GET/applications/{appInstanceId}/subscriptions/{subscriptionType}/{subscriptionId}

|  |  |
| --- | --- |
| Description | If none of the parameters are passed |
| **Identifier** | MEC-MP1-SUB-INSID-001 |
| **Test Purpose** | Check response received from server when all parameter is null |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{appInstanceId}/subscriptions/{subscriptionType}  /{subscriptionId} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement without specifying any parameters |
|  | |
| **Expected Response code and Message** | 400 Bad Request |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – APPINSTANCEID – 200

|  |  |
| --- | --- |
| Description | If valid appInstanceId is passed |
| **Identifier** | MEC-MP1-SUB-INSID-002 |
| **Test Purpose** | Check response received from server when appInstanceId is valid |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{***valid\_appInstanceId***}/subscriptions  /{subscriptionType}/{subscriptionId} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying valid appInstanceId |
|  | |
| **Expected Response code and Message** | 200 Success |
| **Expected Response Example Value** | {  "SerAvailabilityNotificationSubscription": {  "subscriptionType": "SerAvailabilityNotificationSubscription",  "callbackReference": "string",  "\_links": {  "self": "/meMp1/example"  },  "filteringCriteria": {  "serInstanceId": "ServiceInstance123",  "serName": "ExampleService",  "serCategory": {  "href": "catItem1",  "id": "id12345",  "name": "RNI",  "version": "version1"  },  "version": "ServiceVersion1",  "state": "ACTIVE",  "transportId": "Rest1",  "transportInfo": {  "id": "TransId12345",  "name": "REST",  "description": "REST API",  "type": "JSON",  "protocol": "HTTP",  "version": "2.0",  "endpoint": {  "uris": [  "/meMp1/service/EntryPoint"  ],  "addresses": [  {  "host": "192.0.2.0",  "port": "8080"  }  ],  "alternative": {}  },  "security": {  "oAuth2Info": {  "grantTypes": "OAUTH2\_CLIENT\_CREDENTIALS",  "tokenEndpoint": "/meMp1/security/TokenEndPoint"  }  },  "implSpecificInfo": {}  },  "serializer": "JSON"  }  }  } |

PARAMETER – APPINSTANCEID – 400

|  |  |
| --- | --- |
| Description | If appInstanceId is not passed |
| **Identifier** | MEC-MP1-SUB-INSID-003 |
| **Test Purpose** | Check response received from server when appInstanceId is null |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{ }/subscriptions/{subscriptionType}  /{subscriptionId} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement without specifying appInstanceId |
|  | |
| **Expected Response code and Message** | 400 Bad Request |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – APPINSTANCEID – 404

|  |  |
| --- | --- |
| Description | If appInstanceId is invalid |
| **Identifier** | MEC-MP1-SUB-INSID-004 |
| **Test Purpose** | Check response received from server when appInstanceId is incorrect |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{***Invalid\_appInstanceId***}/subscriptions  /{subscriptionType}/{subscriptionId} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying invalid appInstanceId |
|  | |
| **Expected Response code and Message** | 404 Not Found |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SUBSCRIPTIONTYPE – 200

|  |  |
| --- | --- |
| Description | If valid subscriptionType is passed |
| **Identifier** | MEC-MP1-SUB-INSID-005 |
| **Test Purpose** | Check response received from server when subscriptionType is valid |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{appInstanceId}/subscriptions  /{***Valid\_subscriptionType***}/{subscriptionId} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying valid subscriptionType |
|  | |
| **Expected Response code and Message** | 200 Success |
| **Expected Response Example Value** | {  "SerAvailabilityNotificationSubscription": {  "subscriptionType": "SerAvailabilityNotificationSubscription",  "callbackReference": "string",  "\_links": {  "self": "/meMp1/example"  },  "filteringCriteria": {  "serInstanceId": "ServiceInstance123",  "serName": "ExampleService",  "serCategory": {  "href": "catItem1",  "id": "id12345",  "name": "RNI",  "version": "version1"  },  "version": "ServiceVersion1",  "state": "ACTIVE",  "transportId": "Rest1",  "transportInfo": {  "id": "TransId12345",  "name": "REST",  "description": "REST API",  "type": "JSON",  "protocol": "HTTP",  "version": "2.0",  "endpoint": {  "uris": [  "/meMp1/service/EntryPoint"  ],  "addresses": [  {  "host": "192.0.2.0",  "port": "8080"  }  ],  "alternative": {}  },  "security": {  "oAuth2Info": {  "grantTypes": "OAUTH2\_CLIENT\_CREDENTIALS",  "tokenEndpoint": "/meMp1/security/TokenEndPoint"  }  },  "implSpecificInfo": {}  },  "serializer": "JSON"  }  }  } |

PARAMETER – SUBSCRIPTIONTYPE – 400

|  |  |
| --- | --- |
| Description | If subscriptionType is not passed |
| **Identifier** | MEC-MP1-SUB-INSID-006 |
| **Test Purpose** | Check response received from server when subscriptionType is null |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{appInstanceId}/subscriptions/{subscriptionType}  /{subscriptionId} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement without specifying subscriptionType |
|  | |
| **Expected Response code and Message** | 400 Bad Request |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SUBSCRIPTIONTYPE – 404

|  |  |
| --- | --- |
| Description | If subscriptionType is invalid |
| **Identifier** | MEC-MP1-SUB-INSID-007 |
| **Test Purpose** | Check response received from server when subscriptionType is incorrect |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{appInstanceId}/subscriptions  /{***invalid\_subscriptionType***}/{subscriptionId} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying invalid subscriptionType |
|  | |
| **Expected Response code and Message** | 404 Not Found |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SUBSCRIPTIONID – 200

|  |  |
| --- | --- |
| Description | If valid subscriptionId is passed |
| **Identifier** | MEC-MP1-SUB-INSID-008 |
| **Test Purpose** | Check response received from server when subscriptionId is valid |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{appInstanceId}/subscriptions  /{subscriptionType}/{***valid\_subscriptionId***} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying valid subscriptionId |
|  | |
| **Expected Response code and Message** | 200 Success |
| **Expected Response Example Value** | {  "SerAvailabilityNotificationSubscription": {  "subscriptionType": "SerAvailabilityNotificationSubscription",  "callbackReference": "string",  "\_links": {  "self": "/meMp1/example"  },  "filteringCriteria": {  "serInstanceId": "ServiceInstance123",  "serName": "ExampleService",  "serCategory": {  "href": "catItem1",  "id": "id12345",  "name": "RNI",  "version": "version1"  },  "version": "ServiceVersion1",  "state": "ACTIVE",  "transportId": "Rest1",  "transportInfo": {  "id": "TransId12345",  "name": "REST",  "description": "REST API",  "type": "JSON",  "protocol": "HTTP",  "version": "2.0",  "endpoint": {  "uris": [  "/meMp1/service/EntryPoint"  ],  "addresses": [  {  "host": "192.0.2.0",  "port": "8080"  }  ],  "alternative": {}  },  "security": {  "oAuth2Info": {  "grantTypes": "OAUTH2\_CLIENT\_CREDENTIALS",  "tokenEndpoint": "/meMp1/security/TokenEndPoint"  }  },  "implSpecificInfo": {}  },  "serializer": "JSON"  }  }  } |

PARAMETER – SUBSCRIPTIONID – 400

|  |  |
| --- | --- |
| Description | If subscriptionId is not passed |
| **Identifier** | MEC-MP1-SUB-INSID-009 |
| **Test Purpose** | Check response received from server when subscriptionId is null |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{appInstanceId}/subscriptions/{subscriptionType}  /{ } |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement without specifying subscriptionId |
|  | |
| **Expected Response code and Message** | 400 Bad Request |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |

PARAMETER – SUBSCRIPTIONID – 404

|  |  |
| --- | --- |
| Description | If subscriptionId is invalid |
| **Identifier** | MEC-MP1-SUB-INSID-010 |
| **Test Purpose** | Check response received from server when subscriptionId is incorrect |
| **Configuration** |  |
| **References** | https://forge.etsi.org/rep/gitweb.cgi/MEC.GS\_011.git/tree |
| **Request** | /mp1/v1/applications/{appInstanceId}/subscriptions  /{subscriptionType}/{***Invalid\_subscriptionId***} |
| Applicability |  |
|  | |
| **Pre-test Conditions** | MEC installed and working |
|  | |
| **Test Sequence** | Request list of all enablement by specifying invalid subscriptionId |
|  | |
| **Expected Response code and Message** | 404 Not Found |
| **Expected Response Example Value** | {  "ProblemDetails": {  "type": "string",  "title": "string",  "status": 0,  "detail": "string",  "instance": "string"  }  } |