Element API software

Element Software

NetApp
November 18, 2025

Table of Contents

Element API software . . . 1
- Learn about managing storage with Element API . . . 1
  - Common objects . . . 1
  - Common methods . . . 1
  - Account API methods . . . 1
  - Administrator API methods . . . 1
  - Cluster API methods . . . 2
  - Cluster creation API methods . . . 2
  - Drive API methods . . . 2
  - Fibre Channel API methods . . . 2
  - Initiator API methods . . . 2
  - LDAP API methods . . . 2
  - Multi-factor authentication API methods . . . 2
  - Session authentication API methods . . . 3
  - Node API methods . . . 3
  - Replication API methods . . . 3
  - Security API methods . . . 3
  - SnapMirror API methods . . . 3
  - System configuration API methods . . . 3
  - Multitenant networking API methods . . . 4
  - Volume API methods . . . 4
  - Volume access group API methods . . . 4
  - Volume snapshot API methods . . . 4
  - Virtual volume API methods . . . 5
  - Find more information . . . 5
- Request object members . . . 5
- Response object members . . . 5
- Request endpoints . . . 6
  - Cluster API methods . . . 6
  - Cluster creation and bootstrap API methods . . . 6
  - Per-node API methods . . . 7
  - Find more information . . . 7
- API authentication . . . 7
  - Find more information . . . 7
- Asynchronous methods . . . 7
  - Find more information . . . 8
- Attributes . . . 8
  - Object member . . . 8
  - Request example . . . 8

Element API software

Learn about managing storage with Element API

The Element API is based on the JSON-RPC protocol over HTTPS. JSON-RPC is a simple text-based RPC protocol based on the lightweight JSON data-interchange format. Client libraries are available for all major programming languages.

You can make API requests via HTTPS POST requests to the API endpoint. The body of the POST request is a JSON-RPC request object. The API does not currently support batch requests (multiple request objects in a single POST). When submitting API requests, you must use "application/json-rpc" as the content-type of the request, and ensure that the body is not form-encoded.

The Element web UI makes use of the API methods described in this document. You can monitor API operations in the UI by enabling the API Log; this enables you to see the methods that are being issued to the system. You can enable both requests and responses to see how the system replies to the methods that are issued.

Unless stated otherwise, all date strings in the API responses are in UTC+0 format.

When the storage cluster is heavily loaded or you submit many consecutive API requests with no intervening delay, a method might fail and return the error "xDBVersionMismatch". If this happens, retry the method call.

Common objects

The Element software API uses JSON objects to represent organized data concepts. Many of these API methods make use of these objects for data input and output. This section documents these commonly used objects; objects that are only used within a single method are documented with that method instead of in this section.

Learn about common objects

Common methods

Common methods are methods used to retrieve information about the storage cluster, the API itself, or ongoing API operations.

Learn about common methods

Account API methods

Account methods enable you to add, remove, view, and modify account and security information.

Learn about account API methods

Administrator API methods

You can use administrator API methods to create, modify, view, and remove storage cluster administrators and assign levels of access and privileges for those with access to a storage cluster.

Learn about administrator API methods

Cluster API methods

Element software cluster API methods enable you to manage the configuration and topology of the storage cluster and the nodes that belong to a storage cluster.

Some cluster API methods operate on nodes that are part of a cluster, or have been configured to join a cluster. You can add nodes to a new cluster or to an existing cluster. Nodes that are ready to be added to a cluster are in a "pending" state, which means they have been configured but not yet added to the cluster.

Learn about cluster API methods

Cluster creation API methods

You can use these API methods to create a storage cluster. All of these methods need to be used against the API endpoint on a single node.

Learn about cluster creation API methods

Drive API methods

You can use drive API methods to add and manage drives that are available to a storage cluster. When you add a storage node to the storage cluster or install new drives in an existing storage node, the drives are available to be added to the storage cluster.

Learn about drive API methods

Fibre Channel API methods

You can use Fibre Channel API methods to add, modify, or remove Fibre Channel node members of a storage cluster.

Learn about Fibre Channel API methods

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.

Learn about initiator API methods

LDAP API methods

You can use the Lightweight Directory Access Protocol (LDAP) to authenticate access to Element storage. The LDAP API methods described in this section enable you to configure LDAP access to the storage cluster.

Learn about LDAP API methods

Multi-factor authentication API methods

You can use multi-factor authentication (MFA) to manage user sessions using a third-party Identity Provider (IdP) via the Security Assertion Markup Language (SAML).

Learn about multi-factor authentication API methods

Session authentication API methods

You can use session-based authentication to manage user sessions.

Learn about session authentication API methods

Node API methods

You can use node API methods to configure individual nodes. These methods operate on single nodes that need to be configured, are configured but not yet participating in a cluster, or are actively participating in a cluster. Node API methods enable you to view and modify settings for individual nodes and the cluster network used to communicate with the node. You must run these methods against individual nodes; you cannot run per-node API methods against the address of the cluster.

Learn about node API methods

Replication API methods

Replication API methods enable you to connect two clusters for continuous data protection (CDP). When you connect two clusters, active volumes within a cluster can be continuously replicated to a second cluster to provide data recovery. By pairing volumes for replication, you can protect your data from events that might render it inaccessible.

Learn about replication API methods

Security API methods

You can integrate Element software with external security-related services, such as an external key management server. These security-related methods enable you to configure Element security features such as external key management for Encryption at Rest.

Learn about security API methods

SnapMirror API methods

SnapMirror API methods are used by the Element web UI for managing snapshots mirrored with remote ONTAP systems. These methods are meant for use by the Element web UI only. If you need API access to SnapMirror functionality, use the ONTAP APIs. Request and return examples are not provided for SnapMirror API methods.

Learn about SnapMirror API methods

System configuration API methods

System configuration API methods enable you to obtain and set configuration values that apply to all nodes in the cluster.

Learn about system configuration API methods

Multitenant networking API methods

Multitenant networking in Element storage clusters allows traffic between multiple clients that are on separate logical networks to be connected to one Element storage cluster without layer 3 routing.

Connections to the storage cluster are segregated in the networking stack through the use of VLAN tagging.

Prerequisites for setting up a multitenant virtual network

- You must have identified the block of client network IP addresses to be assigned to the virtual networks on the storage nodes.
- You must have identified a client storage network IP (SVIP) address to be used as an endpoint for all storage traffic.

Virtual networking order of operations

1. Use the AddVirtualNetwork method to bulk provision the IP addresses you enter.

   After you add a virtual network, the cluster automatically performs the following steps:

   - Each storage node creates a virtual network interface.
   - Each storage node is assigned a VLAN address that can be routed to using the virtual SVIP.
   - VLAN IP addresses persist on each node in the event of a node reboot.
2. When the virtual network interface and VLAN addresses have been assigned, you can assign client network traffic to the virtual SVIP.

Learn about multitenant networking API methods

Volume API methods

Element software volume API methods enable you to manage volumes that reside on a storage node. You can create, modify, clone, and delete volumes with these methods. You can also use volume API methods to gather and display data measurements for a volume.

Learn about volume API methods

Volume access group API methods

Volume access group methods enable you to add, remove, view, and modify volume access groups, which are collections of volumes that users can access using either iSCSI or Fibre Channel initiators.

Learn about volume access group API methods

Volume snapshot API methods

Element software volume snapshot API methods enable you to manage volume snapshots. You can create, modify, clone, and delete volume snapshots using the volume snapshot API methods.

Learn about volume snapshot API methods

Virtual volume API methods

Element software virtual volume API methods enable you to manage virtual volumes (VVols). You can view existing VVols with these API methods as well as create, modify, and delete virtual volume storage containers. Although you cannot use these methods to operate on normal volumes, you can use the normal volume API methods to list information about VVols.

Learn about virtual volume API methods

Find more information

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

Request object members

Each Element software API request has the following basic parts:

Name	Description	Type	Default value	Required
method	Name of the method to be invoked.	string	None	Yes
parameters	Object containing the parameters to the method being invoked. Named parameters are required. Positional parameters (passed as an array) are not allowed.	JSON object	{}	No
id	Identifier used to match the request to response, returned in the result.	string or integer	{}	No

Response object members

Each Element software API response body has the following basic parts:

Name	Description	Type
result	The object returned by the method. The system returns an object with named members corresponding to the documented return value for the method. This member is not present if an error has occurred.	JSON object
error	The object returned when an error occurs. This member is present only if an error has occurred.	Object
id	An identifier used to match the request to response, as provided in the request.	string or integer
unusedParameters	A warning message that at least one incorrect parameter has been passed to the API method and has not been used.	Object

Request endpoints

There are three types of request endpoints used in the API (storage cluster, storage cluster creation, and per-node). You should always use the latest endpoint supported by your version of Element software.

The three request endpoints in the API are designated in the following ways:

Cluster API methods

The HTTPS endpoint for storage-cluster-wide API requests is `https://<mvip>/json-rpc/<api-version>`, where:

- `<mvip>` is the management virtual IP address for the storage cluster.
- `<api-version>` is the version of the API you are using.

Cluster creation and bootstrap API methods

The HTTPS endpoint for creating a storage cluster and accessing bootstrap API requests is `https://<nodeIP>/json-rpc/<api-version>`, where:

- `<nodeIP>` is the IP address of the node you are adding to the cluster.
- `<api-version>` is the version of the API you are using.

Per-node API methods

The HTTPS endpoint for individual storage node API requests is `https://<nodeIP>:442/json-rpc/<api-version>`, where:

- `<nodeIP>` is the management IP address of the storage node; 442 is the port the HTTPS server is running on.
- `<api-version>` is the version of the API you are using.

Find more information

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

API authentication

You can authenticate with the system when using the API by including an HTTP Basic authentication header with all API requests. If you omit authentication information, the system rejects the unauthenticated request with an HTTP 401 response. The system supports HTTP Basic authentication over TLS.

Use the cluster admin account for API authentication.

Find more information

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

Asynchronous methods

Some API methods are asynchronous, which means that the operation they perform might not be complete when the method returns. Asynchronous methods return a handle that you can query to see the status of the operation; status information for some operations might include a percentage of completion.

When you query an asynchronous operation, its result can be one of the following types:

- `DriveAdd`: The system is adding a drive to the cluster.
- `BulkVolume`: The system is performing a copy operation between volumes, such as a backup or restore.
- `Clone`: The system is cloning a volume.
- `DriveRemoval`: The system is copying data from a drive in preparation to remove it from the cluster.
- `RtfiPendingNode`: The system is installing compatible software on a node before adding it to the cluster.

Note the following points when using asynchronous methods or obtaining the status of a running asynchronous operation:

- Asynchronous methods are indicated in the individual method documentation.
- Asynchronous methods return an “asyncHandle”, which is a handle that is known by the issuing API method. You can use the handle to poll for the status or result of the asynchronous operation.
- You can obtain the result of individual asynchronous methods with the GetAsyncResult method. When you use GetAsyncResult to query a completed operation, the system returns the result and automatically purges the result from the system. When you use GetAsyncResult to query an incomplete operation, the system returns the result but does not purge it.
- You can obtain the status and results of all running or completed asynchronous methods using the ListAsyncResults method. In this case, the system does not purge results for completed operations.

Find more information

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

Attributes

Many of the API requests and responses use objects as well as simple types. Objects are a collection of key-value pairs, where the value is a simple type or possibly another object. Attributes are custom name-value pairs that can be set by the user in JSON objects. Some methods enable you to add attributes when creating or modifying objects.

There is a 1000-byte limit on encoded attribute objects.

Object member

This object contains the following member:

Name	Description	Type
attributes	List of name-value pairs in JSON object format.	JSON object

Request example

The following request example uses the AddClusterAdmin method:

```
{
   "method": "AddClusterAdmin",
   "params": {
      "username": "joeadmin",
      "password": "68!5Aru268)$",
      "access": [
         "volume",
         "reporting"
      ],
      "attributes": {
         "name1": "value1",
         "name2": "value2",
         "name3": "value3"
      }
   }
}
```

Copyright information

Trademark information