Cluster creation API methods

Element Software

NetApp
November 18, 2025

Table of Contents

Cluster creation API methods

CheckProposedCluster

You can use the `CheckProposedCluster` method to test a set of storage nodes before creating a storage cluster with them to identify possible errors or faults that would occur from the attempt, such as unbalanced mixed node capabilities or node types that are not supported for two-node storage clusters.

Parameters

This method has the following input parameter:

Name	Description	Type	Default value	Required
nodes	A list of storage IP addresses of the initial set of storage nodes making up the storage cluster.	string array	None	Yes
force	Set to true to run on all storage nodes in the storage cluster.	boolean	None	No

Return values

This method has the following return values:

Name	Description	Type
proposedClusterValid	Indicates whether or not the proposed storage nodes would make up a valid storage cluster. Possible values: • true • false	boolean
proposedClusterErrors	Errors that would occur if a storage cluster was created using the proposed storage nodes.	string array

Request example

Requests for this method are similar to the following example:

```
{
    "method": "CheckProposedCluster",
    "params": {
    "nodes": [
        "192.168.1.11",
        "192.168.1.12",
        "192.168.1.13",
        "192.168.1.14"
    ]
},
    "id": 1
}
```

Response example

This method returns a response similar to the following example:

```
{
    "id": 1,
    "result": {
        "proposedClusterValid": true,
        "proposedClusterErrors": [ ]
    }
}
```

New since version

11.0

CreateCluster

You can use the `CreateCluster` method to initialize the node in a cluster that has ownership of the "mvip" and "svip" addresses. Each new cluster is initialized using the management IP (MIP) of the first node in the cluster. This method also automatically adds all the nodes being configured into the cluster. You only need to use this method once each time a new cluster is initialized.

> After you log in to the master node for the cluster and run the GetBootStrapConfig method to get the IP addresses for the rest of the nodes that you want to include in the cluster, you can run the CreateCluster method against the master node for the cluster.

Parameters

This method has the following input parameters:

Name	Description	Type	Default value	Required
acceptEula	Indicate your acceptance of the End User License Agreement when creating this cluster. To accept the EULA, set this parameter to true.	boolean	None	Yes
attributes	List of name-value pairs in JSON object format.	JSON object	None	No
enableSoftwareEncryptionAtRest	Enable this parameter to use software-based encryption at rest. Defaults to false on all clusters. After software encryption at rest is enabled, it cannot be disabled on the cluster.	boolean	true	No
mvip	Floating (virtual) IP address for the cluster on the management network.	string	None	Yes
nodes	CIP/SIP addresses of the initial set of nodes making up the cluster. This node's IP must be in the list.	string array	None	Yes
orderNumber	Alphanumeric sales order number. Required on software-based platforms.	string	None	No (hardware-based platforms) Yes (software-based platforms)
password	Initial password for the cluster admin account.	string	None	Yes

Name	Description	Type	Default value	Required
serialNumber	Nine-digit alphanumeric Serial Number. May be required on software-based platforms.	string	None	No (hardware-based platforms) Yes (software-based platforms)
svip	Floating (virtual) IP address for the cluster on the storage (iSCSI) network.	string	None	Yes
username	User name for the cluster admin.	string	None	Yes

Return values

This method has no return values.

Request example

Requests for this method are similar to the following example:

```
{
  "method": "CreateCluster",
  "params": {
    "acceptEula": true,
    "mvip": "10.0.3.1",
    "svip": "10.0.4.1",
    "username": "Admin1",
    "password": "9R7ka4rEPa2uREtE",
    "attributes": {
      "clusteraccountnumber": "axdf323456"
    },
    "nodes": [
      "10.0.2.1",
      "10.0.2.2",
      "10.0.2.3",
      "10.0.2.4"
    ]
  },
  "id": 1
}
```

Response example

This method returns a response similar to the following example:

```
{
"id" : 1,
"result" : {}
}
```

New since version

9.6

Find more information

- GetBootstrapConfig
- SolidFire and Element Software Documentation
- Documentation for earlier versions of NetApp SolidFire and Element products

GetBootstrapConfig

You can use the `GetBootstrapConfig` method to get cluster and node information from the bootstrap configuration file. Use this API method on an individual node before it has been joined with a cluster. The information this method returns is used in the cluster configuration interface when you create a cluster.

Parameters

This method has no input parameters.

Return values

This method has the following return values:

Name	Description	Type
clusterName	Name of the cluster.	string
mvip	Cluster MVIP address. Empty if the node is not part of a cluster.	string
nodeName	Name of the node.	string

Name	Description	Type
nodes	List of information about each node that is actively waiting to join the cluster. Possible values: • chassisType: (string) Hardware platform of the node. • cip: (string) Cluster IP address of the node. • compatible: (boolean) Indicates if the node is compatible with the node the API call was executed against. • hostname: (string) Host name of the node. • mip: (string) The IPv4 management IP address of the node. • mipV6: (string) The IPv6 management IP address of the node. • nodeType: (string)Model name of the node. • version: (string)Version of software currently installed on the node.	JSON object array
svip	Cluster SVIP address. Null if the node is not part of a cluster.	string
version	Version of Element software currently installed on the node that was called by this API method.	string

Request example

Requests for this method are similar to the following example:

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

Response example

This method returns a response similar to the following example:

```
{
    "id":1,
    "result":{
        "clusterName":"testname",
        "nodeName":"testnode",
        "svip": "10.117.1.5",
        "mvip": "10.117.1.6",
        "nodes":[
            {
                "chassisType":"R630",
                "cip":"10.117.115.16",
                "compatible":true,
                "hostname":"NLABP1132",
                "mip":"10.117.114.16",
                "mipV6":"fd20:8b1e:b256:45a::16",
                "nodeType":"SF2405",
                "role":"Storage",
                "version":"11.0"
            },
            {
                "chassisType":"R630",
                "cip":"10.117.115.17",
                "compatible":true,
                "hostname":"NLABP1133",
                "mip":"10.117.114.17",
                "mipV6":"fd20:8b1e:b256:45a::17",
                "nodeType":"SF2405",
                "role":"Storage",
                "version":"11.0"
            },
            {
                "chassisType":"R630",
                "cip":"10.117.115.18",
                "compatible":true,
                "hostname":"NLABP1134",
                "mip":"10.117.114.18",
                "mipV6":"fd20:8b1e:b256:45a::18",
                "nodeType":"SF2405",
                "role":"Storage",
                "version":"11.0"
            }
        ],
        "version":"11.0"
    }
}
```

New since version

9.6

Find more information

CreateCluster

Copyright information

Trademark information