MX7000 Advance Filtering
Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 2019</td>
<td>Initial release</td>
</tr>
</tbody>
</table>

Acknowledgements

This paper was produced by the following members of the Dell EMC storage engineering team:

Author: Dahir Herzi and Gilberto Osuna Gonzalez
# Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledgements</td>
<td>2</td>
</tr>
<tr>
<td>Introduction</td>
<td>4</td>
</tr>
<tr>
<td>Audit Logs</td>
<td>4</td>
</tr>
<tr>
<td>Navigate to Audit Logs</td>
<td>4</td>
</tr>
<tr>
<td>Audit Log Page Sections</td>
<td>4</td>
</tr>
<tr>
<td>Audit Log Filters</td>
<td>5</td>
</tr>
<tr>
<td>Audit Log Export</td>
<td>8</td>
</tr>
<tr>
<td>Alert Logs</td>
<td>9</td>
</tr>
<tr>
<td>Navigate to Alert Logs</td>
<td>10</td>
</tr>
<tr>
<td>Alert Log Page Sections</td>
<td>10</td>
</tr>
<tr>
<td>Alert Log Filters</td>
<td>11</td>
</tr>
<tr>
<td>Appendix</td>
<td>14</td>
</tr>
<tr>
<td>Appendix I. Using REST to apply filters to Audit Logs</td>
<td>14</td>
</tr>
<tr>
<td>Audit Logs Filters (REST)</td>
<td>14</td>
</tr>
<tr>
<td>Get all Audit Logs</td>
<td>16</td>
</tr>
<tr>
<td>Get a Single Audit Log</td>
<td>17</td>
</tr>
<tr>
<td>Appendix II. Using REST to apply filters to Alert Logs</td>
<td>18</td>
</tr>
<tr>
<td>Alert Logs Filters (REST)</td>
<td>18</td>
</tr>
<tr>
<td>Get all Alert Logs</td>
<td>19</td>
</tr>
<tr>
<td>Get a Single Alert Log</td>
<td>23</td>
</tr>
</tbody>
</table>
Introduction

The purpose of this whitepaper is to describe Audit logs and Alert logs in MX7000 chassis. Audit logging presents information about the operations/actions that have been invoked in the MX7000 environment. For audit logs it displays details in a categorized manner and informs about the time in which an action took place. Also it provides details on the user that invoked the operation, the source of the request, the message id and a brief description explaining the operation or action that was performed.

As per Alert logs, it provides information about notification events generated by devices or internal chassis components. The type of events can be SNMP Traps, REDFISH events or internal chassis events. The alert details surfaced are extracted from the incoming events and are rendered on the alerts page. The alert information contains valuable details about the nature of the issue, the severity of the event, and in most cases a recommended action to perform that could resolve the issue reported by a device. In addition, the events are categorized, also events provide other details such as reception time, source or originating device information, a message id and a detailed descriptive message.

Both alerts and audits can be filtered to find relevant information. This helps the user locate specific alerts or audit logs in an efficient manner. Locating an audit entry or an alert by filtering criteria helps reduce time spent on the audit logs or the alert pages.

More information will be provided to explain how we can filter entries of the audit logs or alerts and how the different filtering criteria can be utilized.

Audit Logs

The primary objective of this section is:

- Locating audit logs on the MX7000 chassis.
- The combination of filters that can be utilized.

Navigate to Audit Logs

Audit logs can be found in the following path: Monitor -> Audit Logs of the MX7000 UI.

Audit Log Page Sections

All the existing audit log entries will be rendered in that page. By default, all audit log records are not filtered and shown sorted based on time stamp. The advanced filters option will be at top along with export as shown...
in figure 2. Expanding the advanced filters will show all available filtering criteria for Audit Logs. There is also an option named ‘Clear All Filters’ which will clear out any applied filters and show all audit log records without filtering.

![Audit Log Filters](image)

**Figure 2** Audit Logs Top Section

At the bottom of this page there’s a navigation bar and details about the amount of records and how many records are shown per page (by default it is 30 records per page). In the navigation it will show the current page and the total number of pages (1 of N). The total number of pages is determined by the total number of records that can be grouped in counts of 30 records per page.

Arrows indicates the navigation for last ( ), next ( ), ( ) previous and first ( ) pages. In addition, the page number can be typed to navigate to it.

![Audit Logs Bottom Section (Navigation Bar)](image)

**Figure 3** – Audit Logs Bottom Section (Navigation Bar)

### Audit Log Filters

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity</td>
<td>Severity to filter from the list of audit logs. Values can be:</td>
</tr>
<tr>
<td></td>
<td>• All</td>
</tr>
<tr>
<td></td>
<td>• Info</td>
</tr>
<tr>
<td></td>
<td>• Warning</td>
</tr>
<tr>
<td></td>
<td>• Critical</td>
</tr>
<tr>
<td>Start Time</td>
<td>Start time to be applied to filter audit log. This filter <strong>MUST</strong> be</td>
</tr>
<tr>
<td></td>
<td>combined with End Time to filter data in a date/time range manner.</td>
</tr>
<tr>
<td>End Time</td>
<td>End time to be applied to filter audit log. This filter <strong>MUST</strong> be</td>
</tr>
<tr>
<td></td>
<td>combined with Start Time to filter data in a date/time range manner.</td>
</tr>
<tr>
<td>User</td>
<td>User, name of the user that generate an audit log.</td>
</tr>
<tr>
<td>Source Address</td>
<td>IP address of the source attached to the creation of the audit log record.</td>
</tr>
<tr>
<td>Category</td>
<td>List of available categories to filter. Values can be:</td>
</tr>
<tr>
<td></td>
<td>• All</td>
</tr>
<tr>
<td></td>
<td>• Audit</td>
</tr>
<tr>
<td></td>
<td>• Configuration</td>
</tr>
</tbody>
</table>
The following shows an example of how all filters can be combined to obtain a more specific list of audit logs.

<table>
<thead>
<tr>
<th>Description Contains</th>
<th>Filter applied to description. This is more like a LIKE and will audit logs based in this condition.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message ID</td>
<td>Message ID to be filter out from audit logs.</td>
</tr>
</tbody>
</table>

Table 1  Audit Logs Filters

Other examples of filters are as follows:

Figure 4  Audit Logs Filters (All filters Applied)

Figure 5  Audit Logs Filters (Filter by Severity)
Figure 6  Audit Logs Filters (Filter by Start and End Time)

<table>
<thead>
<tr>
<th>Severity</th>
<th>Start Time</th>
<th>End Time</th>
<th>User</th>
<th>Source Address</th>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Aug 3, 2018 12:26 PM</td>
<td></td>
<td>root</td>
<td>C02R00003</td>
<td>10.122.227.46</td>
<td>Audit</td>
</tr>
<tr>
<td></td>
<td>Aug 3, 2018 11:55 PM</td>
<td></td>
<td>root</td>
<td>C0300011</td>
<td>10.122.227.46</td>
<td>Configuration</td>
</tr>
<tr>
<td></td>
<td>Aug 3, 2018 9:05 PM</td>
<td></td>
<td>root</td>
<td>C02R00001</td>
<td>10.122.227.46</td>
<td>Audit</td>
</tr>
<tr>
<td></td>
<td>Aug 3, 2018 9:58:54 AM</td>
<td></td>
<td>root</td>
<td>C02R00001</td>
<td>10.122.227.46</td>
<td>Audit</td>
</tr>
<tr>
<td></td>
<td>Aug 3, 2018 4:06:58 PM</td>
<td></td>
<td>root</td>
<td>C030011</td>
<td>10:30:171.246</td>
<td>Configuration</td>
</tr>
<tr>
<td></td>
<td>Aug 3, 2018 4:02:28 PM</td>
<td></td>
<td>root</td>
<td>C0300100</td>
<td>10:30:171.246</td>
<td>Configuration</td>
</tr>
<tr>
<td></td>
<td>Aug 3, 2018 3:52:48 PM</td>
<td></td>
<td>root</td>
<td>C0300109</td>
<td>10:30:171.246</td>
<td>Configuration</td>
</tr>
</tbody>
</table>

Figure 7  Audit Logs Filters (Filter by User)

<table>
<thead>
<tr>
<th>Severity</th>
<th>Start Time</th>
<th>End Time</th>
<th>User</th>
<th>Source Address</th>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Aug 17, 2018 10:25:02 AM</td>
<td></td>
<td>root</td>
<td>C01R00001</td>
<td>10.250.127.221</td>
<td>Audit</td>
</tr>
<tr>
<td></td>
<td>Aug 17, 2018 9:25:44 AM</td>
<td></td>
<td>root</td>
<td>C01R00001</td>
<td>10.250.125.226</td>
<td>Audit</td>
</tr>
<tr>
<td></td>
<td>Aug 16, 2018 9:49:29 PM</td>
<td></td>
<td>root</td>
<td>C01R00003</td>
<td>10.30:181.50</td>
<td>Audit</td>
</tr>
<tr>
<td></td>
<td>Aug 18, 2018 4:06:02 PM</td>
<td></td>
<td>root</td>
<td>C030011</td>
<td>10:30:181.50</td>
<td>Configuration</td>
</tr>
<tr>
<td></td>
<td>Aug 10, 2018 4:41:17 PM</td>
<td></td>
<td>root</td>
<td>C0300109</td>
<td>10:30:181.00</td>
<td>Audit</td>
</tr>
<tr>
<td></td>
<td>Aug 10, 2018 4:32:02 AM</td>
<td></td>
<td>root</td>
<td>C0300109</td>
<td>10:30:181.00</td>
<td>Audit</td>
</tr>
</tbody>
</table>

Figure 8  Audit Logs Filters (Filter by Source Address)
Audit Log Export

Audit Logs entries can be export to a CSV file by clicking the Export button located in the top of the Audit Log Page, a sub menu will be shown and from there we select Export All.
Alert Logs

Export all will show a ‘Save As’ window and from there we can navigate to the location where we want to save our current Audit Logs.

The generated audit log CSV file can be opened by using Excel or a text editor. It will show the details as indicated in below Figure 14.

Alert Logs

The following section describes:

- The alert logs section of MX7000.
- Locating the alert logs page. The combination of filters that can be utilized.
Alert log entries are the events generated by devices such as SLEDs, IOMs, chassis controller or internal events by MX7000 chassis. These events are recorded and presented in the UI through the Alert Log page. These events are generated, received and processed. The source of the event is a device (except for internal events) and the type can be SNMP or REDFISH events (in case of EC).

**Navigate to Alert Logs**

Alert logs can be found in the following path: Alerts -> Alert Log of the MX7000 UI.

![Alert Log Page](image)

**Alert Log Page Sections**

All alerts will be available on above mentioned page. By default the alerts are not filtered and they are rendered in a sorted order based on the time stamp. The advanced filters option will be at top along with export. Expanding the advanced filters will show all available filters for Alerts.

![Advanced Filters](image)

At the bottom of this page there’s a navigation bar and details about the amount of records and how many records are shown per page (by default it is 30 records per page). In the navigation it will show the current page and the total number of pages (1 of N). The total number of pages are determined by the total number of records that can be grouped in counts of 30 records per page.

![Navigation Bar](image)
At the right side of the page there’s an information block that shows relevant information from the event generated by the device. Like the domain corresponding to the event, detailed message of the event, recommended action to resolve the issue, and Message ID (EEMI Message ID).

Audit Log Filters

At the right side of the page there’s an information block that shows relevant information from the event generated by the device. Like the domain corresponding to the event, detailed message of the event, recommended action to resolve the issue, and Message ID (EEMI Message ID).

Alert Log Filters

Filter Description

Severity Severity to filter from the list of alert logs. Values can be:
- All
- Unknown
- Info
- Normal
- Warning
- Critical

Acknowledge Indicate if the alert is acknowledged or unacknowledged.

Start Date Start Date to be applied to filter audit log. This filter MUST be combined with End Date to filter data in a date/time range manner.

End Date End Date to be applied to filter audit log. This filter MUST be combined with Start Date to filter data in a date/time range manner.

Source Name Source name or identifier of the device that generates the alert.

Category To indicate which category, we need to filter out. This is combined with sub category. If we require a specific subcategory. Values can be:
- All
- Audit
- Configuration
- Miscellaneous
- Storage
- System Health
- Updates
Alert Log Filters

<table>
<thead>
<tr>
<th>Subcategory</th>
<th>Indicates which subcategories of the selected category can be filter. This list is variable, and it can contain 1 to N subcategories per selected category. Is recommended to select a subcategory to filter more accurately.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message</td>
<td>A message that can be filter out from the list of alerts. This will use a LIKE to filter out alerts.</td>
</tr>
</tbody>
</table>

Table 2  Alert Log Filters

following filters that can be applied to the existing view of recorded logs. Also, all of them can be combined to be stricter in the filter.

![Alert Log Filters (All filters Applied)](image1)

Other examples of filters are as follows:

![Alert Log Filters (Filter by Severity)](image2)
### Alert Log Filters

#### Alert Log Filters (Filter by Acknowledge)

<table>
<thead>
<tr>
<th>Severity</th>
<th>Acknowledge</th>
<th>Start Date</th>
<th>End Date</th>
<th>Source Name</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>8/1/2018</td>
<td>8/17/2018</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Alert Log Filters (Filter by Start Date and End Date)

<table>
<thead>
<tr>
<th>Severity</th>
<th>Acknowledge</th>
<th>Start Date</th>
<th>End Date</th>
<th>Source Name</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>8/1/2018</td>
<td>8/17/2018</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Alert Log Filters (Filter by Source Name)

<table>
<thead>
<tr>
<th>Severity</th>
<th>Acknowledge</th>
<th>Start Date</th>
<th>End Date</th>
<th>Source Name</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>8/14/2018</td>
<td>8/14/2018</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**USR0004**

**Domain:** User Tracking

**Detailed Description:**
Log-in attempt alert for root from 10:32:19:171 using VSMAAN, IP will be blocked for 60 seconds. System Display Name (SDN), System Service Tag (SST), LAN, VLAN, (SDN), (SST), (LAN), (VLAN) are embedded. 1 - Chassis Service Tag (MMD1)

**Recommended Action:**
1. Check the LCD screen and make sure the alarm display and event display are correct.
2. Check the Lifecycle Controller (LCC) Log to see if more unauthorized access attempts are occurring than would be expected due to forgotten account names or passwords.

**PSU0003**

**Domain:** Power Supply

**Detailed Description:**
The power input for power supply 6 is lost.

**Recommended Action:**
Verify the input power to the power supply. This may be the input power to the power supply or the power supply itself.
Appendix

Alert Log Filters (Filter by Category and Subcategory)

The following URIs can be used to access audit logs.

/api/ApplicationService/AuditLogs → To return a collection of audit logs.

/api/ApplicationService/AuditLogs(id) → Returns a single audit log entry.

Audit Logs Filters (REST)

The next table list the attributes that can be filter out by using a REST call.

Appendix

REST calls can be performed to retrieve filtered audit and alert logs. Perform REST requests.

More details in how to install the and how to use it can be found in the Doc section of the tool: https://www.getpostman.com/docs/v6/.

Appendix I. Using REST to apply filters to Audit Logs
Audit Logs Filters (REST)

<table>
<thead>
<tr>
<th>Filter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity</td>
<td>Filter by the severity of the EEMI message. Critical, Warning and Informational.</td>
</tr>
<tr>
<td>Message</td>
<td>Filter by the EEMI message contents.</td>
</tr>
<tr>
<td>Category</td>
<td>Filter by the Category that the EEMI message comes under.</td>
</tr>
<tr>
<td>UserName</td>
<td>Filter by the Authenticated user who generated the EEMI message.</td>
</tr>
<tr>
<td>IpAddress</td>
<td>Filter by IP address of the authenticated user.</td>
</tr>
<tr>
<td>MessageID</td>
<td>Filter by the EEMI message identifier.</td>
</tr>
<tr>
<td>CreatedDateBegin</td>
<td>Filter by Created Date (start) of the EEMI message.</td>
</tr>
<tr>
<td>CreatedDateEnd</td>
<td>Filter by Created Date (end) of the EEMI message.</td>
</tr>
</tbody>
</table>

Above filters can be combined to have a more explicit list in response. The following is a sample of using filters in the REST request.

API request with all filters:

```
/api/ApplicationService/AuditLogs?$top=30&$skip=0&$filter=Severity eq '2000' and UserName eq 'root' and IpAddress eq '127.0.0.1' and Category eq 'Configuration' and Message eq 'EPS Event management plugin doc' and CreatedDate ge '2018-08-01 05:00:00.000' and CreatedDate le '2018-08-18 04:59:59.000'
```

Figure 26  Audit Logs REST Filters
Get all Audit Logs

Get all Audit Logs

<table>
<thead>
<tr>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/api/ApplicationService/AuditLogs</td>
<td>Returns a collection of audit logs.</td>
</tr>
</tbody>
</table>

The operation to perform is **GET**. The following is an output sample of the response:

![HTTP GET Request](image)

**Figure 27** All Audit Logs REST

**Payload Output Sample:**

```json
{
    "@odata.context": "/api/$metadata#Collection(ApplicationService.AuditLog)",
    "@odata.count": 6,
    "value": [
        {
            "@odata.type": "#ApplicationService.AuditLog",
            "@odata.id": "/api/ApplicationService/AuditLogs(387)",
            "Id": 387,
            "Severity": "1000",
            "Message": "Successfully logged off from GUI.",
            "Category": "Audit",
            "UserName": "root",
            "IpAddress": "10.210.136.126",
            "MessageArgs": "GUI",
            "MessageID": "CUSR0003",
            "CreatedDate": "2018-08-21T16:47:23.554Z"
        },
        {
            "@odata.type": "#ApplicationService.AuditLog",
            "@odata.id": "/api/ApplicationService/AuditLogs(386)",
            "Id": 386,
            "Severity": "1000",
            "Message": "Successfully logged in from GUI.",
            "Category": "Audit",
            "UserName": "root",
            "IpAddress": "10.210.136.126",
            "MessageArgs": "GUI",
            "MessageID": "CMON0001",
            "CreatedDate": "2018-08-21T15:42:07.185Z"
        },
        {
            "@odata.type": "#ApplicationService.AuditLog",
            "@odata.id": "/api/ApplicationService/AuditLogs(385)",
```
Get a Single Audit Log

```
"Id": 385,
"Severity": "1000",
"Message": "Successfully logged in from GUI.",
"Category": "Audit",
"UserName": "root",
"IpAddress": "10.210.136.126",
"MessageArgs": "GUI",
"MessageID": "CMON0001",
"CreatedDate": "2018-08-21T14:41:08.004Z"
},

{"@odata.type": "#ApplicationService.AuditLog",
"@odata.id": "/api/ApplicationService/AuditLogs(384)",
"Id": 384,
"Severity": "1000",
"Message": "The job Inventory Refresh with id 27478 of type inventory has been scheduled to run now.",
"Category": "Configuration",
"UserName": "root",
"IpAddress": "10.210.136.126",
"MessageArgs": "Inventory Refresh || 27478 || inventory",
"MessageID": "CJOB0159",
"CreatedDate": "2018-08-20T20:25:03.612Z"
},

{"@odata.type": "#ApplicationService.AuditLog",
"@odata.id": "/api/ApplicationService/AuditLogs(383)",
"Id": 383,
"Severity": "1000",
"Message": "The alert(s) with ID(s) Multiple or All Event(s). are deleted.",
"Category": "Configuration",
"UserName": "root",
"IpAddress": "10.210.136.126",
"MessageArgs": "Multiple or All Event(s)",
"MessageID": "CMON0176",
"CreatedDate": "2018-08-20T20:06:06.616Z"
},

{"@odata.type": "#ApplicationService.AuditLog",
"@odata.id": "/api/ApplicationService/AuditLogs(382)",
"Id": 382,
"Severity": "1000",
"Message": "Successfully logged off from SSH.",
"Category": "Audit",
"UserName": "root",
"IpAddress": "10.210.136.126",
"MessageArgs": "SSH",
"MessageID": "CUSR0003",
"CreatedDate": "2018-08-20T20:00:20.682Z"
}
```

Get a Single Audit Log

<table>
<thead>
<tr>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/api/ApplicationService/AuditLogs(id)</td>
<td>Returns a single EEMI audit message.</td>
</tr>
</tbody>
</table>

The operation to perform is **GET**. The following is an output sample of the response:
Appendix II. Using REST to apply filters to Alert Logs

The following URIs can be used to access alert logs.

/api/AlertService/Alerts → Returns a collection of alert logs.

/api/AlertService/Alerts(id) → Returns a single alert log entry.

Alert Logs Filters (REST)

The next table list the attributes that can be filtered out by using a REST call.

<table>
<thead>
<tr>
<th>Filter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlertDeviceId</td>
<td>Filter by device id – default 0</td>
</tr>
<tr>
<td>AlertDeviceIdentifier</td>
<td>Filter by device identifier</td>
</tr>
<tr>
<td>AlertDeviceType</td>
<td>Filter by device type – default 0</td>
</tr>
<tr>
<td>SeverityType</td>
<td>Filter by severity type – default 0</td>
</tr>
<tr>
<td>StatusType</td>
<td>Filter by status type – default 0</td>
</tr>
</tbody>
</table>
Get all Alert Logs

<table>
<thead>
<tr>
<th>CategoryId</th>
<th>Filter by category id – default 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>SubCategoryId</td>
<td>Filter by sub category id – default 0</td>
</tr>
<tr>
<td>SubCategoryName</td>
<td>Filter by sub category name</td>
</tr>
<tr>
<td>Message</td>
<td>Filter by message</td>
</tr>
<tr>
<td>TimeStampBegin</td>
<td>Filter by alert time (begin)</td>
</tr>
<tr>
<td>TimeStampEnd</td>
<td>Filter by alert time (end)</td>
</tr>
<tr>
<td>AlertDeviceName</td>
<td>Filter by alert device name</td>
</tr>
</tbody>
</table>

Table 4 Alert Log Filters
Above filters can be combined to have a more explicit list in response. The following is a sample of using filters in the REST request.

API request with all filters:

/api/AlertService/Alerts?$filter=CategoryId eq 1004 and SeverityType eq 1 and StatusType eq 1000 and TimeStamp ge '2018-08-01 05:00:00.000' and TimeStamp le '2018-08-04 04:59:59.000' and AlertDeviceName eq 'test' and SubCategoryId eq 107 and Message eq 'test'&$top=30&$skip=0

Figure 29 Alert Logs REST Filters

Get all Alert Logs

<table>
<thead>
<tr>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/api/AlertService/Alerts</td>
<td>Returns a collection of alert logs.</td>
</tr>
</tbody>
</table>
The operation to perform is **GET**. The following is an output sample of the response:

```json
{
    "@odata.context": "/api/$metadata#Collection(AlertService.Alert)",
    "@odata.count": 44,
    "value": [
        {
            "@odata.type": "#AlertService.Alert",
            "@odata.id": "/api/AlertService/Alerts(919)",
            "Id": 919,
            "SeverityType": 8,
            "SeverityName": "Warning",
            "AlertDeviceId": 26990,
            "AlertDeviceName": "D123499",
            "AlertDeviceType": 1000,
            "AlertDeviceIpAddress": "10.35.0.153",
            "AlertDeviceM acAddress": "d0:94:66:2d:b8:44",
            "AlertDeviceIdentifier": "D123499",
            "AlertDeviceAssetTag": 
            "DefinitionId": 1564564330,
            "CatalogName": "iDRAC",
            "CategoryId": 1003,
            "CategoryName": "Audit",
            "SubCategoryId": 56,
            "SubCategoryName": "User Tracking",
            "StatusType": 2000,
            "StatusName": "Not-Acknowledged",
            "TimeStamp": "2018-08-21 19:59:55.183",
            "Message": "Login attempt alert for root from 10.32.19.128 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDN: WIN-02GOADDJTDCC - FQDD: IDRAC.Embedded.1 - Chassis Service Tag: MCM2",
            "EemiMessage": "N/A",
            "RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords."
        }
    ]
}
```

**Figure 30**  All Alert Logs REST

**Payload Sample Output:**

```json
{
    "@odata.context": "/api/$metadata#Collection(AlertService.Alert)",
    "@odata.count": 44,
    "value": [
        {
            "@odata.id": "/api/AlertService/Alerts(919)",
            "Id": 919,
            "SeverityType": 8,
            "SeverityName": "Warning",
            "AlertDeviceId": 26990,
            "AlertDeviceName": "D123499",
            "AlertDeviceType": 1000,
            "AlertDeviceIpAddress": "10.35.0.153",
            "AlertDeviceMacAddress": "d0:94:66:2d:b8:44",
            "AlertDeviceIdentifier": "D123499",
            "AlertDeviceAssetTag": 
            "DefinitionId": 1564564330,
            "CatalogName": "iDRAC",
            "CategoryId": 1003,
            "CategoryName": "Audit",
            "SubCategoryId": 56,
            "SubCategoryName": "User Tracking",
            "StatusType": 2000,
            "StatusName": "Not-Acknowledged",
            "TimeStamp": "2018-08-21 19:59:55.183",
            "Message": "Login attempt alert for root from 10.32.19.128 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDN: WIN-02GOADDJTDCC - FQDD: IDRAC.Embedded.1 - Chassis Service Tag: MCM2",
            "EemiMessage": "N/A",
            "RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords."
        }
    ]
}
```
Get all Alert Logs

{ "@odata.type": "#AlertService.Alert", "@odata.id": "/api/AlertService/Alerts(918)", "Id": 918, "SeverityType": 8, "SeverityName": "Warning", "AlertDeviceId": 26990, "AlertDeviceName": "D123499", "AlertDeviceType": 1000, "AlertDeviceIpAddress": "10.35.0.153", "AlertDeviceMacAddress": "d0:94:66:2d:b8:44", "AlertDeviceIdentifier": "D123499", "AlertDeviceAssetTag": ", "DefinitionId": 1564564330, "CatalogName": "iDRAC", "CategoryId": 1003, "CategoryName": "Audit", "SubCategoryId": 56, "SubCategoryName": "User Tracking", "StatusType": 2000, "StatusName": "Not-Acknowledged", "TimeStamp": "2018-08-21 19:59:49.003", "Message": "Login attempt alert for root from 10.32.19.171 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDD: WIN-02GODDDHDJTTC - FQDD: iDRAC.Embedded.1 - Chassis Service Tag: MCM2 ", "EmiMessage": "N/A", "RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords.", "AlertMessageId": "USR0034", "AlertMessage": "Login attempt alert for root from 10.32.19.171 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDD: WIN-02GODDDHDJTTC - FQDD: iDRAC.Embedded.1 - Chassis Service Tag: MCM2 ", "EmiMessage": "N/A", "RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords.", "AlertMessageId": "USR0034", "AlertMessage": "Login attempt alert for root from 10.32.19.171 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDD: WIN-02GODDDHDJTTC - FQDD: iDRAC.Embedded.1 - Chassis Service Tag: MCM2 ", "EmiMessage": "N/A", "RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords.", "AlertMessageId": "USR0034", "AlertMessage": "Login attempt alert for root from 10.32.19.171 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDD: WIN-02GODDDHDJTTC - FQDD: iDRAC.Embedded.1 - Chassis Service Tag: MCM2 ", "EmiMessage": "N/A", "RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords.", "AlertMessageId": "USR0034", "AlertMessage": "Login attempt alert for root from 10.32.19.171 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDD: WIN-02GODDDHDJTTC - FQDD: iDRAC.Embedded.1 - Chassis Service Tag: MCM2 ", "EmiMessage": "N/A", "RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords."}
Get all Alert Logs

"AlertDeviceId": 26990,
"AlertDeviceName": "D123499",
"AlertDeviceType": 1000,
"AlertDeviceIpAddress": "10.35.0.153",
"AlertDeviceMacAddress": "d0:94:66:2d:b8:44",
"AlertDeviceIdentifier": "D123499",
"AlertDeviceAssetTag": "",
"DefinitionId": 1564564330,
"CatalogName": "iDRAC",
"CategoryId": 1003,
"CategoryName": "Audit",
"SubCategoryId": 56,
"SubCategoryName": "User Tracking",
"StatusType": 2000,
"StatusName": "Not-Acknowledged",
"TimeStamp": "2018-08-21 18:59:50.741",
"Message": "Login attempt alert for root from 10.32.19.171 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDN: WIN-02GODDHDJTC - FQDD: iDRAC.Embedded.1 - Chassis Service Tag: MCM2 ",
"EemiMessage": "N/A",
"RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords.",
"AlertMessageId": "USR0034",
"AlertVarBindDetails": "<?xml version="1.0" encoding="utf-8"?><trap><agentAddress>10.35.0.153</agentAddress><enterpriseOID>.1.3.6.1.4.1.674.10892.5.3.2.4</enterpriseOID><specificTrapId>8490</specificTrapId><varbinds><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.1.0</oid><datatype>OctetString</datatype><value>USR0034</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.2.0</oid><datatype>OctetString</datatype><value>Login attempt alert for root from 10.32.19.171 using WS-MAN, IP will be blocked for 60 seconds.</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.3.0</oid><datatype>Integer32</datatype><value>4</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.4.0</oid><datatype>OctetString</datatype><value>D123499</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.5.0</oid><datatype>OctetString</datatype><value>WIN-02GODDHDJTC</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.6.0</oid><datatype>OctetString</datatype><value>iDRAC</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.7.0</oid><datatype>OctetString</datatype><value>iDRAC.Embedded.1</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.8.0</oid><datatype>OctetString</datatype><value>"root", "10.32.19.171", "WS-MAN", "60"
</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.9.0</oid><datatype>OctetString</datatype><value>MCM2</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.10.0</oid><datatype>OctetString</datatype><value>"root", "10.32.19.171", "WS-MAN", "60"
</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.11.0</oid><datatype>OctetString</datatype><value>"D123499"</value></varbind><varbinds/>
</varbinds></trap>"};

"@odata.type": "#AlertService.Alert",
"@odata.id": "/api/AlertService/Alerts(916)",
"Id": 916,
"SeverityType": 8,
"SeverityName": "Warning",
"AlertDeviceId": 26990,
"AlertDeviceName": "D123499",
"AlertDeviceType": 1000,
"AlertDeviceIpAddress": "10.35.0.153",
"AlertDeviceMacAddress": "d0:94:66:2d:b8:44",
"AlertDeviceIdentifier": "D123499",
"AlertDeviceAssetTag": "",
"DefinitionId": 1564564330,
"CatalogName": "iDRAC",
"CategoryId": 1003,
"CategoryName": "Audit",
"SubCategoryId": 56,
"SubCategoryName": "User Tracking",
"StatusType": 2000,
"StatusName": "Not-Acknowledged",
"TimeStamp": "2018-08-21 18:59:46.002",
"Message": "Login attempt alert for root from 10.32.19.171 using WS-MAN, IP will be blocked for 60 seconds. - System Display Name: iDRAC - System Service Tag: D123499 - FQDN: WIN-02GODDHDJTC - FQDD: iDRAC.Embedded.1 - Chassis Service Tag: MCM2 ",
"EemiMessage": "N/A";
Get a Single Alert Log

"RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords."

"AlertMessageId": "USR0034",
"AlertVarBindDetails": "<?xml version="1.0" encoding="utf-8"?>
<trap><agentAddress>10.35.0.153</agentAddress><enterpriseOID>.1.3.6.1.4.1.674.10892.5.3.1.1.0</enterpriseOID><specificTrapId>8490</specificTrapId><varbinds><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.2.0</oid><datatype>OctetString</datatype><value>USR0034</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.3.0</oid><datatype>OctetString</datatype><value>WIN-02GODDHDJTC</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.4.0</oid><datatype>OctetString</datatype><value>iDRAC.Embedded.1</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.5.0</oid><datatype>OctetString</datatype><value>iDRAC</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.6.0</oid><datatype>OctetString</datatype><value>\"root\",\"10.32.19.128\",\"WS-MAN\",\"60\"</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.7.0</oid><datatype>Integer32</datatype><value>4</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.8.0</oid><datatype>OctetString</datatype><value>\"Login attempt alert for root from 10.32.19.128 using WS-MAN, IP will be blocked for 60 seconds.\"</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.9.0</oid><datatype>OctetString</datatype><value>\"MCM2\"</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.10.0</oid><datatype>OctetString</datatype><value>\"\"</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.11.0</oid><datatype>OctetString</datatype><value>\"\"</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.12.0</oid><datatype>OctetString</datatype><value>\"\"</value></varbind><varbind><oid>1.3.6.1.4.1.674.10892.5.3.1.13.0</oid><datatype>OctetString</datatype><value>\"\"</value></varbind></varbinds></trap>
"AlertMessageType": "SNMP",
"MessageArgs": "",
"AlertDeviceGroup": 0

Get a Single Alert Log

<table>
<thead>
<tr>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/api/AlertService/Alerts(id)</td>
<td>Returns a single alert log message.</td>
</tr>
</tbody>
</table>

The operation to perform is **GET**. The following is an output sample of the response:

![Figure 31: Single Alert Logs REST](image-url)
Payload Sample Output:

```json
{
"@odata.context": "/api/$metadata#AlertService.Alert/$entity",
"@odata.type": "#AlertService.Alert",
"@odata.id": "/api/AlertService/Alerts(919)",
"Id": 919,
"SeverityType": 8,
"SeverityName": "Warning",
"AlertDeviceId": 26990,
"AlertDeviceName": "WIN-02G0DDHDJTC",
"AlertDeviceType": 1000,
"AlertDeviceIpAddress": "10.35.0.153",
"AlertDeviceMacAddress": "d0:94:66:2d:b8:44",
"AlertDeviceIdentifier": "D123499",
"AlertDeviceAssetTag": "",
"DefinitionId": 1564564330,
"CatalogName": "iDRAC",
"CategoryId": 1003,
"CategoryName": "Audit",
"SubCategoryId": 56,
"SubCategoryName": "User Tracking",
"StatusType": 2000,
"StatusName": "Not Acknowledged",
"TimeStamp": "2018-08-21 19:59:55.183",
"Message": "Login attempt alert for root from 10.32.19.128 using WS-MAN, IP will be blocked for 60 seconds. System Display Name: iDRAC - System Service Tag: D123499 - FQDN: WIN-02G0DDHDJTC - FQDD: iDRAC.Embedded.1 - Chassis Service Tag: MCM2 ",
"EemiMessage": "N/A",
"RecommendedAction": "Contact the iDRAC administrator and make sure the username and password credentials used are correct. Check the Lifecycle Controller Log (LC Log) to see if more unauthorized iDRAC access attempts are occurring than would be expected due to forgotten account names or passwords.",
"AlertMessageId": "USR0034",
"AlertVarBindDetails": "<xml version="1.0`