appMembers":[
    {
      "agmAppId":"641258",
      "appClass":"POSTGRESQLGroup",
      "appClassMetaDataTOList":[
        {
          "appid":0,
          "parameterId":6298,
          "value":"5438"
        },
        {
          "appid":0,
          "parameterId":6299,
          "value":"admin"
        }
      ],
      {
        "appid":0,

```

```

        "parameterId":6300
    },
    {
        "appid":0,
        "parameterId":6301
    },
    {
        "appid":0,
        "parameterId":6302,
        "value":"password"
    }
],
"appName":"postgresql_5432",
"apptype":"POSTGRESQLInstance",
"cdsAppId":"696636",
"cdsClusterId":"145053552328",
"clustername":"SKY7SP4",
"critical":false,
"depth":0,
"displayAppName":"postgresql_5432(gtc_postgres_nonroot_host)",
"displayAppType":"POSTGRESQLInstance",
"exclusionList":"sandy1",
"executionStatus":"",
"failoverState":0,
"friendlyType":"POSTGRESQLInstance",
"fromMount":false,
"id":0,
"inclusionList":"",
"infoRequired":false,
"ip":"",
"isAsm":false,
"isClustered":false,
"iscustomapp":true,
"isGroup":false,
"isTbd":false,
"memberName":"postgresql_5432",
"memberUuId":"postgresql_5432",
"onVaultPoolHash":"1917492342,1917492342,1917492342",
"originalAppId":0,
"policyOpCode":3,
"recoveryType":{
    "description":"Select POSTGRESQL applications to recover at the DR site",
    "name":"Recover POSTGRESQL Instance",
    "recoveryTypeId":6170,
    "recoveryTypeOrder":"1"
},
"scriptParameterTOList":[

],
"sequenceNumber":0,
"sourceClusterName":"SKY7SP4",
"sourceHostId":633887,
"sourceHostName":"gtc_postgres_nonroot_host",
"targetClusterId":"144690228738",
"targetClusterName":"rd10skysp4",
"targetHostId":0,
"targetHostName":"cent75maxdb1caf1",
"targetList":"",
"validForAppGroup":false,
"vm":false,

```

```

        "vmalreadyPresentInRecPlan":false,
        "volumesToList":[
            ]
        }
    ],
    "delay":0,
    "memberOrderIndex":2,
    "vmMembers":[
    ]
},
{
    "appMembers":[
        {
            "agmAppId":"721689",
            "appClass":"DB2Group",
            "appClassMetaDataTOList":[
                {
                    "appid":0,
                    "parameterId":6418,
                    "value":"new_db2_instance"
                },
                {
                    "appid":0,
                    "parameterId":6419
                }
            ],
            "appName":"inst1",
            "apptype":"DB2Instance",
            "cdsAppId":"726455",
            "cdsClusterId":"145053552328",
            "clustername":"SKY7SP4",
            "critical":false,
            "depth":0,
            "displayAppName":"inst1(db2_pass2_root)",
            "displayAppType":"DB2Instance",
            "exclusionList":"",
            "executionStatus":"",
            "failoverState":0,
            "friendlyType":"DB2Instance",
            "fromMount":false,
            "id":0,
            "inclusionList":"",
            "infoRequired":false,
            "ip":"",
            "isAsm":false,
            "isClustered":false,
            "iscustomapp":true,
            "isGroup":false,
            "isTbd":false,
            "memberName":"inst1",
            "memberUuId":"inst1",
            "onVaultPoolHash":"1917492342",
            "originalAppId":0,
            "policyOpCode":3,
            "recoveryType":{
                "description":"Select DB2 applications to recover at the DR site",
                "name":"Recover DB2 Instance",
                "recoveryTypeId":6164,

```

```

        "recoveryTypeOrder": "1"
    },
    "scriptParameterTOList": [
        ],
        "sequenceNumber": 0,
        "sourceClusterName": "SKY7SP4",
        "sourceHostId": 721377,
        "sourceHostName": "db2_pass2_root",
        "targetClusterId": "144690228738",
        "targetClusterName": "rd10skysp4",
        "targetHostId": 0,
        "targetHostName": "cent75maxdb1caf1",
        "targetList": "",
        "validForAppGroup": false,
        "vm": false,
        "vmaAlreadyPresentInRecPlan": false,
        "volumesToList": [
            ]
        ]
    },
    "delay": 0,
    "memberOrderIndex": 3,
    "vmMembers": [
        ]
    ]
}

```

Response:

```

{
  "status": "success",
  "statusCode": "1017",
  "statusMessage": "Application group added successfully. ",
  "responseData": ""
}

```

Update Application Group

Use this API to update the application group with VM in sequence 1 and CAF database – PostgreSQL in sequence 2 and Db2 in sequence 2.

Method: POST

Request: https://{host}/armnew/api/applicationgroup/update?uniqueKey=1618981068103&appgroupname=AG_Apps2&suppressWarningNicValidation=true

Body Param:

```

{
  "agmId": 6147,
  "description": "",
  "id": 6498,
  "name": "new_ag_VM_DBs",
  "organizationToList": [
    ],
  "postScript": null,
  "postScriptType": null,
  "postScriptVMName": null,

```

```

"postScriptVMUUID":null,
"preScript":null,
"preScriptType":null,
"preScriptVMName":null,
"preScriptVMUUID":null,
"sequenceOrder":[
  {
    "appMembers":[
    ],
    "delay":0,
    "memberOrderIndex":1,
    "vmMembers":[
      {
        "actualCPUAllocation":"2",
        "actualMemoryInMbs":8192,
        "actualNumberOfNics":1,
        "agmAppId":"387576",
        "appliancegroupid":0,
        "apptype":"VM",
        "bootdisksize":50,
        "cdsAppId":"9821",
        "cdsClusterId":"143428285568",
        "clustername":"SKY5SP4",
        "consumedCloudTOList":[
          {
            "cloudid":6159,
            "consumedCloudId":6504,
            "inherited":true,
            "recoveryplanid":6462,
            "virtualMachineId":6503
          }
        ],
        "consumedMigrationTOList":[
          {
            "consumedMigrationId":6505,
            "consumedMigrationValue":"Migrate for Actual Only",
            "inherited":true,
            "recoveryplanid":6462,
            "virtualMachineId":6503
          }
        ],
        "consumedNetworkTOList":[
          {
            "consumedNetworkId":6506,
            "inherited":true,
            "mode":"both",
            "networkid":6161,
            "recoveryplanid":6462,
            "virtualMachineId":6503
          },
          {
            "consumedNetworkId":6507,
            "inherited":true,
            "mode":"both",
            "networkid":6161,
            "nicName":"Network adapter 1",
            "recoveryplanid":6462,
            "virtualMachineId":6503
          }
        ]
      }
    ]
  }
]

```

```

    ],
    "coresPerSocket":0,
    "critical":false,
    "cupResourceAllocation":{
        "corePerSocket":0,
        "limit":0,
        "numberOfCpus":0,
        "reservation":0,
        "shares":0,
        "sharesLevel":0
    },
    "defaultDnsIpToconfigure":[

    ],
    "displayAppName":"Cent7.5_MaxDB1_CAF1",
    "displayAppType":"VM",
    "executionStatus":"",
    "hddInGbs":0,
    "hostName":"",
    "id":6503,
    "infoRequired":false,
    "ip":"",
    "isAsm":false,
    "memberName":"Cent7.5_MaxDB1_CAF1",
    "memberUuId":"Cent7.5_MaxDB1_CAF1",
    "memoryResourceAllocation":{
        "corePerSocket":0,
        "limit":0,
        "numberOfCpus":0,
        "reservation":0,
        "shares":0,
        "sharesLevel":0
    },
    "nicForDns":"",
    "nicForStaticIp":"",
    "nicNames":[
        {
            "deviceConfigId":0,
            "interfaceLabel":"Network adapter 1",
            "macAddress":"",
            "nicType":"null",
            "pciSlotNumber":0
        }
    ],
    "numberOfCpus":0,
    "numberOfVirtualDisks":0,
    "onVaultPoolHash":"-171848852,1917492342,-171848852,1917492342,-
171848852,1917492342",
    "originalBootdisksize":0,
    "originalDisplayAppName":"Cent7.5_MaxDB1_CAF1",
    "osName":"linux",
    "passwd":"",
    "policyOpCode":3,
    "portGroup":[

    ],
    "postExternalScript":[

    ],
    "postscript":"",

```

```

    "preExternalScript":[
    ],
    "prescript":"",
    "recoveryType":{
        "description":"Select VMware VMs or System State applications (connector
backups) to convert into cloud instances for DR recovery to the cloud. Each source VM or server
may be recovered only once and will retain its original host name.",
        "name":"Recover VMs or Physical Servers to Cloud",
        "recoveryTypeId":6,
        "recoveryTypeOrder":"1"
    },
    "sequence":1,
    "sequenceNumber":1,
    "shutdownOrder":0,
    "sourceClusterName":"SKY5SP4",
    "sourceGateway":"",
    "sourceHostName":"",
    "sourceIp":"",
    "sourceSubnet":"",
    "targetClusterId":"144740187595",
    "targetClusterName":"awssky",
    "testCoresPerSocket":0,
    "testCPUAllocation":"2",
    "testMemoryInMbs":8192,
    "testNumberOfNics":1,
    "testVirtualSocket":0,
    "uniqueGridIndex":"110737274568818",
    "username":"",
    "validForAppGroup":false,
    "vappName":"",
    "vappStartupOrder":0,
    "virtualSocket":0,
    "vm":true,
    "vmAllNicDataList":[
        {
            "enforcedStaticDnsIp":false,
            "enforcedStaticIp":true,
            "gateway":"",
            "interfaceLabel":"Network adapter 1",
            "ipAddressToConfigure":"",
            "macAddress":"",
            "pciSlotNumber":0
        }
    ],
    "vmAlreadyPresentInRecPlan":false,
    "vmNewName":"recovered-cent75maxdb1caf1"
}
],
},
{
    "appMembers":[
        {
            "agmAppId":"641258",
            "appClass":"POSTGRESQLGroup",
            "appClassMetaDataTOList":[
                {
                    "appid":6511,
                    "parameterId":6298,
                    "value":"5438"
                }
            ]
        }
    ]
}

```



```

    },
    {
      "appid":6511,
      "parameterId":6299,
      "value":"admin"
    },
    {
      "appid":6511,
      "parameterId":6300,
      "value":"admin"
    },
    {
      "appid":6511,
      "parameterId":6301,
      "value":"cGFzc3dvcmQ="
    },
    {
      "appid":6511,
      "parameterId":6302,
      "value":"password"
    }
  ],
  "applicationChildMembers":[

  ],
  "appName":"postgresql_5432",
  "apptype":"POSTGRESQLInstance",
  "cdsAppId":"696636",
  "cdsClusterId":"145053552328",
  "clustername":"rd10skysp4",
  "critical":false,
  "depth":0,
  "displayAppName":"postgresql_5432(gtc_postgres_nonroot_host)",
  "displayAppType":"POSTGRESQLInstance",
  "exclusionList":"sandy1",
  "executionStatus":"",
  "failoverState":0,
  "friendlyType":"POSTGRESQLInstance",
  "fromMount":false,
  "id":6511,
  "inclusionList":"",
  "infoRequired":false,
  "ip":"",
  "isAsm":false,
  "isClustered":false,
  "iscustomapp":true,
  "isGroup":false,
  "isTbd":true,
  "memberName":"postgresql_5432",
  "memberUuId":"postgresql_5432",
  "onVaultPoolHash":"1917492342,1917492342,1917492342",
  "originalAppId":0,
  "policyOpCode":3,
  "provisioningParameters":"<provisioningoptions><PORT>5438</PORT><OSUSER>admin</OSUSER><BASEDIR>password</BASEDIR></provisioningoptions>",
  "recoveryType":{
    "description":"Select POSTGRESQL applications to recover at the DR site",
    "name":"Recover POSTGRESQL Instance",
    "recoveryTypeId":6170,
    "recoveryTypeOrder":"1"
  }

```

```

    },
    "scriptParameterTOList":[

    ],
    "sequenceNumber":2,
    "sourceHostId":633887,
    "sourceHostName":"gtc_postgres_nonroot_host",
    "targetClusterId":"144690228738",
    "targetHostId":0,
    "targetHostName":"recovered-cent75maxdb1caf1",
    "targetList":"",
    "validForAppGroup":false,
    "vm":false,
    "vmalreadyPresentInRecPlan":false,
    "volumesToList":[

    ]
  }
],
"delay":0,
"memberOrderIndex":2,
"vmMembers":[

]
},
{
  "appMembers":[
    {
      "agmAppId":"721689",
      "appClass":"DB2Group",
      "applicationChildMembers":[

      ],
      "appName":"inst1",
      "apptype":"DB2Instance",
      "cdsAppId":"726455",
      "cdsClusterId":"145053552328",
      "clustername":"rd10skysp4",
      "critical":false,
      "depth":0,
      "displayAppName":"inst1(db2_pass2_root)",
      "displayAppType":"DB2Instance",
      "executionStatus":"",
      "failoverState":0,
      "friendlyType":"DB2Instance",
      "fromMount":false,
      "id":6521,
      "infoRequired":false,
      "ip":"",
      "isAsm":false,
      "isClustered":false,
      "iscustomapp":true,
      "isGroup":false,
      "isTbd":false,
      "memberName":"inst1",
      "memberUuId":"inst1",
      "onVaultPoolHash":"1917492342",
      "originalAppId":0,
      "policyOpCode":3,

```

```

"provisioningParameters": "<provisioningoptions><TARGET_INSTANCE>new_db2_instance</TARGET_INSTANCE></provisioningoptions>",
  "recoveryType": {
    "description": "Select DB2 applications to recover at the DR site",
    "name": "Recover DB2 Instance",
    "recoveryTypeId": 6164,
    "recoveryTypeOrder": "1"
  },
  "sequenceNumber": 3,
  "sourceHostId": 721377,
  "sourceHostName": "db2_pass2_root",
  "targetClusterId": "144690228738",
  "targetHostId": 0,
  "targetHostName": "recovered-cent75maxdb1caf1",
  "validForAppGroup": false,
  "vm": false,
  "vmAlreadyPresentInRecPlan": false
},
"delay": 0,
"memberOrderIndex": 3,
"vmMembers": [
]
}
]
}
]
}

```

Response:

```

{
  "status": "success",
  "statusCode": "1017",
  "statusMessage": "Application group updated successfully. ",
  "responseData": ""
}

```

Remove Application Group

Use this API to delete application group from the Resiliency Director Cloud Recovery.

Method: GET

Request:

https://{HOST}/armnew/api/applicationgroup/remove?ids=6249&_=1618998953811

Response

```

{
  "status": "success",
  "statusCode": "1016",
  "statusMessage": "Application group deleted successfully.",
  "responseData": ""
}

```

Managing Recovery Plans

This interface and its corresponding REST API allows you to add a recovery plan in the Resiliency Director Cloud Recovery. This interface contains the following RESTful API:

[Add Recovery Plan](#)

[Update Recovery Plan](#)

[Fetch Recovery Types](#)
[Remove Recovery Plan](#)

Add Recovery Plan

Use this API to add recovery plan to the Resiliency Director Cloud Recovery.

Method: POST

Request

<https://{HOST}/armnew/api/recoveryplan/add?suppresswarningforresourceaccessibility=true>

Body Param

```
{
  "recoveryPlanMember": {
    "vmprefixName": "recovered-",
    "organizationIds": [],
    "armServerId": null,
    "name": "RPCloud",
    "storageServerId": null,
    "state": "",
    "editedOn": "3456789",
    "editedBy": "",
    "externalNetworkName": "",
    "routerLocation": "",
    "ipAddressPool": "",
    "memoryRequired": "",
    "cpuRequired": "",
    "nicsRequired": "",
    "sourcepooltype": "snapshot",
    "consumedNetworkTOList": [
      {
        "networkid": 57351
      }
    ],
    "consumedCloudTOList": [
      {
        "cloudid": 57349
      }
    ],
    "consumedMigrationTOList": [
      {
        "consumedMigrationValue": "Migrate for Actual Only"
      }
    ]
  },
  "appGroups": [
    {
      "startupSequence": 1,
      "appGroup": {
        "id": 166092,
        "name": "AGCloud",
        "description": "",
        "cdsId": 0,
        "agmId": 160779,
        "cdsHostName": "sourcesky",
        "serverHostName": "",
        "createdOn": "11/15/2019 01:18:14",
        "editedOn": "11/15/2019 01:18:14",
        "noOfVms": 1,

```

```

"noOfNonVms": 0,
"recoveryPlan": "",
"noOfVApps": 0,
"sequenceOrder": [
{
  "memberOrderIndex": 1,
  "delay": 0,
  "vmMembers": [
    {
      "memberUuid": "Win2012_100",
      "memberName": "Win2012_100",
      "vm": true,
      "executionStatus": "",
      "cdsAppId": "556199",
      "agmAppId": "400160",
      "cdsClusterId": "141807825785",
      "targetClusterId": "1415057143",
      "clustername": "rdgcptarget",
      "targetClusterName": "rdgcptarget",
      "validForAppGroup": false,
      "displayAppName": "Win2012_100",
      "displayAppType": "VM",
      "apptype": "VM",
      "sourceHostName": "",
      "recoveryType": {
        "recoveryTypeId": 6,
        "name": "Recover VMs or Physical Servers to Cloud",
        "recoveryTypeOrder": "1"
      },
      "infoRequired": false,
      "policyOpCode": 1,
      "id": 166094,
      "vappName": "",
      "numberOfCpus": 0,
      "actualNumberOfNics": 1,
      "testNumberOfNics": 0,
      "testMemoryInMbs": 4096,
      "actualMemoryInMbs": 4096,
      "shutdownOrder": 0,
      "numberOfVirtualDisks": 0,
      "hddInGbs": 0,
      "hostName": "",
      "ip": "",
      "targetClusterIp": "172.23.16.72",
      "cupResourceAllocation": {
        "reservation": 0,
        "limit": 0,
        "shares": 0,
        "sharesLevel": 0,
        "numberOfCpus": 0,
        "corePerSocket": 0
      },
      "memoryResourceAllocation": {
        "reservation": 0,
        "limit": 0,
        "shares": 0,
        "sharesLevel": 0,
        "numberOfCpus": 0,
        "corePerSocket": 0
      }
    }
  ],
  "id": 166094,
  "vappName": "",
  "numberOfCpus": 0,
  "actualNumberOfNics": 1,
  "testNumberOfNics": 0,
  "testMemoryInMbs": 4096,
  "actualMemoryInMbs": 4096,
  "shutdownOrder": 0,
  "numberOfVirtualDisks": 0,
  "hddInGbs": 0,
  "hostName": "",
  "ip": "",
  "targetClusterIp": "172.23.16.72",
  "cupResourceAllocation": {
    "reservation": 0,
    "limit": 0,
    "shares": 0,
    "sharesLevel": 0,
    "numberOfCpus": 0,
    "corePerSocket": 0
  },
  "memoryResourceAllocation": {
    "reservation": 0,
    "limit": 0,
    "shares": 0,
    "sharesLevel": 0,
    "numberOfCpus": 0,
    "corePerSocket": 0
  }
}
],

```

```

    "testCPUAllocation": 4,
    "actualCPUAllocation": 4,
    "prescript": "",
    "postscript": "",
    "username": "",
    "passwd": "",
    "nicNames": [
      {
        "deviceConfigId": 0,
        "macAddress": "",
        "nicType": "null",
        "interfaceLabel": "Network adapter 1",
        "pciSlotNumber": 0
      }
    ],
    "vmAllNicDataList": [
      {
        "interfaceLabel": "Network adapter 1",
        "enforcedStaticIp": true,
        "enforcedStaticDnsIp": false,
        "ipAddressToConfigure": "",
        "gateway": "",
        "macAddress": "",
        "pciSlotNumber": 0
      }
    ],
    "nicForStaticIp": "",
    "nicForDns": "",
    "defaultDnsIpToconfigure": [],
    "sourceIp": "",
    "sourceSubnet": "",
    "sourceGateway": "",
    "portGroup": [],
    "osName": "windows",
    "consumedNetworkTOList": [],
    "consumedCloudTOList": [],
    "preExternalScript": [],
    "postExternalScript": [],
    "coresPerSocket": 0,
    "testCoresPerSocket": 0,
    "virtualSocket": 0,
    "testVirtualSocket": 0,
    "sequenceNumber": 1,
    "bootdisksize": 40,
    "originalBootdisksize": 0,
    "critical": false,
    "vappStartupOrder": 0,
    "vmAlreadyPresentInRecPlan": false,
    "consumedMigrationTOList": [],
    "vmNewName": "recovered-win2012100"
  }
],
"appMembers": [],
"preExternalScript": [],
"postExternalScript": [],
"consumedCloudTOList": [],
"consumedMigrationTOList": [],
"consumedNetworkTOList": [],
"id": "id_246"
}

```

```

    ],
    "recoveryPlanId": 0,
    "executionStatus": "",
    "peerRecoveryPlanId": 0,
    "cpuRequired": 0,
    "memoryRequired": 0,
    "customerName": "",
    "peerAppGrpId": 166092,
    "preExternalScript": [],
    "postExternalScript": [],
    "organizationToList": [],
    "startupSequence": 0,
    "critical": false,
    "armServerId": null
  }
}
],
"schedule": {
  "scheduleName": "asd",
  "scheduleStatus": "DISABLED",
  "scheduleType": "RECOVERY_PLAN",
  "startTime": "",
  "endTime": "",
  "repeatIntervalType": "",
  "repeatIntervalValue": "",
  "executeOnT0": null,
  "exceptOnT0": null
}
}

```

Body Param (For VM in sequence 1 and CAF database – PostgreSQL in sequence 2 and Db2 in sequence 3)

```

{
  "appGroups": [
    {
      "appGroup": {
        "agmId": 6147,
        "applianceGroupRequired": false,
        "armServerId": null,
        "cdsHostName": "SKY5SP4,SKY7SP4",
        "cdsId": 0,
        "cpuRequired": 0,
        "createdOn": "04/30/2021 08:03:08",
        "critical": false,
        "customerName": "",
        "description": "",
        "editedOn": "04/30/2021 08:03:08",
        "executionStatus": "",
        "id": 6445,
        "memoryRequired": 0,
        "name": "new_ag_VM_DB",
        "noOfNonVms": 2,
        "noOfVApps": 0,
        "noOfVms": 1,
        "organizationToList": [
        ],
        "peerAppGrpId": 6445,
        "peerRecoveryPlanId": 0,
        "postExternalScript": [

```

```

],
"preExternalScript":[

],
"recoveryPlan":"","
"recoveryPlanId":0,
"sequenceOrder":[
{
  "appMembers":[

  ],
  "consumedCloudTOList":[

  ],
  "consumedMigrationTOList":[

  ],
  "consumedNetworkTOList":[

  ],
  "delay":0,
  "id":"id_7",
  "memberOrderIndex":1,
  "postExternalScript":[

  ],
  "preExternalScript":[

  ],
  "vmMembers":[
    {
      "actualCPUAllocation":2,
      "actualMemoryInMbs":8192,
      "actualNumberOfNics":1,
      "agmAppId":"387576",
      "appliancegroupid":0,
      "apptype":"VM",
      "bootdisksize":50,
      "cdsAppId":"9821",
      "cdsClusterId":"143428285568",
      "clustername":"","
      "consumedCloudTOList":[

      ],
      "consumedMigrationTOList":[

      ],
      "consumedNetworkTOList":[

      ],
      "coresPerSocket":0,
      "critical":false,
      "cupResourceAllocation":{
        "corePerSocket":0,
        "limit":0,
        "numberOfCpus":0,
        "reservation":0,
        "shares":0,
        "sharesLevel":0
      }
    }
  ]
}
]

```



```

    },
    "defaultDnsIpToconfigure":[

    ],
    "displayAppName":"Cent7.5_MaxDB1_CAF1",
    "displayAppType":"VM",
    "executionStatus":"",
    "hddInGbs":0,
    "hostName":"",
    "id":6447,
    "infoRequired":false,
    "ip":"",
    "isAsm":false,
    "memberName":"Cent7.5_MaxDB1_CAF1",
    "memberUid":"Cent7.5_MaxDB1_CAF1",
    "memoryResourceAllocation":{
        "corePerSocket":0,
        "limit":0,
        "numberOfCpus":0,
        "reservation":0,
        "shares":0,
        "sharesLevel":0
    },
    "nicForDns":"",
    "nicForStaticIp":"",
    "nicNames":[
        {
            "deviceConfigId":0,
            "interfaceLabel":"Network adapter 1",
            "macAddress":"",
            "nicType":"null",
            "pciSlotNumber":0
        }
    ],
    "numberOfCpus":0,
    "numberOfVirtualDisks":0,
    "onVaultPoolHash":"-171848852,1917492342,-171848852,1917492342,-
171848852,1917492342",
    "originalBootdisksize":0,
    "osName":"linux",
    "passwd":"",
    "policyOpCode":3,
    "portGroup":[

    ],
    "postExternalScript":[

    ],
    "postscript":"",
    "preExternalScript":[

    ],
    "prescript":"",
    "recoveryType":{
        "description":"Select VMware VMs or System State applications
(connector backups) to convert into cloud instances for DR recovery to the cloud. Each source VM
or server may be recovered only once and will retain its original host name.",
        "name":"Recover VMs or Physical Servers to Cloud",
        "recoveryTypeId":6,
        "recoveryTypeOrder":"1"
    }
}

```

```

    },
    "sequenceNumber":1,
    "shutdownOrder":0,
    "sourceClusterName":"SKY5SP4",
    "sourceGateway": "",
    "sourceHostName": "",
    "sourceIp": "",
    "sourceSubnet": "",
    "targetClusterId":"144740187595",
    "targetClusterName":"awssky",
    "testCoresPerSocket":0,
    "testCPUAllocation":2,
    "testMemoryInMbs":8192,
    "testNumberOfNics":0,
    "testVirtualSocket":0,
    "username": "",
    "validForAppGroup":false,
    "vappName": "",
    "vappStartupOrder":0,
    "virtualSocket":0,
    "vm":true,
    "vmAllNicDataList":[
      {
        "enforcedStaticDnsIp":false,
        "enforcedStaticIp":true,
        "gateway": "",
        "interfaceLabel":"Network adapter 1",
        "ipAddressToConfigure": "",
        "macAddress": "",
        "pciSlotNumber":0
      }
    ],
    "vmAlreadyPresentInRecPlan":false,
    "vmNewName":"recovered-cent75maxdb1caf1"
  }
]
},
{
  "appMembers":[
    {
      "agmAppId":"641258",
      "appClass":"POSTGRESQLGroup",
      "appClassMetaDataTOList":[
        {
          "appClassPropertiesTO":{
            "defaultValue": "",
            "description":"Specify PostgreSQL target server port number",
            "isHidden":false,
            "isRequired":true,
            "name":"POSTGRESQL",
            "propertyId":6298,
            "propertyLabel":"PostgreSQL Target Server Port",
            "propertyName":"PORT",
            "propertyType":"STRING"
          },
          "appid":6451,
          "parameterId":6298,
          "value":"5438"
        },
      ],
    },
  ],
}

```

```

    "appClassPropertiesT0":{
      "defaultvalue":"",
      "description":"Specify PostgreSQL target OS user name",
      "isHidden":false,
      "isRequired":true,
      "name":"POSTGRESQL",
      "propertyId":6299,
      "propertyLabel":"PostgreSQL Target OS User Name",
      "propertyName":"OSUSER",
      "propertyType":"STRING"
    },
    "appid":6451,
    "parameterId":6299,
    "value":"admin"
  },
  {
    "appClassPropertiesT0":{
      "defaultvalue":"",
      "description":"Specify PostgreSQL target DB user name",
      "isHidden":false,
      "isRequired":false,
      "name":"POSTGRESQL",
      "propertyId":6300,
      "propertyLabel":"PostgreSQL Target DB User Name",
      "propertyName":"DBUSER",
      "propertyType":"STRING"
    },
    "appid":6451,
    "parameterId":6300
  },
  {
    "appClassPropertiesT0":{
      "defaultvalue":"",
      "description":"Target PostgreSQL Password",
      "isHidden":false,
      "isRequired":false,
      "name":"POSTGRESQL",
      "propertyId":6301,
      "propertyLabel":"PostgreSQL Target DB Password",
      "propertyName":"password",
      "propertyType":"ENCRYPT"
    },
    "appid":6451,
    "parameterId":6301
  },
  {
    "appClassPropertiesT0":{
      "defaultvalue":"",
      "description":"Specify PostgreSQL target server home directory",
      "isHidden":false,
      "isRequired":true,
      "name":"POSTGRESQL",
      "propertyId":6302,
      "propertyLabel":"PostgreSQL Target Server Home Directory",
      "propertyName":"BASEDIR",
      "propertyType":"STRING"
    },
    "appid":6451,
    "parameterId":6302,
    "value":"password"
  }

```

```

    }
  ],
  "applicationChildMembers":[
    ],
    "appName":"postgresql_5432",
    "apptype":"POSTGRESQLInstance",
    "cdsAppId":"696636",
    "cdsClusterId":"145053552328",
    "clustername":"rd10skysp4",
    "critical":false,
    "depth":0,
    "displayAppName":"postgresql_5432(gtc_postgres_nonroot_host)",
    "displayAppType":"POSTGRESQLInstance",
    "exclusionList":"sandy1",
    "executionStatus":"",
    "failoverState":0,
    "friendlyType":"POSTGRESQLInstance",
    "fromMount":false,
    "id":6451,
    "infoRequired":false,
    "ip":"",
    "isAsm":false,
    "isClustered":false,
    "iscustomapp":true,
    "isGroup":false,
    "isTbd":false,
    "memberName":"postgresql_5432",
    "onVaultPoolHash":"1917492342,1917492342,1917492342",
    "originalAppId":0,
    "policyOpCode":3,
    "provisioningParameters":"<provisioningoptions><PORT>5438</
PORT><OSUSER>admin</OSUSER><BASEDIR>password</BASEDIR></provisioningoptions>",
    "recoveryType":{
      "description":"Select POSTGRESQL applications to recover at the DR
site",
      "name":"Recover POSTGRESQL Instance",
      "recoveryTypeId":6170,
      "recoveryTypeOrder":"1"
    },
    "restoreOptions":"",
    "scriptParameters":"",
    "sequenceNumber":2,
    "sourceHostId":0,
    "sourceHostName":"gtc_postgres_nonroot_host",
    "targetClusterId":"144690228738",
    "targetHostId":0,
    "targetHostName":"recovered-cent75maxdb1caf1",
    "targetList":"",
    "validForAppGroup":false,
    "vm":false,
    "vmalreadyPresentInRecPlan":false
  }
],
"delay":0,
"id":"id_8",
"memberOrderIndex":2,
"postExternalScript":[
],

```

```

    "preExternalScript":[

    ],
    "vmMembers":[

    ]
  },
  {
    "appMembers":[
      {
        "agmAppId":"721689",
        "appClass":"DB2Group",
        "appClassMetaDataTOList":[
          {
            "appClassPropertiesT0":{
              "defaultvalue":"",
              "description":"Target Instance Name",
              "isHidden":false,
              "isRequired":true,
              "name":"DB2",
              "propertyId":6418,
              "propertyLabel":"Target Instance Name",
              "propertyName":"TARGET_INSTANCE",
              "propertyType":"EDITABLE_SELECT"
            },
            "appid":6458,
            "parameterId":6418,
            "value":"new_db2_instance"
          },
          {
            "appClassPropertiesT0":{
              "defaultvalue":"",
              "description":"Target Node Number",
              "isHidden":false,
              "isRequired":false,
              "name":"DB2",
              "propertyId":6419,
              "propertyLabel":"Target Node Number",
              "propertyName":"TARGET_NODE_NUM",
              "propertyType":"STRING"
            },
            "appid":6458,
            "parameterId":6419
          }
        ],
        "applicationChildMembers":[

        ],
        "appName":"inst1",
        "apptype":"DB2Instance",
        "cdsAppId":"726455",
        "cdsClusterId":"145053552328",
        "clustername":"rd10skysp4",
        "critical":false,
        "depth":0,
        "displayAppName":"inst1(db2_pass2_root)",
        "displayAppType":"DB2Instance",
        "executionStatus":"",
        "failoverState":0,
        "friendlyType":"DB2Instance",

```

```

        "fromMount":false,
        "id":6458,
        "infoRequired":false,
        "ip":"",
        "isAsm":false,
        "isClustered":false,
        "iscustomapp":true,
        "isGroup":false,
        "isTbd":false,
        "memberName":"inst1",
        "onVaultPoolHash":"1917492342",
        "originalAppId":0,
        "policyOpCode":3,

"provisioningParameters":"<provisioningoptions><TARGET_INSTANCE>new_db2_instance</
TARGET_INSTANCE></provisioningoptions>",
        "recoveryType":{
            "description":"Select DB2 applications to recover at the DR site",
            "name":"Recover DB2 Instance",
            "recoveryTypeId":6164,
            "recoveryTypeOrder":"1"
        },
        "restoreOptions":"",
        "scriptParameters":"",
        "sequenceNumber":3,
        "sourceHostId":0,
        "sourceHostName":"db2_pass2_root",
        "targetClusterId":"144690228738",
        "targetHostId":0,
        "targetHostName":"recovered-cent75maxdb1caf1",
        "targetList":"",
        "validForAppGroup":false,
        "vm":false,
        "vmalreadyPresentInRecPlan":false
    }
    ],
    "delay":0,
    "id":"id_9",
    "memberOrderIndex":3,
    "postExternalScript":[

    ],
    "preExternalScript":[

    ],
    "vmMembers":[

    ]
    },
    "serverHostName":"",
    "startupSequence":0
    },
    "startupSequence":1
    }
    ],
    "recoveryPlanMember":{
        "appliancegroupid":"6154",
        "armServerId":null,
        "consumedCloudTOList":[

```

```

        {
            "cloudid":6159
        }
    ],
    "consumedMigrationTOList":[
        {
            "consumedMigrationValue":"Migrate for Actual Only"
        }
    ],
    "consumedNetworkTOList":[
        {
            "networkid":6161
        }
    ],
    "cpuRequired":"",
    "editedBy":"",
    "editedOn":"3456789",
    "externalNetworkName":"",
    "ipAddressPool":"",
    "memoryRequired":"",
    "name":"New_RP_VM_DB",
    "nicsRequired":"",
    "organizationIds":[

    ],
    "routerLocation":"",
    "sourcepooltype":"snapshot",
    "state":"",
    "storageServerId":null,
    "vmprefixName":"recovered-"
},
"schedule":{
    "endTime":"",
    "exceptOnTO":null,
    "executeOnTO":null,
    "repeatIntervalType":"",
    "resetAfterExecution":false,
    "scheduleName":"asd",
    "scheduleStatus":"DISABLED",
    "scheduleType":"RECOVERY_PLAN",
    "startTime":""
}
}

```

Response

```

{
    "status": "success",
    "statusCode": "1021",
    "statusMessage": "Recovery plan added successfully. ",
    "responseData": []
}

```

Parameters	Description
vmprefixName	Custom prefix name to a recovery plan
name	Name of a Recovery plan

Update Recovery Plan

Use this API to update the recovery plan with VM in sequence 1 and CAF database - PostgreSQL in sequence 2 and Db2 in sequence 3.

Method: POST

Request: <https://{host}/armnew/api/recoveryplan/update?suppresswarningforresourceaccessibility=true>

Body Param:

```
{
  "appGroups": [
    {
      "appGroup": {
        "agmId": 6147,
        "applianceGroupRequired": false,
        "armServerId": null,
        "cdsHostName": "SKY5SP4",
        "cdsId": 0,
        "cpuRequired": 0,
        "createdOn": "04/30/2021 08:03:08",
        "critical": false,
        "customerName": "",
        "description": "",
        "editedOn": "04/30/2021 08:53:38",
        "executionStatus": "",
        "id": 6498,
        "memoryRequired": 0,
        "name": "new_ag_VM_DBs",
        "noOfNonVms": 2,
        "noOfVApps": 0,
        "noOfVms": 1,
        "organizationToList": [

        ],
        "peerAppGrpId": 6498,
        "peerRecoveryPlanId": 0,
        "postExternalScript": [

        ],
        "preExternalScript": [

        ],
        "recoveryPlan": "New_RP_VM_DBs",
        "recoveryPlanId": 6462,
        "sequenceOrder": [
          {
            "appMembers": [

            ],
            "consumedCloudTOList": [

            ],
            "consumedMigrationTOList": [

            ],
            "consumedNetworkTOList": [

            ],
            "delay": 0,

```



```

    "id": "id_23",
    "memberOrderIndex": 1,
    "postExternalScript": [

    ],
    "preExternalScript": [

    ],
    "vmMembers": [
        {
            "actualCPUAllocation": 2,
            "actualMemoryInMbs": 8192,
            "actualNumberOfNics": 1,
            "agmAppId": "387576",
            "appliancegroupid": 0,
            "apptype": "VM",
            "bootdisksize": 50,
            "cdsAppId": "9821",
            "cdsClusterId": "143428285568",
            "clustername": "",
            "consumedCloudTOList": [

            ],
            "consumedMigrationTOList": [

            ],
            "consumedNetworkTOList": [

            ],
            "coresPerSocket": 0,
            "critical": false,
            "cupResourceAllocation": {
                "corePerSocket": 0,
                "limit": 0,
                "numberOfCpus": 0,
                "reservation": 0,
                "shares": 0,
                "sharesLevel": 0
            },
            "defaultDnsIpToconfigure": [

            ],
            "displayAppName": "Cent7.5_MaxDB1_CAF1",
            "displayAppType": "VM",
            "executionStatus": "",
            "hddInGbs": 0,
            "hostName": "",
            "id": 6569,
            "infoRequired": false,
            "ip": "",
            "isAsm": false,
            "memberName": "Cent7.5_MaxDB1_CAF1",
            "memberUuId": "Cent7.5_MaxDB1_CAF1",
            "memoryResourceAllocation": {
                "corePerSocket": 0,
                "limit": 0,
                "numberOfCpus": 0,
                "reservation": 0,
                "shares": 0,
                "sharesLevel": 0
            }
        }
    ]
}

```

```

    },
    "nicForDns": "",
    "nicForStaticIp": "",
    "nicNames": [
        {
            "deviceConfigId": 0,
            "interfaceLabel": "Network adapter 1",
            "macAddress": "",
            "nicType": "null",
            "pciSlotNumber": 0
        }
    ],
    "numberOfCpus": 0,
    "numberOfVirtualDisks": 0,
    "onVaultPoolHash": "-171848852,1917492342,-171848852,1917492342,-
171848852,1917492342",
    "originalBootdisksize": 0,
    "osName": "linux",
    "passwd": "",
    "policyOpCode": 3,
    "portGroup": [

    ],
    "postExternalScript": [

    ],
    "postscript": "",
    "preExternalScript": [

    ],
    "prescript": "",
    "recoveryType": {
        "description": "Select VMware VMs or System State applications
(connector backups) to convert into cloud instances for DR recovery to the cloud. Each source VM
or server may be recovered only once and will retain its original host name.",
        "name": "Recover VMs or Physical Servers to Cloud",
        "recoveryTypeId": 6,
        "recoveryTypeOrder": "1"
    },
    "sequenceNumber": 1,
    "shutdownOrder": 0,
    "sourceClusterName": "SKY5SP4",
    "sourceGateway": "",
    "sourceHostName": "",
    "sourceIp": "",
    "sourceSubnet": "",
    "targetClusterId": "144740187595",
    "targetClusterName": "awssky",
    "testCoresPerSocket": 0,
    "testCPUAllocation": 2,
    "testMemoryInMbs": 8192,
    "testNumberOfNics": 1,
    "testVirtualSocket": 0,
    "username": "",
    "validForAppGroup": false,
    "vappName": "",
    "vappStartupOrder": 0,
    "virtualSocket": 0,
    "vm": true,
    "vmAllNicDataList": [

```

```

        {
            "enforcedStaticDnsIp":false,
            "enforcedStaticIp":true,
            "gateway": "",
            "interfaceLabel": "Network adapter 1",
            "ipAddressToConfigure": "",
            "macAddress": "",
            "pciSlotNumber": 0
        }
    ],
    "vmAlreadyPresentInRecPlan": false,
    "vmNewName": "recovered-cent75maxdb1caf1"
}
]
},
{
    "appMembers": [
        {
            "agmAppId": "641258",
            "appClass": "POSTGRESQLGroup",
            "appClassMetaDataTOList": [
                {
                    "appid": 6577,
                    "parameterId": 6298,
                    "value": "5438"
                },
                {
                    "appid": 6577,
                    "parameterId": 6299,
                    "value": "admin"
                },
                {
                    "appid": 6577,
                    "parameterId": 6300,
                    "value": "admin"
                },
                {
                    "appid": 6577,
                    "parameterId": 6301,
                    "value": "cGFzc3dvcmQ="
                },
                {
                    "appid": 6577,
                    "parameterId": 6302,
                    "value": "password"
                }
            ],
            "applicationChildMembers": [
                ],
            "appName": "postgresql_5432",
            "apptype": "POSTGRESQLInstance",
            "cdsAppId": "696636",
            "cdsClusterId": "145053552328",
            "clustername": "rd10skysp4",
            "critical": false,
            "depth": 0,
            "displayAppName": "postgresql_5432(gtc_postgres_nonroot_host)",
            "displayAppType": "POSTGRESQLInstance",
            "exclusionList": "sandy1",

```

```

        "executionStatus": "",
        "failoverState": 0,
        "friendlyType": "POSTGRESQlInstance",
        "fromMount": false,
        "id": 6577,
        "inclusionList": "",
        "infoRequired": true,
        "ip": "",
        "isAsm": false,
        "isClustered": false,
        "iscustomapp": true,
        "isGroup": false,
        "isTbd": false,
        "memberName": "postgresql_5432",
        "onVaultPoolHash": "1917492342,1917492342,1917492342",
        "originalAppId": 0,
        "policyOpCode": 3,
        "provisioningParameters": "<provisioningoptions><PORT>5438</
PORT><OSUSER>admin</OSUSER><DBUSER>admin</DBUSER><password>password</
password><BASEDIR>password</BASEDIR></provisioningoptions>",
        "recoveryType": {
            "description": "Select POSTGRESQl applications to recover at the DR
site",
            "name": "Recover POSTGRESQl Instance",
            "recoveryTypeId": 6170,
            "recoveryTypeOrder": "1"
        },
        "restoreOptions": "",
        "scriptParameters": "",
        "scriptParameterTOList": [

        ],
        "sequenceNumber": 2,
        "sourceHostId": 0,
        "sourceHostName": "gtc_postgres_nonroot_host",
        "targetClusterId": "144690228738",
        "targetHostId": 0,
        "targetHostName": "recovered-cent75maxdb1caf1",
        "targetList": "",
        "validForAppGroup": false,
        "vm": false,
        "vmalreadyPresentInRecPlan": false,
        "volumesToList": [

        ]
    }
],
    "delay": 0,
    "id": "id_24",
    "memberOrderIndex": 2,
    "postExternalScript": [

    ],
    "preExternalScript": [

    ],
    "vmMembers": [

    ]
},

```

```

{
  "appMembers":[
    {
      "agmAppId":"721689",
      "appClass":"DB2Group",
      "appClassMetaDataTOList":[
        {
          "appClassPropertiesT0":{
            "defaultvalue":"",
            "description":"Target Instance Name",
            "isHidden":false,
            "isRequired":true,
            "name":"DB2",
            "propertyId":6418,
            "propertyLabel":"Target Instance Name",
            "propertyName":"TARGET_INSTANCE",
            "propertyType":"EDITABLE_SELECT"
          },
          "appid":6587,
          "parameterId":6418,
          "value":"new_db2_instance"
        },
        {
          "appClassPropertiesT0":{
            "defaultvalue":"",
            "description":"Target Node Number",
            "isHidden":false,
            "isRequired":false,
            "name":"DB2",
            "propertyId":6419,
            "propertyLabel":"Target Node Number",
            "propertyName":"TARGET_NODE_NUM",
            "propertyType":"STRING"
          },
          "appid":6587,
          "parameterId":6419
        }
      ],
      "applicationChildMembers":[
        ],
      "appName":"inst1",
      "apptype":"DB2Instance",
      "cdsAppId":"726455",
      "cdsClusterId":"145053552328",
      "clustername":"rd10skysp4",
      "critical":false,
      "depth":0,
      "displayAppName":"inst1(db2_pass2_root)",
      "displayAppType":"DB2Instance",
      "executionStatus":"",
      "failoverState":0,
      "friendlyType":"DB2Instance",
      "fromMount":false,
      "id":6587,
      "inclusionList":"AMAZON",
      "infoRequired":false,
      "ip":"",
      "isAsm":false,
      "isClustered":false,

```

```

        "iscustomapp":true,
        "isGroup":false,
        "isTbd":false,
        "memberName":"inst1",
        "onVaultPoolHash":"1917492342",
        "originalAppId":0,
        "policyOpCode":3,

"provisioningParameters":"<provisioningoptions><TARGET_INSTANCE>new_db2_instance</
TARGET_INSTANCE></provisioningoptions>",
        "recoveryType":{
            "description":"Select DB2 applications to recover at the DR site",
            "name":"Recover DB2 Instance",
            "recoveryTypeId":6164,
            "recoveryTypeOrder":"1"
        },
        "sequenceNumber":3,
        "sourceHostId":0,
        "sourceHostName":"db2_pass2_root",
        "targetClusterId":"144690228738",
        "targetHostId":0,
        "targetHostName":"recovered-cent75maxdb1caf1",
        "validForAppGroup":false,
        "vm":false,
        "vmalreadyPresentInRecPlan":false
    }
    ],
    "delay":0,
    "id":"id_25",
    "memberOrderIndex":3,
    "postExternalScript":[

    ],
    "preExternalScript":[

    ],
    "vmMembers":[

    ]
    }
    ],
    "serverHostName":"","
    "startupSequence":1
    },
    "startupSequence":1
    }
    ],
    "recoveryPlanMember":{
        "appliancegroupid":"6154",
        "consumedCloudTOList":[
            {
                "cloudid":6159
            }
        ],
        "consumedMigrationTOList":[
            {
                "consumedMigrationValue":"Migrate for Actual Only"
            }
        ],
        "consumedNetworkTOList":[

```

```

        {
            "networkid":6161
        }
    ],
    "cpuRequired":"","
    "editedBy":"","
    "editedOn":"3456789",
    "externalNetworkName":"","
    "ipAddressPool":"","
    "memoryRequired":"","
    "name":"New_RP_VM_DBs",
    "nicsRequired":"","
    "organizationIds":[

    ],
    "recoveryPlanId":6462,
    "routerLocation":"","
    "sourcepooltype":"snapshot",
    "state":"","
    "storageServerId":null,
    "vmpprefixName":"recovered-"
},
"schedule":{
    "endTime":"","
    "exceptOnTO":null,
    "executeOnTO":null,
    "repeatIntervalType":"","
    "resetAfterExecution":false,
    "scheduleName":"asd",
    "scheduleStatus":"DISABLED",
    "scheduleType":"RECOVERY_PLAN",
    "startTime":""
}
}

```

Response:

```

{
    "status":"success",
    "statusCode":"1000",
    "statusMessage":"Operation performed successfully.",
    "responseData":[]
}

```

Fetch Recovery Types

Use this API to get all the available recovery types.

Method: GET

Request:

https://{HOST}/armnew/api/fetchrecoverytypes?_=1618937512398

Response:

```

{
    "status":"success",
    "statusCode":"1000",
    "statusMessage":"Operation performed successfully.",
    "responseData":[]
    {
        "recoveryPlanName":"RP_CAF6",
    }
}

```

```

    "recoveryPlanId":13097,
    "lastRun":"2021-04-17 14:46:51",
    "organizationList":[

],
    "state":"Ready (info complete)",
    "editedOn":"2021-04-20 13:19:19",
    "editedBy":"",
    "cdsId":0,
    "scheduleId":0,
    "serverid":0,
    "armServerId":0,
    "nextFireTime":"",
    "scheduleStatus":"DISABLED",
    "disabled":false,
    "applicationsCount":"3 in 3 groups",
    "vmprefixName":"",
    "appliancegroupid":11835,
    "sourcepooltype":"onvault",
    "applianceGroupName":"ApplianceGrp2",
    "applianceGroupRequired":false
},
{
    "recoveryPlanName":"RP5_VM5",
    "recoveryPlanId":11712,
    "cdsIP":"rd9skysp4",
    "lastRun":"2021-04-17 04:54:01",
    "organizationList":[

],
    "state":"Success",
    "editedOn":"2021-04-17 02:59:50",
    "editedBy":"",
    "cdsId":0,
    "scheduleId":0,
    "serverid":0,
    "armServerId":0,
    "nextFireTime":"",
    "scheduleStatus":"DISABLED",
    "disabled":false,
    "executionMode":"TEST",
    "applicationsCount":"3 in 1 group",
    "vmprefixName":"rec105",
    "appliancegroupid":11835,
    "sourcepooltype":"onvault",
    "applianceGroupName":"ApplianceGrp2",
    "applianceGroupRequired":false
},
{
    "recoveryPlanName":"RP_DB2_APP",
    "recoveryPlanId":7306,
    "lastRun":"2021-04-07 06:03:45",
    "organizationList":[

],
    "state":"Ready (info complete)",
    "editedOn":"2021-04-07 05:50:19",
    "editedBy":"",
    "cdsId":0,
    "scheduleId":0,

```



```

        "serverid":0,
        "armServerId":0,
        "nextFireTime":"","
        "scheduleStatus":"DISABLED",
        "disabled":false,
        "applicationsCount":"1 in 1 group",
        "vmprefixName":"","
        "appliancegroupid":7216,
        "sourcepooltype":"onvault",
        "applianceGroupName":"APpGrp1",
        "applianceGroupRequired":false
    },
    {
        "recoveryPlanName":"RP_DB2",
        "recoveryPlanId":7198,
        "cdsIP":"rd9skysp4",
        "lastRun":"2021-04-07 05:48:30",
        "organizationList":[

        ],
        "state":"Ready (info complete)",
        "editedOn":"2021-04-07 04:46:57",
        "editedBy":"","
        "cdsId":0,
        "scheduleId":0,
        "serverid":0,
        "armServerId":0,
        "nextFireTime":"","
        "scheduleStatus":"DISABLED",
        "disabled":false,
        "applicationsCount":"1 in 1 group",
        "vmprefixName":"shivaazure-",
        "appliancegroupid":7216,
        "sourcepooltype":"onvault",
        "applianceGroupName":"APpGrp1",
        "applianceGroupRequired":false
    }
]
}

```

Remove Recovery Plan

Use this API to delete the recovery plan(s) managed by Resiliency Director with the parameters.

Method: GET

Request:

https://{host}/armnew/api/recoveryplan/remove?ids=5825&_id=1618980626445

Response:

```

{
    "status":"success",
    "statusCode":"1000",
    "statusMessage":"Operation performed successfully.",
    "responseData":""
}

```

Update Password

Use this API to update password of the Resiliency Director Cloud Recovery with the parameters provided in the request.

Method: POST

Request

https://{HOST}:{PORT}/armnew/api/resetuserpasswd

Body Param

```
{  
  "username": "admin",  
  "oldpassword": "cGFzc3dvcmRuZXc=",  
  "newpassword": "cGFzc3dvcmQ="
```

Appendix A: JavaScript Example

```
var HOST = '172.29.10.195';

var NAME = 'admin';

var PASSWORD = 'password';

var SESSIONID = '';

var actifio = actifio || {};

actifio.version = new function() {
    function getVersion() {
        var config = {
            method : actifio.ajax.GET
        }
        actifio.ajax.fireAjax(actifio.ajax.createUrl('version'),
            versionSuccess, versionError, config);
    }

    function versionSuccess(response) {
        console.log(JSON.stringify(response));
        versionInfoCode = response.statusCode;
        if (response.status == "success") {
            var major = response.responseData.major;
            var minor = response.responseData.minor;
            console.log(major + minor);
        } else {
            console.log('error occured');
        }
    }

    function versionError(response) {
        console.log('error occured');
    }
}
```

```

        this.getVersion = getVersion;

    }

    aktifio.session={
        function login() {

            var username = NAME;
            var password = PASSWORD;
            var data = '{"username":"' + username + '", "password":"' + password
                        + '"}';
            var config = {
                method : aktifio.ajax.POST,
                data : data
            }
            aktifio.ajax.fireAjax(aktifio.ajax.createUrl('login'),
                                loginSuccess, loginError, config);

        }

        function loginSuccess(response) {
            console.log(JSON.stringify(response));
            loginInfoCode = response.statusCode;
            if (response.status == "success") {
                var username = response.responseData.username;
                SESSIONID = response.responseData.sessionid;
                console.log(username + sSESSIONID);

            } else {
                console.log('error occured');
            }
        }
    }

```

```

    }

    function loginError(response) {
        console.log('error occurred');
    }

    function logout() {
        var config = {
            method : actifio.ajax.GET
        }

        actifio.ajax.fireAjax(actifio.ajax.createUrl('logout', SESSIONID),
                               logoutSuccess, logoutError, config);
    }

    function logoutSuccess(response) {
        console.log(JSON.stringify(response));
        logoutInfoCode = response.statusCode;
        if (response.status == "success") {

            console.log('Logout successfull');

        } else {
            console.log('error occurred');
        }
    }

    function logoutError(response) {
        console.log('error occurred');
    }

    this.logout = logout;

    this.login = login;
};

```

```

activo.ajax = new function() {
    function fireAjax(url, successFn, errorFn, config) {
        config = config || {};
        var ajaxConfig = {
            type : config.method || 'GET',
            dataType : 'json',
            contentType : "application/json; charset=UTF-8",
            url : url,
            data : config.data || {},
            cache : config.cache || false,
            async : config.async || true,
            success : successFn,
            error : errorFn
        };
        $.ajax(ajaxConfig);
    }
    function createUrl(request, query) {
        var url = 'https://' + HOST + '/armnew/api/' + request;
        if (query) {
            url += '?' + query.join('&');
        }
        return url;
    }
    this.fireAjax = fireAjax;
    this.createUrl = createUrl;
    this.GET = 'GET';
    this.POST = 'POST';
}

```

Appendix B: Java Example

```
public class RestApi {  
    public static final String HOST = "172.29.10.195";  
    public static final String NAME = "admin";  
    public static final String PASSWORD = "password";  
    public RestApi() {  
        try {  
            // Create a trust manager that does not validate certificate chains  
            TrustManager[] trustAllCerts = new TrustManager[] { new  
X509TrustManager() {  
                public java.security.cert.X509Certificate[] getAcceptedIssuers()  
{  
                    return null;  
                }  
                public void checkClientTrusted(X509Certificate[] certs,  
                    String authType) {  
                }  
                public void checkServerTrusted(X509Certificate[] certs,  
                    String authType) {  
                }  
            } };  
            // Install the all-trusting trust manager  
            SSLContext sc = SSLContext.getInstance("SSL");  
            sc.init(null, trustAllCerts, new java.security.SecureRandom());  
            HttpURLConnection  
                .setDefaultSSLSocketFactory(sc.getSocketFactory());  
            // Create all-trusting host name verifier  
            HostnameVerifier allHostsValid = new HostnameVerifier() {  
                public boolean verify(String hostname, SSLSession session) {  
                    return true;  
                }  
            };  
        }  
    }  
};
```

```

        // Install the all-trusting host verifier
        HttpURLConnection.setDefaultHostnameVerifier(allHostsValid);
    } catch (NoSuchAlgorithmException e) {
        e.printStackTrace();
    } catch (KeyManagementException e) {
        e.printStackTrace();
    }
}

public static void main(String[] args) throws JSONException {
    RestApi api = new RestApi();
    String sessionid = api.login(NAME, PASSWORD);
    api.version();
    api.logout(sessionid);
}

// Function to get version
public void version() {
    String params = "";
    String json = apiRequest("GET", "version", params);
    System.out.println(json);
}

// Function for Login
public String login(String name, String password) throws JSONException {
    JSONObject obj = new JSONObject();
    try {
        obj.put("username", name);
        obj.put("password", password);
    } catch (JSONException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

    String json = apiRequest("POST", "login", obj.toString());
    JSONObject jobject = new JSONObject(json); // json

```



```

JSONObject data = jobject.getJSONObject("responseData");

String sessionid = data.getString("sessionid");
System.out.println("session id is: " + sessionid);
System.out.println(json);
return sessionid;
}

// Function for Logout
public void logout(String sessionid) {
    StringBuilder urlbuilder = new StringBuilder();
    urlbuilder.append("?");
    urlbuilder.append("sessionid=" + sessionid);
    urlbuilder.append("&");
    // Get the json response
    String json = apiRequest("GET", "logout", urlbuilder.toString());
    System.out.println(json);
}

// Generate the URL from the parameters and return the response
private String apiRequest(String method, String apiCall, String param) {
    BufferedReader bufferedReader = null;
    String response = null;
    try {
        StringBuilder urlBuilder = new StringBuilder("https://" + HOST
            + "/armnew/api/" + apiCall);
        if (method == "GET") {
            urlBuilder.append(param);
        }
        // Open connection
        URL url = new URL(urlBuilder.toString());

        HttpURLConnection connection = (HttpURLConnection) url
            .openConnection();

```

```

// Set GET or POST
connection.setRequestMethod(method);

connection.setRequestProperty("Content-Type", "application/json");

if (method == "POST") {
    connection.setDoOutput(true);

    OutputStreamWriter writer = new OutputStreamWriter(
        connection.getOutputStream());

    writer.write(param);
    writer.flush();
}

// Make request
connection.connect();

// Read json from proper stream
if (connection.getResponseCode() == HttpURLConnection.HTTP_OK) {
    bufferedReader = new BufferedReader(new InputStreamReader(
        connection.getInputStream()));
} else {
    bufferedReader = new BufferedReader(new InputStreamReader(
        connection.getErrorStream()));
}

response = bufferedReader.readLine();
} catch (MalformedURLException e) {
    e.printStackTrace();
} catch (IOException e) {
    e.printStackTrace();
} finally {
    // Close BufferedReader
    try {
        if (bufferedReader != null) {
            bufferedReader.close();
        }
    } catch (IOException e) {

```

```
        e.printStackTrace();
    }
}
// Return the json
return response;
}
}
```

