



# Initiator API methods

## Element Software

NetApp  
March 05, 2025

# Table of Contents

- Initiator API methods ..... 1
  - Find more information ..... 1
  - CreateInitiators ..... 1
    - Parameter ..... 1
    - Return value ..... 4
    - Error ..... 4
    - Request example ..... 4
    - Response example ..... 4
    - New since version ..... 5
    - Find more information ..... 5
  - DeleteInitiators ..... 5
    - Parameter ..... 5
    - Return values ..... 6
    - Error ..... 6
    - Request example ..... 6
    - Response example ..... 6
    - New since version ..... 6
  - ListInitiators ..... 6
    - Parameters ..... 7
    - Return value ..... 7
    - Exceptions ..... 7
    - Request example ..... 7
    - Response example ..... 8
    - New since version ..... 8
  - ModifyInitiators ..... 8
    - Parameter ..... 8
    - Return value ..... 11
    - Request example ..... 11
    - Response example ..... 11
    - New since version ..... 12
    - Find more information ..... 12

# Initiator API methods

Initiator methods enable you to add, remove, view, and modify iSCSI initiator objects, which handle communication between the storage system and external storage clients.

- [CreateInitiators](#)
- [DeleteInitiators](#)
- [ListInitiators](#)
- [ModifyInitiators](#)

## Find more information

- [SolidFire and Element Software Documentation](#)
- [Documentation for earlier versions of NetApp SolidFire and Element products](#)

## CreateInitiators

You can use `CreateInitiators` to create multiple new initiator IQNs or World Wide Port Names (WWPNs) and optionally assign them aliases and attributes. When you use `CreateInitiators` to create new initiators, you can also add them to volume access groups.

If the operation fails to create one of the initiators provided in the parameter, the method returns an error and does not create any initiators (no partial completion is possible).

### Parameter

This method has the following input parameter:



Name	Description	Type	Default value	Required
initiators	<p>A list of objects containing characteristics of each new initiator. Objects:</p> <ul style="list-style-type: none"> <li>• <b>alias:</b> (Optional) The friendly name to assign to this initiator. (string)</li> <li>• <b>attributes:</b> (Optional) A set of JSON attributes to assign to this initiator. (JSON object)</li> <li>• <b>chapUsername:</b> (Optional) The unique CHAP username for this initiator. Defaults to the initiator name (IQN) if not specified during creation and <code>requireChap</code> is true. (string)</li> <li>• <b>initiatorSecret:</b> (Optional) The CHAP secret used to authenticate the initiator. Defaults to a randomly generated secret if not specified during creation and <code>requireChap</code> is true. (string)</li> <li>• <b>name:</b> (Required) The name of the initiator (IQN or WWPN) to create. (string)</li> <li>• <b>requireChap:</b> (Optional) True if CHAP is required during session login for</li> </ul>	JSON object array	None	Yes

## Return value

This method has the following return value:  
(boolean)

Name	Description	Type
initiators	List of objects describing the newly created initiators.	initiator array

## Error

This method can return the following error:  
Defaults to a

Name	Description
xInitiatorExists	Returned if the initiator name you chose already exists.

## Request example

Requests for this method are similar to the following example:  
(Optional) The

```
{
  "id": 3291,
  "method": "CreateInitiators",
  "params": {
    "initiators": [
      {
        "name": "iqn.1993-08.org.debian:01:288170452",
        "alias": "example1"
      },
      {
        "name": "iqn.1993-08.org.debian:01:297817012",
        "alias": "example2"
      }
    ]
  }
}
```

## Response example

This method returns a response similar to the following example:  
(integer)

```

{
  "id": 3291,
  "result": {
    "initiators": [
      {
        "alias": "example1",
        "attributes": {},
        "initiatorID": 145,
        "initiatorName": "iqn.1993-08.org.debian:01:288170452",
        "volumeAccessGroups": []
      },
      {
        "alias": "example2",
        "attributes": {},
        "initiatorID": 146,
        "initiatorName": "iqn.1993-08.org.debian:01:297817012",
        "volumeAccessGroups": []
      }
    ]
  }
}

```

## New since version

9.6

## Find more information

[ListInitiators](#)

## DeleteInitiators

You can use `DeleteInitiators` to delete one or more initiators from the system (and from any associated volumes or volume access groups).

If `DeleteInitiators` fails to delete one of the initiators provided in the parameter, the system returns an error and does not delete any initiators (no partial completion is possible).

## Parameter

This method has the following input parameter:

Name	Description	Type	Default value	Required
initiators	An array of IDs of initiators to delete.	integer array	None	Yes

## Return values

This method has no return values.

## Error

This method can return the following error:

Name	Description
xInitiatorDoesNotExist	Returned if the initiator name you choose does not exist.

## Request example

Requests for this method are similar to the following example:

```
{
  "id": 5101,
  "method": "DeleteInitiators",
  "params": {
    "initiators": [
      145,
      147
    ]
  }
}
```

## Response example

This method returns a response similar to the following example:

```
{
  "id": 5101,
  "result": {}
}
```

## New since version

9.6

## ListInitiators

You can use the `ListInitiators` method to get the list of initiator IQNs or World Wide Port Names (WWPNs).



## Parameters

This method has the following input parameters:

Name	Description	Type	Default value	Required
initiators	A list of initiator IDs to retrieve. You can supply this parameter or the startInitiatorID parameter, but not both.	integer array	None	No
startInitiatorID	The initiator ID at which to begin the listing. You can supply this parameter or the initiators parameter, but not both.	integer	0	No
limit	The maximum number of initiator objects to return.	integer	(unlimited)	No

## Return value

This method has the following return value:

Name	Description	Type
initiators	List of the initiator information.	<a href="#">initiator</a> array

## Exceptions

This method can have the following exception:

Name	Description
xInvalidParameter	Thrown if you include both the startInitiatorID and the initiators parameters in the same method call.

## Request example

Requests for this method are similar to the following example:

```
{
  "method": "ListInitiators",
  "params": {},
  "id" : 1
}
```

## Response example

This method returns a response similar to the following example:

```
{
  "id": 1,
  "result": {
    "initiators": [
      {
        "alias": "",
        "attributes": {},
        "initiatorID": 2,
        "initiatorName": "iqn.1993-08.org.debian:01:c84ffd71216",
        "volumeAccessGroups": [
          1
        ]
      }
    ]
  }
}
```

## New since version

9.6

## ModifyInitiators

You can use the `ModifyInitiators` method to change the attributes of one or more existing initiators.

You cannot change the name of an existing initiator. If you need to change the name of an initiator, delete it first with the [DeleteInitiators](#) method and create a new one with the [CreateInitiators](#) method.

If `ModifyInitiators` fails to change one of the initiators provided in the parameter, the method returns an error and does not modify any initiators (no partial completion is possible).

## Parameter

This method has the following input parameter:

Name	Description	Type	Default value	Required
------	-------------	------	---------------	----------

<p>initiators</p>	<p>A list of objects containing characteristics of each initiator to modify. Possible objects:</p> <ul style="list-style-type: none"> <li>• alias: (Optional) A new friendly name to assign to the initiator. (string)</li> <li>• attributes: (Optional) A new set of JSON attributes to assign to the initiator. (JSON object)</li> <li>• chapUsername: (Optional) A new unique CHAP username for this initiator. (string)</li> <li>• forceDuringUpgrade: Complete the initiator modification during an upgrade.</li> <li>• initiatorID: (Required) The ID of the initiator to modify. (integer)</li> <li>• initiatorSecret: (Optional) A new CHAP secret used to authenticate the initiator. (string)</li> <li>• requireChap: (Optional) True if CHAP is required for this initiator. (boolean)</li> <li>• targetSecret: (Optional) A new CHAP secret used to authenticate the</li> </ul>	<p>JSON object array</p>	<p>None</p>	<p>Yes</p>
-------------------	---	--------------------------	-------------	------------

## Return value

This method has the following return value:

Name	Description	Type
initiators	List of objects describing the newly modified initiators.	initiator array

## Request example

Requests for this method are similar to the following example:

```
{
  "id": 6683,
  "method": "ModifyInitiators",
  "params": {
    "initiators": [
      {
        "initiatorID": 2,
        "alias": "alias1",
        "volumeAccessGroupID": null
      },
      {
        "initiatorID": 3,
        "alias": "alias2",
        "volumeAccessGroupID": 1
      }
    ]
  }
}
```

## Response example

This method returns a response similar to the following example:

access group, it is removed from the old volume access group. If this key is present but null, the initiator is removed from its current volume access group, but not placed in any new volume access group. (integer)

```
{
  "id": 6683,
  "result": {
    "initiators": [
      {
        "alias": "alias1",
        "attributes": {},
        "initiatorID": 2,
        "initiatorName": "iqn.1993-08.org.debian:01:395543635",
        "volumeAccessGroups": []
      },
      {
        "alias": "alias2",
        "attributes": {},
        "initiatorID": 3,
        "initiatorName": "iqn.1993-08.org.debian:01:935573135",
        "volumeAccessGroups": [
          1
        ]
      }
    ]
  }
}
```

## New since version

9.6

## Find more information

- [CreateInitiators](#)
- [DeleteInitiators](#)

## 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.