Alarms and Notifications for ETSI LCM Operations

ETSI Alarms

ESC provides alarms and notifications to the NFVO. The NFVO has to subscribe to these alarms and notifications and send requests to ESC.

The NFVO can receive information about the alarms in the following ways:

Query All Alarms

The NFVO can get a list of all the alarms from the alarms resource.

Method Type:

```
GET
```

VNFM Endpoint:

```
/vnffm/v1/alarms
```

HTTP Request Headers:

```
Accept:application/json
```

For example, to query all alarms with the event type as ENVIRONMENTAL_ALARM

Method Type:

```
GET
```

VNFM Endpoint:

```
http://localhost:8250/vnffm/v1/alarms?eventType="ENVIRONMENTAL_ALARM"
```

HTTP Request Headers:

```
Accept:application/json
```

While querying for multiple alarms, the NFVO can use the URI query parameters to filter the results. The following attribute names are supported for the URI query of the alarms:

- id
- managedObjectId
- rootCauseFaultyResource.faultyResourceType
- eventType
- perceivedSeverity
- probableCause

Note The URI query parameters are for querying multiple alarms only.

Query an Individual Alarm

The NFVO can query a particular alarm from the *alarmId* resource.

Method Type:

```
GET
```

VNFM Endpoint

```
/vnffm/v1/alarms/{alarmId}
```

HTTP Request Header:

```
Accept:application/json
```

Modify an Individual Alarm

To modify an alarm, the NFVO must send a PATCH request to the *AlarmModifications* resource.

Method Type:

```
PATCH
```

VNFM Endpoint:

```
/vnffm/v1/alarms/{alarmId}
```

HTTP Request Header:

```
Content-Type:application/json
```

The supported attribute is ackState, and the supported attribute value is ACKNOWLEDGE. All other modification payloads are rejected.

Alarm Extensions

ETSI provides an extension for the alarms to interact with the third party tools. You must send a POST request to create the alarms.

Method Type

```
POST
```

VNFM Endpoint

```
/vnffm/v1/extension/alarms
```

HTTP Request Header

```
Content-Type:application/json
```

Request Payload

```
[admin@davwebst-esc-4-2-0-49-keep ETSI]$ cat CreateAlarm.json
{
    "id": "alm87032",
    "externalAlarmId": "ext-id-xx11214",
    "managedObjectId": "930fb087-c1b9-4660-bec8-2a8d97dc1df5",
    "rootCauseFaultyResource": {
        "id": "fres7629",
        "faultyResource": {
            "resourceId": "res7727"
        },
        "faultyResourceType": "NETWORK"
    },
    "alarmRaisedTime": "2018-05-30T13:55:15.645000+00",
    "ackState": "UNACKNOWLEDGED",
    "perceivedSeverity": "MAJOR",
    "eventTime": "2018-05-30T13:55:15.645000+00",
    "eventType": "ENVIRONMENTAL_ALARM",
    "probableCause": "Server room overheading",
    "isRootCause": "false"
}
```

Subscribing to Notifications

The NFVO can subscribe to the ETSI notifications related to fault management from ESC.

Create a Subscription

The NFVO sends a POST request to subscribe to the notifications.

Method Type

POST

VNFM Endpoint

```
/vnffm/v1/subscriptions
```

Response Payload

```
{
  "filter" : {
    "notificationTypes" : [
      "AlarmNotification",
      "AlarmClearedNotification",
      "AlarmListRebuiltNotification"
    ],
    "perceivedSeverities" : [
      "CRITICAL",
      "MAJOR"
    ]
  },
  "callbackUri" : "https://nfvo.endpoint.listener",
  "authentication" : {
    "authType" : "BASIC",
    "paramsBasic" : {
      "userName" : "admin",
      "password" : "pass123"
    }
  }
}
```

This creates a new subscription resource and a new identifier. The callbackUri is the only mandatory parameter. The others are all optional. You can verify if the callbackuri is valid and reachable by sending a GET request.

Query all Subscriptions

The NFVO can query information about its subscriptions by sending a GET request to the *subscriptions* resource.

Method Type

```
GET
```

VNFM Endpoint

```
/vnffm/v1/subscriptions
```

HTTP Request Headers

```
Accept:application/json
```

For example, to query all alert subscriptions, when the callbackUri is

```
http://10.10.1.44:9202/alerts/subscriptions/callback
```

```
GET
```

VNFM Endpoint

```
http://localhost:8250/vnffm/v1/subscriptions?callbackUri="http://10.10.1.44:9202/alerts/subscriptions/callback"
```

HTTP Request Headers

```
Accept:application/json
```

The NFVO can use the URI query parameters to filter the results. The following attribute names are supported for the URI query of the subscriptions:

- id
- filter
- callbackUri

Note The URI query parameters are for querying multiple subscriptions only.

Query an Individual Subscription

You must know the subscription ID to query an individual subscription.

Method Type

```
GET
```

VNFM Endpoint

```
/vnffm/v1/subscriptions/{subscriptionId}
```

HTTP Request Header

```
Accept:application/json
```

Delete a Subscription

You can delete a subscription if the NFVO does not need it. Send a delete request to the individual subscription.

Method Type

```
DELETE
```

VNFM Endpoint

```
/vnffm/v1/subscriptions/{subscriptionId}
```

HTTP Request Header

```
http://localhost:8250/vnffm/v1/subscriptions/682791f8-34ad-487e-811a-553036bf49b2
```