



# **Manage NVMe services**

## **ONTAP 9.10.1 REST API Documentation**

NetApp  
January 31, 2025

# Table of Contents

- Manage NVMe services . . . . . 1
  - Protocols NVMe services endpoint overview . . . . . 1
  - Retrieve NVMe services . . . . . 6
  - Create an NVMe service . . . . . 36
  - Delete an NVMe service . . . . . 58
  - Retrieve an NVMe service . . . . . 61
  - Update an NVMe service . . . . . 78
  - Retrieve NVMe protocol historical performance metrics . . . . . 96

# Manage NVMe services

## Protocols NVMe services endpoint overview

### Overview

A Non-Volatile Memory Express (NVMe) service defines the properties of the NVMe controller target for an SVM. There can be at most one NVMe service for an SVM. An SVM's NVMe service must be created before NVMe host initiators can connect to the SVM.

The Non-Volatile Memory Express (NVMe) service REST API allows you to create, update, delete, and discover NVMe services for SVMs.

### Performance monitoring

Performance of the SVM can be monitored by the `metric.*` and `statistics.*` properties. These show the performance of the SVM in terms of IOPS, latency and throughput. The `metric.*` properties denote an average whereas `statistics.*` properties denote a real-time monotonically increasing value aggregated across all nodes.

### Examples

#### Creating an NVMe service for an SVM

The simplest way to create an NVMe service is to specify only the SVM, either by name or UUID. By default, the new NVMe service is enabled.

In this example, the `return_records` query parameter is used to retrieve the new NVMe service object in the REST response.

```
# The API:
POST /api/protocols/nvme/services

# The call:
curl -X POST 'https://<mgmt-
ip>/api/protocols/nvme/services?return_records=true' -H 'accept:
application/hal+json' -d '{ "svm": { "name": "svm1" } }'

# The response:
{
  "num_records": 1,
  "records": [
    {
      "svm": {
        "uuid": "bfb1beb0-dc69-11e8-b29f-005056bb7341",
        "name": "svm1",
        "_links": {
          "self": {
            "href": "/api/svm/svms/bfb1beb0-dc69-11e8-b29f-005056bb7341"
          }
        }
      },
      "enabled": true,
      "_links": {
        "self": {
          "href": "/api/protocols/nvme/services/bfb1beb0-dc69-11e8-b29f-
005056bb7341"
        }
      }
    }
  ]
}
```

## Retrieving the NVMe services for all SVMs in the cluster

```
# The API:
GET /api/protocols/nvme/services

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/nvme/services' -H 'accept:
application/hal+json'

# The response:
```

```

{
  "records": [
    {
      "svm": {
        "uuid": "ab60c350-dc68-11e8-9711-005056bbe408",
        "name": "svm0",
        "_links": {
          "self": {
            "href": "/api/svm/svms/ab60c350-dc68-11e8-9711-005056bbe408"
          }
        }
      },
      "_links": {
        "self": {
          "href": "/api/protocols/nvme/services/ab60c350-dc68-11e8-9711-005056bbe408"
        }
      }
    },
    {
      "svm": {
        "uuid": "bfb1beb0-dc69-11e8-b29f-005056bb7341",
        "name": "svm1",
        "_links": {
          "self": {
            "href": "/api/svm/svms/bfb1beb0-dc69-11e8-b29f-005056bb7341"
          }
        }
      },
      "_links": {
        "self": {
          "href": "/api/protocols/nvme/services/bfb1beb0-dc69-11e8-b29f-005056bb7341"
        }
      }
    }
  ],
  "num_records": 2,
  "_links": {
    "self": {
      "href": "/api/protocols/nvme/services"
    }
  }
}

```

## Retrieving details for a specific NVMe service

The NVMe service is identified by the UUID of its SVM.

```
# The API:
GET /api/protocols/nvme/services/{svm.uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/nvme/services/bfb1beb0-dc69-11e8-b29f-005056bb7341' -H 'accept: application/hal+json'

# The response:
{
  "svm": {
    "uuid": "bfb1beb0-dc69-11e8-b29f-005056bb7341",
    "name": "svm1",
    "_links": {
      "self": {
        "href": "/api/svm/svms/bfb1beb0-dc69-11e8-b29f-005056bb7341"
      }
    }
  },
  "enabled": true,
  "_links": {
    "self": {
      "href": "/api/protocols/nvme/services/bfb1beb0-dc69-11e8-b29f-005056bb7341"
    }
  }
}
```

## Disabling an NVMe service

Disabling an NVMe service shuts down all active NVMe connections for the SVM and prevents the creation of new NVMe connections.

The NVMe service to update is identified by the UUID of its SVM.

```
# The API:
PATCH /api/protocols/nvme/services/{svm.uuid}

# The call:
curl -X PATCH 'https://<mgmt-ip>/api/protocols/nvme/services/bfb1beb0-
dc69-11e8-b29f-005056bb7341' -H 'accept: application/hal+json' -d '{
"enabled": "false" }'
```

You can retrieve the NVMe service to confirm the change.

```
# The API:
GET /api/protocols/nvme/services/{svm.uuid}

# The call:
curl -X GET 'https://<mgmt-ip>/api/protocols/nvme/services/bfb1beb0-dc69-
11e8-b29f-005056bb7341' -H 'accept: application/hal+json'

# The response:
{
"svm": {
  "uuid": "bfb1beb0-dc69-11e8-b29f-005056bb7341",
  "name": "svm1",
  "_links": {
    "self": {
      "href": "/api/svm/svms/bfb1beb0-dc69-11e8-b29f-005056bb7341"
    }
  }
},
"enabled": false,
"_links": {
  "self": {
    "href": "/api/protocols/nvme/services/bfb1beb0-dc69-11e8-b29f-
005056bb7341"
  }
}
}
```

## Deleting an NVMe service

The NVMe service must be disabled before it can be deleted. In addition, all NVMe interfaces, subsystems, and subsystem maps associated with the SVM must first be deleted.

The NVMe service to delete is identified by the UUID of its SVM.

```
# The API:
DELETE /api/protocols/nvme/services/{svm.uuid}

# The call:
curl -X DELETE 'https://<mgmt-ip>/api/protocols/nvme/services/bfb1beb0-
dc69-11e8-b29f-005056bb7341' -H 'accept: application/hal+json'
```

## Retrieve NVMe services

GET /protocols/nvme/services

**Introduced In:** 9.6

Retrieves NVMe services.

### Expensive properties

There is an added cost to retrieving values for these properties. They are not included by default in GET results and must be explicitly requested using the `fields` query parameter. See [Requesting specific fields](#) to learn more.

- `statistics.*`
- `metric.*`

### Related ONTAP commands

- `vserver nvme show`

### Learn more

- [DOC /protocols/nvme/services](#)

### Parameters

Name	Type	In	Required	Description
<code>metric.tcp.duration</code>	string	query	False	Filter by <code>metric.tcp.duration</code> <ul style="list-style-type: none"><li>• Introduced in: 9.10</li></ul>
<code>metric.tcp.timestamp</code>	string	query	False	Filter by <code>metric.tcp.timestamp</code> <ul style="list-style-type: none"><li>• Introduced in: 9.10</li></ul>



Name	Type	In	Required	Description
metric.tcp.latency.total	integer	query	False	Filter by metric.tcp.latency.total  • Introduced in: 9.10
metric.tcp.latency.write	integer	query	False	Filter by metric.tcp.latency.write  • Introduced in: 9.10
metric.tcp.latency.other	integer	query	False	Filter by metric.tcp.latency.other  • Introduced in: 9.10
metric.tcp.latency.read	integer	query	False	Filter by metric.tcp.latency.read  • Introduced in: 9.10
metric.tcp.iops.total	integer	query	False	Filter by metric.tcp.iops.total  • Introduced in: 9.10
metric.tcp.iops.write	integer	query	False	Filter by metric.tcp.iops.write  • Introduced in: 9.10
metric.tcp.iops.other	integer	query	False	Filter by metric.tcp.iops.other  • Introduced in: 9.10

Name	Type	In	Required	Description
metric.tcp.iops.read	integer	query	False	Filter by metric.tcp.iops.read  • Introduced in: 9.10
metric.tcp.status	string	query	False	Filter by metric.tcp.status  • Introduced in: 9.10
metric.tcp.throughput.read	integer	query	False	Filter by metric.tcp.throughput.read  • Introduced in: 9.10
metric.tcp.throughput.total	integer	query	False	Filter by metric.tcp.throughput.total  • Introduced in: 9.10
metric.tcp.throughput.write	integer	query	False	Filter by metric.tcp.throughput.write  • Introduced in: 9.10
metric.duration	string	query	False	Filter by metric.duration  • Introduced in: 9.7
metric.iops.total	integer	query	False	Filter by metric.iops.total  • Introduced in: 9.7

Name	Type	In	Required	Description
metric.iops.write	integer	query	False	Filter by metric.iops.write  • Introduced in: 9.7
metric.iops.other	integer	query	False	Filter by metric.iops.other  • Introduced in: 9.7
metric.iops.read	integer	query	False	Filter by metric.iops.read  • Introduced in: 9.7
metric.throughput.read	integer	query	False	Filter by metric.throughput.read  • Introduced in: 9.7
metric.throughput.total	integer	query	False	Filter by metric.throughput.total  • Introduced in: 9.7
metric.throughput.write	integer	query	False	Filter by metric.throughput.write  • Introduced in: 9.7
metric.timestamp	string	query	False	Filter by metric.timestamp  • Introduced in: 9.7

Name	Type	In	Required	Description
metric.latency.total	integer	query	False	Filter by metric.latency.total  • Introduced in: 9.7
metric.latency.write	integer	query	False	Filter by metric.latency.write  • Introduced in: 9.7
metric.latency.other	integer	query	False	Filter by metric.latency.other  • Introduced in: 9.7
metric.latency.read	integer	query	False	Filter by metric.latency.read  • Introduced in: 9.7
metric.fc.duration	string	query	False	Filter by metric.fc.duration  • Introduced in: 9.10
metric.fc.timestamp	string	query	False	Filter by metric.fc.timestamp  • Introduced in: 9.10
metric.fc.latency.total	integer	query	False	Filter by metric.fc.latency.total  • Introduced in: 9.10
metric.fc.latency.write	integer	query	False	Filter by metric.fc.latency.write  • Introduced in: 9.10

Name	Type	In	Required	Description
metric.fc.latency.other	integer	query	False	Filter by metric.fc.latency.other  • Introduced in: 9.10
metric.fc.latency.read	integer	query	False	Filter by metric.fc.latency.read  • Introduced in: 9.10
metric.fc.iops.total	integer	query	False	Filter by metric.fc.iops.total  • Introduced in: 9.10
metric.fc.iops.write	integer	query	False	Filter by metric.fc.iops.write  • Introduced in: 9.10
metric.fc.iops.other	integer	query	False	Filter by metric.fc.iops.other  • Introduced in: 9.10
metric.fc.iops.read	integer	query	False	Filter by metric.fc.iops.read  • Introduced in: 9.10
metric.fc.status	string	query	False	Filter by metric.fc.status  • Introduced in: 9.10

Name	Type	In	Required	Description
metric.fc.throughput.read	integer	query	False	Filter by metric.fc.throughput.read  • Introduced in: 9.10
metric.fc.throughput.total	integer	query	False	Filter by metric.fc.throughput.total  • Introduced in: 9.10
metric.fc.throughput.write	integer	query	False	Filter by metric.fc.throughput.write  • Introduced in: 9.10
metric.status	string	query	False	Filter by metric.status  • Introduced in: 9.7
enabled	boolean	query	False	Filter by enabled
statistics.tcp.latency_raw.total	integer	query	False	Filter by statistics.tcp.latency_raw.total  • Introduced in: 9.10
statistics.tcp.latency_raw.write	integer	query	False	Filter by statistics.tcp.latency_raw.write  • Introduced in: 9.10
statistics.tcp.latency_raw.other	integer	query	False	Filter by statistics.tcp.latency_raw.other  • Introduced in: 9.10

Name	Type	In	Required	Description
statistics.tcp.latency_raw.read	integer	query	False	Filter by statistics.tcp.latency_raw.read  • Introduced in: 9.10
statistics.tcp.status	string	query	False	Filter by statistics.tcp.status  • Introduced in: 9.10
statistics.tcp.iops_raw.total	integer	query	False	Filter by statistics.tcp.iops_raw.total  • Introduced in: 9.10
statistics.tcp.iops_raw.write	integer	query	False	Filter by statistics.tcp.iops_raw.write  • Introduced in: 9.10
statistics.tcp.iops_raw.other	integer	query	False	Filter by statistics.tcp.iops_raw.other  • Introduced in: 9.10
statistics.tcp.iops_raw.read	integer	query	False	Filter by statistics.tcp.iops_raw.read  • Introduced in: 9.10
statistics.tcp.timestamp	string	query	False	Filter by statistics.tcp.timestamp  • Introduced in: 9.10

Name	Type	In	Required	Description
statistics.tcp.throughput_raw.read	integer	query	False	Filter by statistics.tcp.throughput_raw.read  • Introduced in: 9.10
statistics.tcp.throughput_raw.total	integer	query	False	Filter by statistics.tcp.throughput_raw.total  • Introduced in: 9.10
statistics.tcp.throughput_raw.write	integer	query	False	Filter by statistics.tcp.throughput_raw.write  • Introduced in: 9.10
statistics.throughput_raw.read	integer	query	False	Filter by statistics.throughput_raw.read  • Introduced in: 9.7
statistics.throughput_raw.total	integer	query	False	Filter by statistics.throughput_raw.total  • Introduced in: 9.7
statistics.throughput_raw.write	integer	query	False	Filter by statistics.throughput_raw.write  • Introduced in: 9.7
statistics.timestamp	string	query	False	Filter by statistics.timestamp  • Introduced in: 9.7



Name	Type	In	Required	Description
statistics.fc.latency_r aw.total	integer	query	False	Filter by statistics.fc.latency_r aw.total  • Introduced in: 9.10
statistics.fc.latency_r aw.write	integer	query	False	Filter by statistics.fc.latency_r aw.write  • Introduced in: 9.10
statistics.fc.latency_r aw.other	integer	query	False	Filter by statistics.fc.latency_r aw.other  • Introduced in: 9.10
statistics.fc.latency_r aw.read	integer	query	False	Filter by statistics.fc.latency_r aw.read  • Introduced in: 9.10
statistics.fc.status	string	query	False	Filter by statistics.fc.status  • Introduced in: 9.10
statistics.fc.iops_raw .total	integer	query	False	Filter by statistics.fc.iops_raw .total  • Introduced in: 9.10
statistics.fc.iops_raw .write	integer	query	False	Filter by statistics.fc.iops_raw .write  • Introduced in: 9.10

Name	Type	In	Required	Description
statistics.fc.iops_raw.other	integer	query	False	Filter by statistics.fc.iops_raw.other  • Introduced in: 9.10
statistics.fc.iops_raw.read	integer	query	False	Filter by statistics.fc.iops_raw.read  • Introduced in: 9.10
statistics.fc.timestamp	string	query	False	Filter by statistics.fc.timestamp  • Introduced in: 9.10
statistics.fc.throughput_raw.read	integer	query	False	Filter by statistics.fc.throughput_raw.read  • Introduced in: 9.10
statistics.fc.throughput_raw.total	integer	query	False	Filter by statistics.fc.throughput_raw.total  • Introduced in: 9.10
statistics.fc.throughput_raw.write	integer	query	False	Filter by statistics.fc.throughput_raw.write  • Introduced in: 9.10
statistics.iops_raw.total	integer	query	False	Filter by statistics.iops_raw.total  • Introduced in: 9.7

Name	Type	In	Required	Description
statistics.iops_raw.write	integer	query	False	Filter by statistics.iops_raw.write  • Introduced in: 9.7
statistics.iops_raw.other	integer	query	False	Filter by statistics.iops_raw.other  • Introduced in: 9.7
statistics.iops_raw.read	integer	query	False	Filter by statistics.iops_raw.read  • Introduced in: 9.7
statistics.status	string	query	False	Filter by statistics.status  • Introduced in: 9.7
statistics.latency_raw.total	integer	query	False	Filter by statistics.latency_raw.total  • Introduced in: 9.7
statistics.latency_raw.write	integer	query	False	Filter by statistics.latency_raw.write  • Introduced in: 9.7
statistics.latency_raw.other	integer	query	False	Filter by statistics.latency_raw.other  • Introduced in: 9.7

Name	Type	In	Required	Description
statistics.latency_read	integer	query	False	Filter by statistics.latency_read  • Introduced in: 9.7
svm.uuid	string	query	False	Filter by svm.uuid
svm.name	string	query	False	Filter by svm.name
fields	array[string]	query	False	Specify the fields to return.
max_records	integer	query	False	Limit the number of records returned.
return_records	boolean	query	False	The default is true for GET calls. When set to false, only the number of records is returned.  • Default value: 1
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached.  • Default value: 1 • Max value: 120 • Min value: 0
order_by	array[string]	query	False	Order results by specified fields and optional [asc

## Response

Status: 200, Ok

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
num_records	integer	Number of records.
records	array[ <a href="#">nvme_service</a> ]	

## Example response

```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "metric": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "duration": "PT15S",
        "fc": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "duration": "PT15S",
          "iops": {
            "read": 200,
            "total": 1000,
            "write": 100
          },
          "latency": {
            "read": 200,
            "total": 1000,
            "write": 100
          },
          "status": "ok",
          "throughput": {
            "read": 200,
            "total": 1000,
```

```
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
"iops": {
  "read": 200,
  "total": 1000,
  "write": 100
},
"latency": {
  "read": 200,
  "total": 1000,
  "write": 100
},
"status": "ok",
"tcp": {
  "_links": {
    "self": {
      "href": "/api/resourceLink"
    }
  },
  "duration": "PT15S",
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
"throughput": {
  "read": 200,
  "total": 1000,
  "write": 100
},
"timestamp": "2017-01-25T11:20:13Z"
```

```

},
"statistics": {
  "fc": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25T11:20:13Z"
  },
  "iops_raw": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency_raw": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "tcp": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput_raw": {
      "read": 200,

```



```

        "total": 1000,
        "write": 100
    },
    "timestamp": "2017-01-25T11:20:13Z"
},
"throughput_raw": {
    "read": 200,
    "total": 1000,
    "write": 100
},
"timestamp": "2017-01-25T11:20:13Z"
},
"svm": {
    "_links": {
        "self": {
            "href": "/api/resourcelink"
        }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
}
]
}

```

## Error

Status: Default, Error

Name	Type	Description
error	error	

## Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

## See Definitions

href

Name	Type	Description
href	string	

\_links

Name	Type	Description
next	<a href="#">href</a>	
self	<a href="#">href</a>	

\_links

Name	Type	Description
self	<a href="#">href</a>	

iops

The rate of I/O operations observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

latency

The round trip latency in microseconds observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

### throughput

The rate of throughput bytes per second observed at the storage object.

Name	Type	Description
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

### fc

The NVMe/FC portion of the aggregated metrics.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.

Name	Type	Description
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

tcp

The NVMe/TCP portion of the aggregated metrics.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:

Name	Type	Description
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	

Name	Type	Description
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
fc	fc	The NVMe/FC portion of the aggregated metrics.
iops	iops	The rate of I/O operations observed at the storage object.
latency	latency	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
tcp	tcp	The NVMe/TCP portion of the aggregated metrics.
throughput	throughput	The rate of throughput bytes per second observed at the storage object.

Name	Type	Description
timestamp	string	The timestamp of the performance data.

#### iops\_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### latency\_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### throughput\_raw



Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

Name	Type	Description
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

fc

The NVMe/FC portion of the aggregated statistics.

Name	Type	Description
iops_raw	<a href="#">iops_raw</a>	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	<a href="#">latency_raw</a>	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

tcp

The NVMe/TCP portion of the aggregated statistics.

Name	Type	Description
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

#### statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

Name	Type	Description
fc	fc	The NVMe/FC portion of the aggregated statistics.
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	<p>Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data".</p> <p>"Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated.</p> <p>"Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.</p>
tcp	tcp	The NVMe/TCP portion of the aggregated statistics.

Name	Type	Description
throughput_raw	<a href="#">throughput_raw</a>	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

#### svm

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

#### nvme\_service

A Non-Volatile Memory Express (NVMe) service defines the properties of the NVMe controller target for an SVM. There can be at most one NVMe service for an SVM. An SVM's NVMe service must be created before NVMe host initiators can connect to the SVM.

An NVMe service is identified by the UUID of its SVM.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
enabled	boolean	The administrative state of the NVMe service. The NVMe service can be disabled to block all NVMe connectivity to the SVM.  This is optional in POST and PATCH. The default setting is <i>true</i> (enabled) in POST.
metric	<a href="#">metric</a>	Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
statistics	<a href="#">statistics</a>	These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.
svm	<a href="#">svm</a>	

#### error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

#### error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Create an NVMe service

POST /protocols/nvme/services

**Introduced In:** 9.6

Creates an NVMe service.

### Required properties

- `svm.uuid` or `svm.name` - The existing SVM in which to create the NVMe service.

### Related ONTAP commands

- `vserver nvme create`

## Learn more

- [DOC /protocols/nvme/services](#)

## Parameters

Name	Type	In	Required	Description
return_records	boolean	query	False	The default is false. If set to true, the records are returned. <ul style="list-style-type: none"><li>• Default value:</li></ul>

## Request Body

Name	Type	Description
_links	<a href="#">_links</a>	
enabled	boolean	The administrative state of the NVMe service. The NVMe service can be disabled to block all NVMe connectivity to the SVM.  This is optional in POST and PATCH. The default setting is <i>true</i> (enabled) in POST.
metric	<a href="#">metric</a>	Performance numbers, such as IOPS latency and throughput, for SVM protocols.
statistics	<a href="#">statistics</a>	These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.
svm	<a href="#">svm</a>	

## Example request

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "metric": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    }
  },
  "duration": "PT15S",
  "fc": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    }
  },
  "duration": "PT15S",
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
{iops": {
  "read": 200,
  "total": 1000,
  "write": 100
},
"latency": {
```



```

    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "tcp": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "duration": "PT15S",
    "iops": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25T11:20:13Z"
  },
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
"statistics": {
  "fc": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,

```

```
    "total": 1000,  
    "write": 100  
  },  
  "status": "ok",  
  "throughput_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "timestamp": "2017-01-25T11:20:13Z"  
},  
"iops_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},  
"latency_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},  
"status": "ok",  
"tcp": {  
  "iops_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "latency_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "status": "ok",  
  "throughput_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "timestamp": "2017-01-25T11:20:13Z"  
},  
"throughput_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},
```

```
    "timestamp": "2017-01-25T11:20:13Z"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
}
```

## Response

Status: 201, Created

Name	Type	Description
_links	<a href="#">_links</a>	
num_records	integer	Number of records.
records	array[ <a href="#">nvme_service</a> ]	

## Example response

```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "metric": {
        "_links": {
          "self": {
            "href": "/api/resourcelink"
          }
        },
        "duration": "PT15S",
        "fc": {
          "_links": {
            "self": {
              "href": "/api/resourcelink"
            }
          },
          "duration": "PT15S",
          "iops": {
            "read": 200,
            "total": 1000,
            "write": 100
          },
          "latency": {
            "read": 200,
            "total": 1000,
            "write": 100
          },
          "status": "ok",
          "throughput": {
            "read": 200,
            "total": 1000,
```

```
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
"iops": {
  "read": 200,
  "total": 1000,
  "write": 100
},
"latency": {
  "read": 200,
  "total": 1000,
  "write": 100
},
"status": "ok",
"tcp": {
  "_links": {
    "self": {
      "href": "/api/resourceLink"
    }
  },
  "duration": "PT15S",
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
"throughput": {
  "read": 200,
  "total": 1000,
  "write": 100
},
"timestamp": "2017-01-25T11:20:13Z"
```

```

},
"statistics": {
  "fc": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25T11:20:13Z"
  },
  "iops_raw": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency_raw": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "tcp": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput_raw": {
      "read": 200,

```

```

        "total": 1000,
        "write": 100
    },
    "timestamp": "2017-01-25T11:20:13Z"
},
"throughput_raw": {
    "read": 200,
    "total": 1000,
    "write": 100
},
"timestamp": "2017-01-25T11:20:13Z"
},
"svm": {
    "_links": {
        "self": {
            "href": "/api/resourcelink"
        }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
}
}
]
}

```

## Error

Status: Default

### ONTAP Error Response Codes

Error Code	Description
1115127	The cluster lacks a valid NVMe license.
2621462	The supplied SVM does not exist.
2621507	NVMe is not allowed for the specified SVM.
2621706	The specified <code>svm.uuid</code> and <code>svm.name</code> do not refer to the same SVM.
2621707	No SVM was specified. Either <code>svm.name</code> or <code>svm.uuid</code> must be supplied.
5374893	The SVM is stopped. The SVM must be running to create an NVMe service.

Error Code	Description
72089650	An NVMe service already exists for the specified SVM.
72089900	An NVMe service cannot be creating in an SVM that is configured for a SAN protocol.

Name	Type	Description
error	error	

### Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

### Definitions



## See Definitions

href

Name	Type	Description
href	string	

\_links

Name	Type	Description
self	<a href="#">href</a>	

iops

The rate of I/O operations observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

latency

The round trip latency in microseconds observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.

Name	Type	Description
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### throughput

The rate of throughput bytes per second observed at the storage object.

Name	Type	Description
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### fc

The NVMe/FC portion of the aggregated metrics.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.

Name	Type	Description
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

tcp

The NVMe/TCP portion of the aggregated metrics.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.

Name	Type	Description
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:

Name	Type	Description
fc	fc	The NVMe/FC portion of the aggregated metrics.
iops	iops	The rate of I/O operations observed at the storage object.
latency	latency	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
tcp	tcp	The NVMe/TCP portion of the aggregated metrics.
throughput	throughput	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

iops\_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### latency\_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### throughput\_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

Name	Type	Description
read	integer	Performance metric for read I/O operations.

Name	Type	Description
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

fc

The NVMe/FC portion of the aggregated statistics.

Name	Type	Description
iops_raw	<a href="#">iops_raw</a>	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	<a href="#">latency_raw</a>	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

tcp

The NVMe/TCP portion of the aggregated statistics.

Name	Type	Description
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.



Name	Type	Description
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

#### statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

Name	Type	Description
fc	fc	The NVMe/FC portion of the aggregated statistics.
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	<p>Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data".</p> <p>"Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated.</p> <p>"Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.</p>
tcp	tcp	The NVMe/TCP portion of the aggregated statistics.

Name	Type	Description
throughput_raw	<a href="#">throughput_raw</a>	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

#### svm

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

#### nvme\_service

A Non-Volatile Memory Express (NVMe) service defines the properties of the NVMe controller target for an SVM. There can be at most one NVMe service for an SVM. An SVM's NVMe service must be created before NVMe host initiators can connect to the SVM.

An NVMe service is identified by the UUID of its SVM.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
enabled	boolean	The administrative state of the NVMe service. The NVMe service can be disabled to block all NVMe connectivity to the SVM.  This is optional in POST and PATCH. The default setting is <i>true</i> (enabled) in POST.
metric	<a href="#">metric</a>	Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
statistics	<a href="#">statistics</a>	These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.
svm	<a href="#">svm</a>	

[\\_links](#)

Name	Type	Description
next	<a href="#">href</a>	
self	<a href="#">href</a>	

[error\\_arguments](#)

Name	Type	Description
code	string	Argument code
message	string	Message argument

[error](#)

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Delete an NVMe service

DELETE /protocols/nvme/services/{svm.uuid}

**Introduced In:** 9.6

Deletes an NVMe service. An NVMe service must be disabled before it can be deleted. In addition, all NVMe interfaces, subsystems, and subsystem maps associated with the SVM must first be deleted.

## Related ONTAP commands

- `vserver nvme delete`

## Learn more

- [DOC /protocols/nvme/services](#)

## Parameters

Name	Type	In	Required	Description
svm.uuid	string	path	True	The unique identifier of the SVM whose NVMe service is to be deleted.

## Response

Status: 200, Ok

## Error

Status: Default

## ONTAP Error Response Codes

Error Code	Description
2621462	The supplied SVM does not exist.
72089651	The supplied SVM does not have an NVMe service.
72089653	There are subsystems associated with the NVMe service SVM. The subsystems must be removed before deleting the NVMe service.
72089654	There are NVMe-oF LIFs associated with the NVMe service SVM. The LIFs must be removed before deleting the NVMe service.
72090028	The NVMe service is enabled. The NVMe service must be disabled before it can be deleted.

Name	Type	Description
error	<a href="#">error</a>	

## Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

### See Definitions

#### error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

#### error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

# Retrieve an NVMe service

GET /protocols/nvme/services/{svm.uuid}

Introduced In: 9.6

Retrieves an NVMe service.

## Related ONTAP commands

- `vserver nvme show`

## Learn more

- [DOC /protocols/nvme/services](#)

## Parameters

Name	Type	In	Required	Description
svm.uuid	string	path	True	The unique identifier of the SVM whose NVMe service is to be retrieved.
fields	array[string]	query	False	Specify the fields to return.

## Response

Status: 200, Ok

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
enabled	boolean	The administrative state of the NVMe service. The NVMe service can be disabled to block all NVMe connectivity to the SVM.  This is optional in POST and PATCH. The default setting is <i>true</i> (enabled) in POST.
metric	<a href="#">metric</a>	Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
statistics	<a href="#">statistics</a>	These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.
svm	<a href="#">svm</a>	



## Example response

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "metric": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    }
  },
  "duration": "PT15S",
  "fc": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    }
  },
  "duration": "PT15S",
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
```

```

    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "tcp": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "duration": "PT15S",
    "iops": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25T11:20:13Z"
  },
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
"statistics": {
  "fc": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,

```

```
    "total": 1000,  
    "write": 100  
  },  
  "status": "ok",  
  "throughput_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "timestamp": "2017-01-25T11:20:13Z"  
},  
"iops_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},  
"latency_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},  
"status": "ok",  
"tcp": {  
  "iops_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "latency_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "status": "ok",  
  "throughput_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "timestamp": "2017-01-25T11:20:13Z"  
},  
"throughput_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},
```

```

    "timestamp": "2017-01-25T11:20:13Z"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  }
}

```

## Error

Status: Default

### ONTAP Error Response Codes

Error Code	Description
2621462	The supplied SVM does not exist.
72089651	The supplied SVM does not have an NVMe service.

Name	Type	Description
error	error	

### Example error

```

{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}

```

# Definitions

## See Definitions

href

Name	Type	Description
href	string	

\_links

Name	Type	Description
self	<a href="#">href</a>	

iops

The rate of I/O operations observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

latency

The round trip latency in microseconds observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.

Name	Type	Description
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### throughput

The rate of throughput bytes per second observed at the storage object.

Name	Type	Description
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### fc

The NVMe/FC portion of the aggregated metrics.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.

Name	Type	Description
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

## tcp

The NVMe/TCP portion of the aggregated metrics.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.



Name	Type	Description
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:

Name	Type	Description
fc	fc	The NVMe/FC portion of the aggregated metrics.
iops	iops	The rate of I/O operations observed at the storage object.
latency	latency	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
tcp	tcp	The NVMe/TCP portion of the aggregated metrics.
throughput	throughput	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

#### iops\_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### latency\_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### throughput\_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

Name	Type	Description
read	integer	Performance metric for read I/O operations.

Name	Type	Description
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

fc

The NVMe/FC portion of the aggregated statistics.

Name	Type	Description
iops_raw	<a href="#">iops_raw</a>	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	<a href="#">latency_raw</a>	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

tcp

The NVMe/TCP portion of the aggregated statistics.

Name	Type	Description
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

#### statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

Name	Type	Description
fc	fc	The NVMe/FC portion of the aggregated statistics.
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	<p>Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data".</p> <p>"Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated.</p> <p>"Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.</p>
tcp	tcp	The NVMe/TCP portion of the aggregated statistics.

Name	Type	Description
throughput_raw	<a href="#">throughput_raw</a>	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

svm

Name	Type	Description
_links	<a href="#">_links</a>	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Update an NVMe service

PATCH /protocols/nvme/services/{svm.uuid}

Introduced In: 9.6



Updates an NVMe service.

## Related ONTAP commands

- `vserver nvme modify`

## Learn more

- [DOC /protocols/nvme/services](#)

## Parameters

Name	Type	In	Required	Description
svm.uuid	string	path	True	The unique identifier of the SVM whose NVMe service is to be updated.

## Request Body

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
enabled	boolean	The administrative state of the NVMe service. The NVMe service can be disabled to block all NVMe connectivity to the SVM.  This is optional in POST and PATCH. The default setting is <i>true</i> (enabled) in POST.
metric	<a href="#">metric</a>	Performance numbers, such as IOPS latency and throughput, for SVM protocols.
statistics	<a href="#">statistics</a>	These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.
svm	<a href="#">svm</a>	

## Example request

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "metric": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    }
  },
  "duration": "PT15S",
  "fc": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    }
  },
  "duration": "PT15S",
  "iops": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "latency": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
{iops": {
  "read": 200,
  "total": 1000,
  "write": 100
},
"latency": {
```

```

    "read": 200,
    "total": 1000,
    "write": 100
  },
  "status": "ok",
  "tcp": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "duration": "PT15S",
    "iops": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "timestamp": "2017-01-25T11:20:13Z"
  },
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
},
"statistics": {
  "fc": {
    "iops_raw": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency_raw": {
      "read": 200,

```

```
    "total": 1000,  
    "write": 100  
  },  
  "status": "ok",  
  "throughput_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "timestamp": "2017-01-25T11:20:13Z"  
},  
"iops_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},  
"latency_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},  
"status": "ok",  
"tcp": {  
  "iops_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "latency_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "status": "ok",  
  "throughput_raw": {  
    "read": 200,  
    "total": 1000,  
    "write": 100  
  },  
  "timestamp": "2017-01-25T11:20:13Z"  
},  
"throughput_raw": {  
  "read": 200,  
  "total": 1000,  
  "write": 100  
},
```

```

    "timestamp": "2017-01-25T11:20:13Z"
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    }
  },
  "name": "svm1",
  "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
}

```

## Response

Status: 200, Ok

## Error

Status: Default

### ONTAP Error Response Codes

Error Code	Description
1115127	The cluster lacks a valid NVMe license.
2621462	The supplied SVM does not exist.
5374893	The SVM is stopped. The SVM must be running to create an NVMe service.
72089651	The supplied SVM does not have an NVMe service.

Name	Type	Description
error	error	

## Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

## See Definitions

href

Name	Type	Description
href	string	

\_links

Name	Type	Description
self	<a href="#">href</a>	

iops

The rate of I/O operations observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

latency

The round trip latency in microseconds observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.

Name	Type	Description
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### throughput

The rate of throughput bytes per second observed at the storage object.

Name	Type	Description
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### fc

The NVMe/FC portion of the aggregated metrics.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.



Name	Type	Description
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

tcp

The NVMe/TCP portion of the aggregated metrics.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.

Name	Type	Description
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

metric

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:

Name	Type	Description
fc	fc	The NVMe/FC portion of the aggregated metrics.
iops	iops	The rate of I/O operations observed at the storage object.
latency	latency	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
tcp	tcp	The NVMe/TCP portion of the aggregated metrics.
throughput	throughput	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

iops\_raw

The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### latency\_raw

The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

#### throughput\_raw

Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.

Name	Type	Description
read	integer	Performance metric for read I/O operations.

Name	Type	Description
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

fc

The NVMe/FC portion of the aggregated statistics.

Name	Type	Description
iops_raw	<a href="#">iops_raw</a>	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	<a href="#">latency_raw</a>	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.

Name	Type	Description
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

tcp

The NVMe/TCP portion of the aggregated statistics.

Name	Type	Description
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.

Name	Type	Description
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput_raw	throughput_raw	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

#### statistics

These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.

Name	Type	Description
fc	fc	The NVMe/FC portion of the aggregated statistics.
iops_raw	iops_raw	The number of I/O operations observed at the storage object. This should be used along with delta time to calculate the rate of I/O operations per unit of time.
latency_raw	latency_raw	The raw latency in microseconds observed at the storage object. This should be divided by the raw IOPS value to calculate the average latency per I/O operation.
status	string	<p>Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data".</p> <p>"Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated.</p> <p>"Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.</p>
tcp	tcp	The NVMe/TCP portion of the aggregated statistics.



Name	Type	Description
throughput_raw	<a href="#">throughput_raw</a>	Throughput bytes observed at the storage object. This should be used along with delta time to calculate the rate of throughput bytes per unit of time.
timestamp	string	The timestamp of the performance data.

#### svm

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
name	string	The name of the SVM.
uuid	string	The unique identifier of the SVM.

#### nvme\_service

A Non-Volatile Memory Express (NVMe) service defines the properties of the NVMe controller target for an SVM. There can be at most one NVMe service for an SVM. An SVM's NVMe service must be created before NVMe host initiators can connect to the SVM.

An NVMe service is identified by the UUID of its SVM.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	
enabled	boolean	The administrative state of the NVMe service. The NVMe service can be disabled to block all NVMe connectivity to the SVM.  This is optional in POST and PATCH. The default setting is <i>true</i> (enabled) in POST.
metric	<a href="#">metric</a>	Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
statistics	<a href="#">statistics</a>	These are raw performance numbers, such as IOPS latency and throughput for SVM protocols. These numbers are aggregated across all nodes in the cluster and increase with the uptime of the cluster.
svm	<a href="#">svm</a>	

#### error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

#### error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Retrieve NVMe protocol historical performance metrics

GET /protocols/nvme/services/{svm.uuid}/metrics

**Introduced In:** 9.7

Retrieves historical performance metrics for NVMe protocol of an SVM.

### Parameters

Name	Type	In	Required	Description
fc.status	string	query	False	Filter by fc.status  • Introduced in: 9.10
fc.iops.total	integer	query	False	Filter by fc.iops.total  • Introduced in: 9.10
fc.iops.write	integer	query	False	Filter by fc.iops.write  • Introduced in: 9.10
fc.iops.other	integer	query	False	Filter by fc.iops.other  • Introduced in: 9.10
fc.iops.read	integer	query	False	Filter by fc.iops.read  • Introduced in: 9.10
fc.throughput.read	integer	query	False	Filter by fc.throughput.read  • Introduced in: 9.10
fc.throughput.total	integer	query	False	Filter by fc.throughput.total  • Introduced in: 9.10
fc.throughput.write	integer	query	False	Filter by fc.throughput.write  • Introduced in: 9.10
fc.duration	string	query	False	Filter by fc.duration  • Introduced in: 9.10

Name	Type	In	Required	Description
fc.latency.total	integer	query	False	Filter by fc.latency.total  • Introduced in: 9.10
fc.latency.write	integer	query	False	Filter by fc.latency.write  • Introduced in: 9.10
fc.latency.other	integer	query	False	Filter by fc.latency.other  • Introduced in: 9.10
fc.latency.read	integer	query	False	Filter by fc.latency.read  • Introduced in: 9.10
status	string	query	False	Filter by status
timestamp	string	query	False	Filter by timestamp
latency.total	integer	query	False	Filter by latency.total
latency.write	integer	query	False	Filter by latency.write
latency.other	integer	query	False	Filter by latency.other
latency.read	integer	query	False	Filter by latency.read
iops.total	integer	query	False	Filter by iops.total
iops.write	integer	query	False	Filter by iops.write
iops.other	integer	query	False	Filter by iops.other
iops.read	integer	query	False	Filter by iops.read

Name	Type	In	Required	Description
throughput.read	integer	query	False	Filter by throughput.read
throughput.total	integer	query	False	Filter by throughput.total
throughput.write	integer	query	False	Filter by throughput.write
tcp.status	string	query	False	Filter by tcp.status  • Introduced in: 9.10
tcp.iops.total	integer	query	False	Filter by tcp.iops.total  • Introduced in: 9.10
tcp.iops.write	integer	query	False	Filter by tcp.iops.write  • Introduced in: 9.10
tcp.iops.other	integer	query	False	Filter by tcp.iops.other  • Introduced in: 9.10
tcp.iops.read	integer	query	False	Filter by tcp.iops.read  • Introduced in: 9.10
tcp.throughput.read	integer	query	False	Filter by tcp.throughput.read  • Introduced in: 9.10

Name	Type	In	Required	Description
tcp.throughput.total	integer	query	False	Filter by tcp.throughput.total  • Introduced in: 9.10
tcp.throughput.write	integer	query	False	Filter by tcp.throughput.write  • Introduced in: 9.10
tcp.duration	string	query	False	Filter by tcp.duration  • Introduced in: 9.10
tcp.latency.total	integer	query	False	Filter by tcp.latency.total  • Introduced in: 9.10
tcp.latency.write	integer	query	False	Filter by tcp.latency.write  • Introduced in: 9.10
tcp.latency.other	integer	query	False	Filter by tcp.latency.other  • Introduced in: 9.10
tcp.latency.read	integer	query	False	Filter by tcp.latency.read  • Introduced in: 9.10
duration	string	query	False	Filter by duration
svm.uuid	string	path	True	Unique identifier of the SVM.

Name	Type	In	Required	Description
interval	string	query	False	<p>The time range for the data. Examples can be 1h, 1d, 1m, 1w, 1y. The period for each time range is as follows:</p> <ul style="list-style-type: none"> <li>• 1h: Metrics over the most recent hour sampled over 15 seconds.</li> <li>• 1d: Metrics over the most recent day sampled over 5 minutes.</li> <li>• 1w: Metrics over the most recent week sampled over 30 minutes.</li> <li>• 1m: Metrics over the most recent month sampled over 2 hours.</li> <li>• 1y: Metrics over the most recent year sampled over a day.</li> <li>• Default value: 1</li> <li>• enum: ["1h", "1d", "1w", "1m", "1y"]</li> </ul>

Name	Type	In	Required	Description
return_timeout	integer	query	False	The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached. <ul style="list-style-type: none"> <li>• Default value: 1</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul>
fields	array[string]	query	False	Specify the fields to return.
max_records	integer	query	False	Limit the number of records returned.
order_by	array[string]	query	False	Order results by specified fields and optional [asc
desc] direction. Default direction is 'asc' for ascending.	return_records	boolean	query	False

## Response

Status: 200, Ok

Name	Type	Description
_links	<a href="#">_links</a>	
num_records	integer	Number of records
records	array[ <a href="#">records</a> ]	



## Example response

```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "records": [
    {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "duration": "PT15S",
      "fc": {
        "duration": "PT15S",
        "iops": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "latency": {
          "read": 200,
          "total": 1000,
          "write": 100
        },
        "status": "ok",
        "throughput": {
          "read": 200,
          "total": 1000,
          "write": 100
        }
      },
      "iops": {
        "read": 200,
        "total": 1000,
        "write": 100
      },
      "latency": {
        "read": 200,
        "total": 1000,
```

```

    "write": 100
  },
  "status": "ok",
  "svm": {
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "tcp": {
    "duration": "PT15S",
    "iops": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "latency": {
      "read": 200,
      "total": 1000,
      "write": 100
    },
    "status": "ok",
    "throughput": {
      "read": 200,
      "total": 1000,
      "write": 100
    }
  },
  "throughput": {
    "read": 200,
    "total": 1000,
    "write": 100
  },
  "timestamp": "2017-01-25T11:20:13Z"
}
]
}

```

## Error

Status: Default, Error

Name	Type	Description
error	error	

## Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

## See Definitions

href

Name	Type	Description
href	string	

\_links

Name	Type	Description
next	<a href="#">href</a>	
self	<a href="#">href</a>	

\_links

Name	Type	Description
self	<a href="#">href</a>	

iops

The rate of I/O operations observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

latency

The round trip latency in microseconds observed at the storage object.

Name	Type	Description
other	integer	Performance metric for other I/O operations. Other I/O operations can be metadata operations, such as directory lookups and so on.
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

### throughput

The rate of throughput bytes per second observed at the storage object.

Name	Type	Description
read	integer	Performance metric for read I/O operations.
total	integer	Performance metric aggregated over all types of I/O operations.
write	integer	Performance metric for write I/O operations.

### fc

The NVMe/FC portion of the aggregated metrics.

Name	Type	Description
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.

Name	Type	Description
latency	latency	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	throughput	The rate of throughput bytes per second observed at the storage object.

#### svm

Name	Type	Description
uuid	string	The unique identifier of the SVM.

#### tcp

The NVMe/TCP portion of the aggregated metrics.

Name	Type	Description
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
iops	<a href="#">iops</a>	The rate of I/O operations observed at the storage object.
latency	<a href="#">latency</a>	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.

records

Performance numbers, such as IOPS latency and throughput, for SVM protocols.

Name	Type	Description
<a href="#">_links</a>	<a href="#">_links</a>	

Name	Type	Description
duration	string	The duration over which this sample is calculated. The time durations are represented in the ISO-8601 standard format. Samples can be calculated over the following durations:
fc	fc	The NVMe/FC portion of the aggregated metrics.
iops	iops	The rate of I/O operations observed at the storage object.
latency	latency	The round trip latency in microseconds observed at the storage object.
status	string	Any errors associated with the sample. For example, if the aggregation of data over multiple nodes fails then any of the partial errors might be returned, "ok" on success, or "error" on any internal uncategorized failure. Whenever a sample collection is missed but done at a later time, it is back filled to the previous 15 second timestamp and tagged with "backfilled_data". "Inconsistent_delta_time" is encountered when the time between two collections is not the same for all nodes. Therefore, the aggregated value might be over or under inflated. "Negative_delta" is returned when an expected monotonically increasing value has decreased in value. "Inconsistent_old_data" is returned when one or more nodes do not have the latest data.
svm	svm	
tcp	tcp	The NVMe/TCP portion of the aggregated metrics.



Name	Type	Description
throughput	<a href="#">throughput</a>	The rate of throughput bytes per second observed at the storage object.
timestamp	string	The timestamp of the performance data.

#### error\_arguments

Name	Type	Description
code	string	Argument code
message	string	Message argument

#### error

Name	Type	Description
arguments	array[ <a href="#">error_arguments</a> ]	Message arguments
code	string	Error code
message	string	Error message
target	string	The target parameter that caused the error.

## Copyright information

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, 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 OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.