SLURM Integration with UFM

Rev 6.2

NOTE:
F F B C D C C C B A B A ' ' B C CB B A C C BCB C CAF E C - F D D
KIND AND SOLELY FOR THE PURPOSE OF AIDING THE CUSTOMER IN TESTING APPLICATIONS THAT USE THE PRODUCTS IN DESIGNATED SOLUTIONS. THE CUSTOMER'S MANUFACTURING TEST ENVIRONMENT HAS NOT MET THE STANDARDS SET BY MELLANOX TECHNOLOGIES TO FULLY QUALIFY THE PRODUCT(S) AND/OR THE SYSTEM USING IT. THEREFORE, MELLANOX TECHNOLOGIES CANNOT AND DOES NOT GUARANTEE OR WARRANT THAT THE PRODUCTS WILL OPERATE WITH THE HIGHEST QUALITY. ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT ARE DISCLAIMED. IN NO EVENT SHALL MELLANOX BE LIABLE TO CUSTOMER OR ANY THIRD PARTIES FOR ANY DIRECT, INDIRECT, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES OF ANY KIND (INCLUDING, BUT NOT LIMITED TO, PAYMENT FOR PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY FROM THE USE OF THE PRODUCT(S) AND RELATED DOCUMENTATION EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Mellanox Technologies
350 Oakmead Parkway Suite 100
Sunnyvale, CA 94085
U.S.A.
www.mellanox.com
Tel: (408) 970-3400
Fax: (408) 970-3403

Table of Contents

1 Introduction

1.1 The Slurm Workload Manager

formerly known as Simple Linux Utility for Resource Management or SLURM, or Slurm, is a free and open-source Job scheduler for Linux and Unix-like kernels, used by many of the world's supercomputers and computer clusters. It provides three key functions. First, it allocates exclusive and/or non-exclusive access to resources (computer nodes) to users for some duration of time so they can perform work. Second, it provides a framework for starting, executing, and monitoring work (typically a parallel job such as MPI) on a set of allocated nodes. Finally, it arbitrates contention for resources by managing a queue of pending jobs.

1.2 Mellanox's Unified Fabric Manager (UFM®)

Mellanox's Unified Fabric Manager (UFM®) is a powerful platform for managing scale-out computing environments.

UFM enables data center operators to monitor, efficiently provision, and operate the modern data center fabric. UFM eliminates the complexity of fabric management, provides deep visibility into traffic and optimizes fabric performance.

1.3 UFM Integration with Slurm:

By integrating SLURM with UFM, you can track Network bandwidth, congestion, errors and resource utilization of compute nodes, which are being used by Slurm jobs.

- Real-time monitoring views
- Alerts notification for error threshold exceeded
- Multiple Objects: diverse monitoring views
- Multiple attributes monitoring
- SUM, AVG, MIN, MAX, RAW functions for monitored attributes

In summary, for the first phase of this integration UFM will support the monitoring feature for SLURM compute nodes.

2 Setup Integration:

The integration between UFM and SLURM need some prerequisites: environments, files and configuration to be done.

2.1 Prerequisites:

UFM 6.2.5 (or newer versions) installed on a RH7x machine with LS auditing enabled.

UFM REST SDK 6.2.5 (or newer versions) installed on SLURM controller.

Python 2.7 on SLURM controller.

UFM-SLURM Integration files which provided independently.

2.2 Setup:

First, you should enable LS auditing feature (review section 6.2) . Then start the UFM in "Management" mode.

Extract UFM-SLURM Integration tar file, then copy the UFM-SLURM integration files to SLURM controller folder for example "/etc/slurm-llnl".

Change the permissions of UFM-SLURM Integration files into 755.

Set UFM configuration info in the UFM-SLURM config file "/etc/ slurm-llnl/ufm_slurm.conf". UFM Server IP {ufm_server}, UFM server credentials {ufm_server_user, ufm_server_pass}, and UFM protocol{protocol} used to connect to UFM (default is https).

Change permissions for the log folder of UFM REST SDK: /usr/opt/ufmrestsdk/log/ into 777 (create the log folder if not exists)

Modify SLURM configuration file on SLURM controller "/etc/slurm-llnl/slurm.conf" and add/modify the following two parameters:

```
PrologSlurmctld=/etc/slurm-llnl/ufm-prolog.sh

EpilogSlurmctld=/etc/slurm-llnl/ufm-epilog.sh
```

2.3 Run:

After setup and configure the files, now you are ready to use the integration. Using the Slurm controller execute the following commands to run your batch job

```
Submitted batch job 1
```

-N4 is the number of compute nodes used to run the jobs. slurm_demo.sh is the job batch file to be run.

After Slurm executed the batch file on compute nodes. The output and result are stored on submitted working directory “slurm-{id} .out “where {id} is id of the submitted job. For example, in the above example after executing sbatch command, the submitted job ID is 1, so the output file is stored in “slurm-1.out”.

Execute the following command to see the output.

```
$cat slurm-1.out
```

In UFM side, a Job scheduler is created for the submitted job with a name “slurm_job_{job_id}” where {job_id} is the id of the submitted job on SLURM side. All the compute nodes which assigned to this job are allocated to this Job scheduler. After the SLURM job is completed, the UFM removes the Job scheduler.

During the execution of the SLURM job, UFM monitors and records traffic and data on the assigned compute nodes, which you could see by UFM feature called “Job scheduler Life Cycle Auditing”. For more information regarding UFM “Job scheduler Auditing” please refer to section 6 or UFM User Manual.

You can use UFM Web GUI to show the created logical servers, located nodes and monitoring data.

Also, from the time that a job is submitted by SLURM server until completion, a log file called “/etc/slurm-llnl /ufm_slurm.log” logs all the actions and errors occurs during the execution. This log file could be changed by modify “log_file_name” parameter in UFM_SLURM config file “/etc/slurm-llnl /ufm_slurm.conf”.

3 UFM SLURM Config file

The integration process uses a configuration file located at "/etc/slurm-llnl/ufm_slurm.conf". This file is used to configure some settings and attributes for integration. Here are the contents:

Attribute Name	Description	Optionality
ufm_server_user	A Username of UFM server used to connect to UFM	Mandatory
ufm_server_pass	The password related to ufm_server_user	Mandatory
ufm_server	UFM server IP address to connect to	Mandatory
log_file_name	The name of integration logging file	Mandatory
protocol	The protocol used to connect to UFM	Mandatory
add_related_nics	(True/False) Determine if to add all the job' node NICs to the logical server or not. Default is True.	Optional

4 Prolog and Epilog:

After submitting jobs on SLURM, there are two scripts that are automatically executed. Prolog is a script which executed when a job is submitted and before running the job itself. Epilog is the script executed when a job is completed.

In UFM-SLURM integration, there are two file scripts:

- ufm-prolog.sh: This script is called automatically by SLURM prolog script. It creates the Job scheduler and assign compute nodes to it.
- ufm-epilog.sh: This script is called automatically by SLURM epilog script. It removes the logical servers and free compute nodes from it.

5 Integration Files:

The integration use scripts and configuration files to work, which should be copied to SLURM controller "/etc"/slurm-llnl". Here is a list of these files:

File Name	Description
ufm-prolog.sh	A bash file which executes jobs related to UFM before SLURM job is executed.
ufm-epilog.sh	A bash file which executes jobs related to UFM after SLURM job is completed.
ufm_slurm.conf	UFM-SLURM integration configuration file.
ufm_slurm_prolog.py	A python script file which creates Job schedulers and assign compute nodes to it. It called by ufm-prolog.sh
ufm_slurm_epilog.py	A python script file which removes the created Job scheduler during prolog. It called by ufm-epilog.sh
ufm_slurm_utils.py	A utility python file contains functions and utilities used by integration process.

6 Job scheduler Auditing

6.1 Feature Overview

The main purpose of this feature is to display information on Job schedulers' job lifecycle events and performance counters of Job schedulers' job, and its related computes. This feature is disabled by default.

A new created job at the job scheduler (SLURM), triggers a creation of a new Logical Server object which is being monitored by UFM during the job lifecycle.

The Logical Server object consists of the list of physical servers that were allocated for the scheduler job.

Once the scheduler job is terminated, the Logical Server object is removed from the UFM.

NOTES:

- *__From this point__, the term "__Logical Server referring to the Job__. (ie: Logical Sever Auditing means: Job scheduler's job auditing)
- This feature is supported on **RedHat 7.x** operating systems **only**.

Logical Server Lifecycle Events

Once the feature is enabled, a new file will be created upon every Logical Server creation, in the following location: **/opt/ufm/files/jobs/auditing**, and all Logical Server lifecycle events will be written to that file. The name of the created file will be in the following format: [env name].[Logical Server name].

Full path example:

/opt/ufm/files/jobs/auditing/env1.ls1

The following is a list of all supported events that can be written to the file:

- Logical Server State Changed
- Logical Server State Change Failed
- Logical Server Added
- Logical Server Removed
- Logical Server Resources Allocated
- Compute Resource Released
- Compute Resource Allocated
- Logical Server Additional Resources Allocated
- Logical Server Resources Released
- Logical Server Compute Resource is Down
- Logical Server Compute Resource is Up
- Attach Network to Logical Server
- Detach Network from Logical Server

6.1.1.1 Logical Server Performance Data

Upon every Logical Server creation, a new entry will be created in graphite database (carbon-cache), where all selected counters are stored for the Logical Server and its computes. Data will be stored in a pre-defined resolution (configured in the gv.cfg file using `accounting_data_flush_interval` parameter).

Default time interval is 30

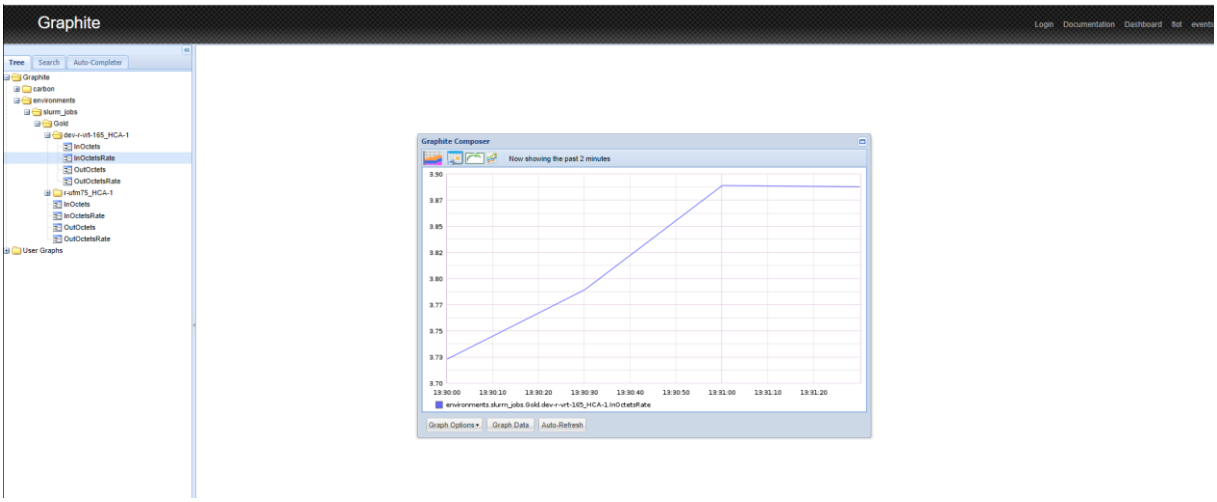
- "OutOctetsRate"
- "NormalizedBandwidth"
- "NormalizedCongestedBandwidth"
- "CumulativeErrors"
- "LinkDowned"

6.3 Logical Server Auditing Information Retrieval

Logical Server Auditing data can be retrieved via UFM REST API. For all available Logical Server Auditing UFM REST APIs, refer to **Logical Server (LS) Auditing REST API** section in UFM_REST_API document.

6.4 Viewing Logical Server Performance Data via Graphite Web UI

Logical Server auditing performance data can be viewed using graphite web UI. URL should be in the following format: <UFM-host-name>:<graphite-port-number[1]>. Note that UFM credentials are required.



6.5 Reconfiguring Logical Server Auditing Statistical Storage

By default, Logical Server performance data is stored in graphite database every 30 seconds, and is kept for a one-month period. For changing this behavior, perform the following:

1. Change the value of the following parameters: `accounting_data_flush_interval`, `data_aging_time_interval` in UFM gv.cfg file.
2. Run the following script to reconfigure graphite and fill in the required data:

```
/opt/ufm/scripts/configure_graphite.sh
```

Note: To avoid data corruption, make sure UFM is stopped before running this script.

NOTE: This operation will delete all previously saved Logical Server performance data from graphite database.

[1] Graphite-port-number can be found in the gv.cfg file in graphite_port_number field.

7 Logical Server REST APIs

7.1 Description

These interfaces allow users to retrieve information on Logical Server lifecycle events and Logical Server performance information during lifecycle

7.2 Lifecycle Events APIs

Main Request URL

GET /ufmRest/resources/environments/auditing

Main Operations

- Get all the Logical Server available for auditing grouped by environment
- Get all the Logical Server available for auditing for specified environment
- Get all the Logical Server lifecycle events for all Logical Servers, grouped by environment
- Get all the Logical Server lifecycle events for specified Logical Server

7.2.1 API - Get All the Logical Servers available for Auditing Grouped by Environment

Retrieve list of Logical Server names grouped by environment name

GET /ufmRest/resources/environments/auditing

NA

```
{
    "env2": [
        "Logical Server_2_2",
        "Logical Server_2_1"
    ],
    "env1": [
        "Logical Server_1_1",
        "Logical Server_1_2"
    ]
}
```

NA

Status	Description
200	OK
400	BAD REQUEST

7.2.2 API - Get all the Logical Servers available for auditing for specified environment

Retrieve list of Logical Server names for specified environment name

GET /ufmRest/resources/environments/<env_name>/auditing

• Request Content Type

NA

```
[
    "Logical Server_1_1",
    "Logical Server_1_2"
]
```

NA

Status	Description
200	OK
400	BAD REQUEST

7.2.3 API - Get all the Logical Server lifecycle events for all Logical Servers, grouped by Environment

Description

Retrieve all Logical Server lifecycle events for all the Logical Server grouped by environment name

GET /ufmRest/resources/environments/<env_name>/logical_servers/auditing

NA


```
[
    {
        "job_id": "env1.Logical Server_1_1",
        "life_cycle_report": [
            {
                "description": "Network Interface env1_Logical Server_1_1_management added to Logical Server Logical Server_1_1",
                "event": "Attach Network to Logical Server",
                "time": "2017-09-20 06:51:14"
            },
            {
                "description": "Created Logical Server Logical Server_1_1",
                "event": "Logical Server Created",
                "time": "2017-09-20 06:51:14"
            },
            {
                "description": "Compute Logical Server_1_1/1 (smg-ib-apl001-gen1 HCA-1:0) allocated to Logical Server Logical Server_1_1",
                "event": "Compute Resource Allocated",
                "time": "2017-09-20 06:51:16"
            },
            {
                "description": "Compute Logical Server_1_1/2 (smg-ib-apl002-gen1 HCA-1:0) allocated to Logical Server Logical Server_1_1",
                "event": "Compute Resource Allocated",
```

```
                "time": "2017-09-20 06:51:16"
            },
            {
                "description": " Logical Server Logical Server_1_1: 2 resources allocated",
                "event": "Logical Server Resources Allocated",
                "time": "2017-09-20 06:51:16"
            },
            {
                "description": "Logical Server state changed from running to allocated",
                "event": "Logical Server State Changed",
                "time": "2017-09-20 06:51:16"
            },
            {
                "description": "Logical Server state changed from allocated to deployed",
                "event": "Logical Server State Changed",
                "time": "2017-09-20 06:51:17"
            },
            {
                "description": "Logical Server state changed from deployed to running",
                "event": "Logical Server State Changed",
                "time": "2017-09-20 06:51:18"
            }
        ]
    },
    {
        "job_id": "env1.Logical Server_1_2",
        "life_cycle_report": [
            {
                "description": "Network Interface env1_Logical Server_1_2_management
added to Logical Server Logical Server_1_2",
                "event": "Attach Network to Logical Server",
                "time": "2017-09-20 06:51:36"
            },
            {
```

```
            "description": "Created Logical Server Logical Server_1_2",
            "event": "Logical Server Created",
            "time": "2017-09-20 06:51:36"
        },
        {
            "description": "Compute Logical Server_1_2/1 (smg-ib-apl003-gen1 HCA-1:0) allocated to Logical Server Logical Server_1_2",
            "event": "Compute Resource Allocated",
            "time": "2017-09-20 06:51:38"
        },
        {
            "description": "Compute Logical Server_1_2/2 (smg-ib-apl005-gen2 HCA-1:0) allocated to Logical Server Logical Server_1_2",
            "event": "Compute Resource Allocated",
            "time": "2017-09-20 06:51:38"
        },
        {
            "description": "Compute Logical Server_1_2/3 (smg-ib-apl004-gen1 HCA-1:0) allocated to Logical Server Logical Server_1_2",
            "event": "Compute Resource Allocated",
            "time": "2017-09-20 06:51:38"
        },
        {
            "description": " Logical Server Logical Server_1_2: 3 resources allocated",
            "event": "Logical Server Resources Allocated",
            "time": "2017-09-20 06:51:38"
        },
        {
            "description": "Logical Server state changed from running to allocated",
            "event": "Logical Server State Changed",
            "time": "2017-09-20 06:51:38"
        },
        {
            "description": "Logical Server state changed from allocated to deployed",
            "event": "Logical Server State Changed",
```

```
            "time": "2017-09-20 06:51:39"
        },
        {
            "description": "Logical Server state changed from deployed to running",
            "event": "Logical Server State Changed",
            "time": "2017-09-20 06:51:40"
        }
    ]
  }
]
```


N/A

Status	Description
200	OK
400	BAD REQUEST

7.2.4 API - Get all the Logical Server lifecycle events for specific Logical Server

Retrieve all Logical Server lifecycle events for all the Logical Server grouped by environment name

GET /ufmRest/resources/environments/<env_name>/logical_servers/<Logical Server_name>/auditing

NA


```
{
    "job_id": "env1.Logical Server_1_2",
    "life_cycle_report": [
        {
            "description": "Network Interface env1_Logical Server_1_2_management added to Logical Server Logical Server_1_2",
            "event": "Attach Network to Logical Server",
            "time": "2017-09-20 06:51:36"
        }, {
            "description": "Created Logical Server Logical Server_1_2",
            "event": "Logical Server Created",
            "time": "2017-09-20 06:51:36"
        }, {
            "description": "Compute Logical Server_1_2/1 (smg-ib-apl003-gen1 HCA-1:0) allocated to Logical Server Logical Server_1_2",
            "event": "Compute Resource Allocated",
            "time": "2017-09-20 06:51:38"
        }, {
            "description": "Compute Logical Server_1_2/2 (smg-ib-apl005-gen2 HCA-1:0) allocated to Logical Server Logical Server_1_2",
            "event": "Compute Resource Allocated",
            "time": "2017-09-20 06:51:38"
        }, {
            "description": "Compute Logical Server_1_2/3 (smg-ib-apl004-gen1 HCA-1:0) allocated to Logical Server Logical Server_1_2",
```

```
            "event": "Compute Resource Allocated",
            "time": "2017-09-20 06:51:38"
        }, {
            "description": " Logical Server Logical Server_1_2: 3 resources allocated",
            "event": "Logical Server Resources Allocated",
            "time": "2017-09-20 06:51:38"
        }, {
            "description": "Logical Server state changed from running to allocated",
            "event": "Logical Server State Changed",
            "time": "2017-09-20 06:51:38"
        }, {
            "description": "Logical Server state changed from allocated to deployed",
            "event": "Logical Server State Changed",
            "time": "2017-09-20 06:51:39"
        }, {
            "description": "Logical Server state changed from deployed to running",
            "event": "Logical Server State Changed",
            "time": "2017-09-20 06:51:40"
        }
    ]
}
```


NA

Status	Description
200	OK
400	BAD REQUEST

7.3 Logical Server Performance Data API

Main Request URL

GET /ufmRest/resources/environments/<env_name>/logical_servers/<ls_name>/performance

Main Operations

- Get Logical Server performance information for specific Logical Server

7.3.1 API - Get Logical Server performance counters for specific Logical Server

Retrieve Logical Server performance counters for specified environment name and Logical Server name

GET /ufmRest/resources/environments/<env_name>/logical_servers/<Logical Server_name>/performance

NA


```
{
    "env1.Logical Server_1_2": {
        "1505890740": {
            "nodes": {
                "smg-ib-apl005-gen2_HCA-1": {
                    "values": [
                        2.0,
                        2.0
                    ]
                },
                "smg-ib-apl004-gen1_HCA-1": {
                    "values": [
                        6691.0,
                        714.0
                    ]
                },
                "smg-ib-apl003-gen1_HCA-1": {
                    "values": [
                        13647.0,
                        8132.0
                    ]
                }
            },
            "values": [
                20340.0,
```

```
            8848.0
        ]
    },
    "1505890650": {
        "nodes": {
            "smg-ib-apl005-gen2_HCA-1": {
                "values": [
                    2.0,
                    2.0
                ]
            },
            "smg-ib-apl004-gen1_HCA-1": {
                "values": [
                    6691.0,
                    714.0
                ]
            },
            "smg-ib-apl003-gen1_HCA-1": {
                "values": [
                    13646.0,
                    8131.0
                ]
            }
        },
        "values": [
            20339.0,
            8847.0
        ]
    },
    "1505890770": {
        "nodes": {
            "smg-ib-apl005-gen2_HCA-1": {
                "values": [
                    2.0,
```

```
                2.0
            ]
        },
        "smg-ib-apl004-gen1_HCA-1": {
            "values": [
                6691.0,
                714.0
            ]
        },
        "smg-ib-apl003-gen1_HCA-1": {
            "values": [
                13647.0,
                8132.0
            ]
        }
    },
    "values": [
        20340.0,
        8848.0
    ]
},
"1505890680": {
    "nodes": {
        "smg-ib-apl005-gen2_HCA-1": {
            "values": [
                2.0,
                2.0
            ]
        },
        "smg-ib-apl004-gen1_HCA-1": {
            "values": [
                6691.0,
                714.0
            ]
```

```
            },
            "smg-ib-apl003-gen1_HCA-1": {
                "values": [
                    13647.0,
                    8132.0
                ]
            }
        },
        "values": [
            20340.0,
            8848.0
        ]
    },
    "1505890710": {
        "nodes": {
            "smg-ib-apl005-gen2_HCA-1": {
                "values": [
                    2.0,
                    2.0
                ]
            },
            "smg-ib-apl004-gen1_HCA-1": {
                "values": [
                    6691.0,
                    714.0
                ]
            },
            "smg-ib-apl003-gen1_HCA-1": {
                "values": [
                    13647.0,
                    8132.0
                ]
            }
        },
```

```
        "values": [
            20340.0,
            8848.0
        ]
    },
    "1505890620": {
        "nodes": {
            "smg-ib-apl005-gen2_HCA-1": {
                "values": [
                    2.0,
                    2.0
                ]
            },
            "smg-ib-apl004-gen1_HCA-1": {
                "values": [
                    6691.0,
                    714.0
                ]
            },
            "smg-ib-apl003-gen1_HCA-1": {
                "values": [
                    13646.0,
                    8131.0
                ]
            }
        },
        "values": [
            20339.0,
            8847.0
        ]
    },
    "counters": [
        "InOctets",
        "OutOctets"
```

```
        ]
    }
}
```

- **Counters names names of counters to be retrieved. Example:**

/ufmRest/resources/environments/<env_name>/logical_servers/<Logical Server_name>/performance?counters=InOctets,OutOctets

Possible values: InOctets, OutOctets, InOctetsRate, OutOctetsRate, NormalizedBandwidth, CumulativeErrors

- **Report start time Starting time point counters to be retrieved from. Example:**

/ufmRest/resources/environments/<env_name>/logical_servers/<Logical Server_name>/performance? start_time=-10min

Possible time units names:

Time Unit	Meaning
s	Seconds
min	Minutes
h	Hours
d	Days
w	Weeks
mon	30 Days (month)
y	365 Days (year)

- **Report end time End time point counters to be retrieved before. Example:**

/ufmRest/resources/environments/<env_name>/logical_servers/<Logical Server_name>/performance?

end_time=-5min

Possible time units names:

Time Unit	**Meaning**
s	Seconds
min	Minutes

h	Hours
d	Days
w	Weeks
mon	30 Days (month)
y	365 Days (year)

Complex Example:

This request will bring data started 10 minutes ago and ended 5 minutes ago for InOctets and OutOctets counters.

/ufmRest/resources/environments/<env_name>/logical_servers/<Logical Server_name>/performance?counters=InOctets,OutOctets&start_time=-10min&end_time=-5min

Status	Description
200	OK
400	BAD REQUEST