



Credential

Astra Automation

NetApp
March 07, 2024

Table of Contents

- Credential 1
 - Credential overview 1
 - Create a new credential 1
 - List all credentials 14
 - Retrieve a credential 25
 - Modify a credential 33
 - Delete a credential 44

Credential

Credential overview

The Credential API provides access to securely persist objects containing sensitive information.

Create a new credential

POST `/accounts/{account_id}/core/v1/credentials`

Indicates the desired values for the Credential API resource to be created.

Parameters

Name	Type	In	Required	Description
account_id	string	path	True	ID of the containing account resource <ul style="list-style-type: none">example: {{.Account}}

Request Body

Indicates the desired values for the Credential API resource to be created.

Name	Type	Required	Description
type	string	True	Media type of the resource. Defined values are: <ul style="list-style-type: none">"application/astra-credential"
version	string	True	Version of the resource. Defined values are: <ul style="list-style-type: none">"1.0""1.1"

Name	Type	Required	Description
name	string	True	<p>JSON string containing a user-specified name. When a credential is created for a user, the name will be set to the UUID of that user. Defined values are:</p> <ul style="list-style-type: none">• Minimum length: 1• Maximum length: 127

Name	Type	Required	Description
keyType	string	False	<p>JSON string containing an optional user-specified data validation type descriptor for the contents of the keyStore. Extended update constraints:</p> <ul style="list-style-type: none"> • If a keyType does not exist in the resource and is not specified in an update, the keyType must not continue to exist in the updated resource. • If a keyType does not exist in the resource and is specified in an update, the keyType must be added to the resource and validation must be performed. If the validation fails, the update will fail. • If a keyType exists in the resource and is not specified in an update, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified matches the existing keyType, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified is different from the existing keyType, the update will fail. Defined values are: <ul style="list-style-type: none"> • "generic" - The credential is a generic collection of data. This is equivalent to not including a keyType. <p>Extended validation is performed as follows: 3</p>

Name	Type	Required	Description
keyStore	keyStore	True	<p>JSON object containing JSON strings. Each string in the keyStore represents part of the credential. The name of each string represents the credential part, and the corresponding key-value stores the value of that part of the credential in base64 format. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the base64 Encoded Data Schema No limits are placed on the lengths of the strings.
valid	string	False	<p>JSON string describing whether the credential is valid for use. Default value, if omitted, is "true". Defined values are:</p> <ul style="list-style-type: none"> • "true" - The credential is eligible for use. • "false" - The credential must not be used.
validFromTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema

- The password is stored as a hash.
- A "change" string must exist within the keyStore with a value containing a base64-encoded string "true" or "false".
- A credential of type passwordHash can only be deleted if the user ID indicated by the name does not

Name	Type	Required	Description
validUntilTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is no longer valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema
metadata	type_astra_metadata_update	False	<p>Client and service-specified metadata associated with the resource. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the Astra Metadata Schema. If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

Example request

- "certificate" - The credential is a PEM-

```
{
  "type": "application/astra-credential",
  "version": "1.1",
  "name": "myCert",
  "keyStore": {
    "privKey": "SGkh",
    "pubKey": "VGhpcyBpcyBhbiBleGFtcGxlLg=="
  }
}
```

- Extended validation is performed as follows:
- A "privkey" string must exist within the keyStore, with a value containing a base64- 5 encoded PEM private

Response

Status: 201, Returns the newly created credential resource in the JSON response body.

Name	Type	Required	Description
type	string	True	Media type of the resource. Defined values are: <ul style="list-style-type: none">• "application/astra-credential"
version	string	True	Version of the resource. Defined values are: <ul style="list-style-type: none">• "1.0"• "1.1"
id	string	True	Globally unique identifier of the resource. Defined values are: <ul style="list-style-type: none">• Conforms to the UUIDv4 Schema
name	string	True	JSON string containing a user-specified name. When a credential is created for a user, the name will be set to the UUID of that user. Defined values are: <ul style="list-style-type: none">• Minimum length: 1• Maximum length: 127

Name	Type	Required	Description
keyType	string	False	<p>JSON string containing an optional user-specified data validation type descriptor for the contents of the keyStore. Extended update constraints:</p> <ul style="list-style-type: none"> • If a keyType does not exist in the resource and is not specified in an update, the keyType must not continue to exist in the updated resource. • If a keyType does not exist in the resource and is specified in an update, the keyType must be added to the resource and validation must be performed. If the validation fails, the update will fail. • If a keyType exists in the resource and is not specified in an update, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified matches the existing keyType, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified is different from the existing keyType, the update will fail. Defined values are: • "generic" - The credential is a generic collection of data. This is equivalent to not including a keyType. <p>Extended validation is performed as follows: 7</p>

Name	Type	Required	Description
valid	string	True	<p>JSON string describing whether the credential is valid for use. Default value, if omitted, is "true". Defined values are:</p> <ul style="list-style-type: none"> • "true" - The credential is eligible for use. • "false" - The credential must not be used.
validFromTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema
validUntilTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is no longer valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema

password.

- The password cleartext must conform to the account's password policy.
- The password is stored as a hash.
- A "change" string must exist within the keyStore with a value containing a base64-encoded string "true" or "false".
- A credential of type passwordHash can only be deleted if the user ID indicated by the name does not

Name	Type	Required	Description
metadata	type_astra_metadata_update	True	<p>Client and service-specified metadata associated with the resource. Defined values are:</p> <ul style="list-style-type: none"> Conforms to the Astra Metadata Schema. If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

Example response

```
{
  "type": "application/astra-credential",
  "version": "1.1",
  "id": "58e26f6a-a631-5122-8e0e-7afe306675c0",
  "name": "myCert",
  "valid": "true",
  "metadata": {
    "labels": [],
    "creationTimestamp": "2022-10-06T20:58:16.305662Z",
    "modificationTimestamp": "2022-10-06T20:58:16.305662Z",
    "createdBy": "8f84cf09-8036-51e4-b579-bd30cb07b269"
  }
}
```

Response

Status: 401, Unauthorized

Name	Type	Required	Description
type	string	True	
title	string	True	

keyStore, with a value containing a base64-encoded PEM certificate.

encoded private key.

keyStore, with a value containing a base64-encoded PEM private

Name	Type	Required	Description
detail	string	True	
status	string	True	
correlationID	string	False	

Service (S3) access.

Example response

- Extended validation is

```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
}
```

with a value containing a base64-encoded S3 access

Response

Status: 400, Bad request

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	
invalidFields	array[invalidFields]	False	List of invalid request body fields

Example response

```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```

Response

Status: 409, Conflict

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	
invalidFields	array[invalidFields]	False	List of invalid request body fields

Example response

```
{
  "type": "https://astra.netapp.io/problems/10",
  "title": "JSON resource conflict",
  "detail": "The request body JSON contains a field that conflicts with an idempotent value.",
  "status": "409"
}
```

Response

Status: 403, Forbidden

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```

Error

Status: 404, Not found

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example error response

```
{
  "type": "https://astra.netapp.io/problems/2",
  "title": "Collection not found",
  "detail": "The collection specified in the request URI wasn't
found.",
  "status": "404"
}
```

Definitions

See Definitions

keyStore

JSON object containing JSON strings. Each string in the keyStore represents part of the credential. The name of each string represents the credential part, and the corresponding key-value stores the value of that part of the credential in base64 format. Defined values are:

- Conforms to the base64 Encoded Data Schema No limits are placed on the lengths of the strings.

Hash mapping strings to string

type_astra_label

Name	Type	Required	Description
name	string	True	
value	string	True	

type_astra_metadata_update

Client and service-specified metadata associated with the resource. Defined values are:

- Conforms to the Astra Metadata Schema If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

Name	Type	Required	Description
labels	array[type_astra_label]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
createdBy	string	False	
modifiedBy	string	False	

invalidParams

Name	Type	Required	Description
name	string	True	Name of the invalid query parameter
reason	string	True	Reason why the query parameter is invalid

invalidFields

Name	Type	Required	Description
name	string	True	Name of the invalid request body field
reason	string	True	Reason why the request body field is invalid

List all credentials

GET /accounts/{account_id}/core/v1/credentials

Returns a JSON array of all matching resources in the collection. If individual fields are specified in the request, the items JSON array will contain the fields requested for each matching resource in the order specified.

Parameters

Name	Type	In	Required	Description
account_id	string	path	True	ID of the containing account resource <ul style="list-style-type: none"> example: <code>{{.Account}}</code>
continue		query	False	Indicates the token to resume listing from. <p>URL examples:</p> <pre>"continue=<continue token="" from="" collection="" metadata="">"</continue></pre> <p>OpenAPI examples:</p> <pre>"<continue token="" from="" collection="" metadata="">"</continue>&lt;/continue&gt;</pre>

Name	Type	In	Required	Description
filter	string	query	False	<p>Indicates which fields should be matched in order to return a resource when listing a collection.</p> <p>Supported operators are:</p> <ul style="list-style-type: none"> • eq - Equality • lt/gt - Less than/Greater than • lte/gte - Less than or equals to/Greater than or equals to <p>URL example: "filter=field%20eq%20%27value%27"</p> <p>OpenAPI examples: "field eq 'value'"</p>
include	string	query	False	<p>Indicates which fields should be returned when listing a collection.</p> <p>URL examples: "include=id", "include=id,name"</p> <p>OpenAPI examples: "id"</p> <ul style="list-style-type: none"> • example: id,name

Name	Type	In	Required	Description
limit	string	query	False	Indicates the maximum number of resources to return when listing a collection. URL examples: "limit=2" OpenAPI examples: "2"
orderBy	string	query	False	Indicates the sort order of resource when listing a collection. URL examples: "orderBy=name", "orderBy=name%20desc" OpenAPI examples: "name desc"

Response

Status: 200, Returns a JSON array of all matching resources in the collection. If individual fields are specified in the request, the items JSON array will contain the fields requested for each matching resource in the order specified.

Name	Type	Required	Description
items	array[Any of: credential_1.1_get_response_body, array[]]	True	
metadata	type_astra_metadata_list	True	

Example response

```
{
  "type": "application/astra-credentials",
  "version": "1.1",
  "items": [
    [
      "f97f3441-075e-4f9b-9d1e-47f9f62a3fc4",
      "myCert"
    ],
    [
      "5e365b4d-034f-4f13-95ef-d3a7fbc0df3d",
      "kubeconfig"
    ],
    [
      "11ceb350-1999-4b79-80a6-46a980a27856",
      "password"
    ]
  ],
  "metadata": {
    "labels": [],
    "creationTimestamp": "2022-10-06T20:58:16.305662Z",
    "modificationTimestamp": "2022-10-06T20:58:16.305662Z",
    "createdBy": "8f84cf09-8036-51e4-b579-bd30cb07b269"
  }
}
```

Response

Status: 401, Unauthorized

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
}
```

Response

Status: 400, Bad request

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	
invalidParams	array[invalidParams]	False	List of invalid query parameters

Example response

```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```

Error

Status: 404, Not found

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example error response

```
{
  "type": "https://astra.netapp.io/problems/2",
  "title": "Collection not found",
  "detail": "The collection specified in the request URI wasn't
found.",
  "status": "404"
}
```

Definitions

See Definitions

type_astra_label

Name	Type	Required	Description
name	string	True	
value	string	True	

type_astra_metadata_update

Client and service-specified metadata associated with the resource. Defined values are:

- Conforms to the Astra Metadata Schema If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

Name	Type	Required	Description
labels	array[type_astra_label]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
createdBy	string	False	
modifiedBy	string	False	

credential_1.1_get_response_body

Name	Type	Required	Description
type	string	True	Media type of the resource. Defined values are: <ul style="list-style-type: none">• "application/astra-credential"
version	string	True	Version of the resource. Defined values are: <ul style="list-style-type: none">• "1.0"• "1.1"
id	string	True	Globally unique identifier of the resource. Defined values are: <ul style="list-style-type: none">• Conforms to the UUIDv4 Schema

Name	Type	Required	Description
name	string	True	<p>JSON string containing a user-specified name. When a credential is created for a user, the name will be set to the UUID of that user.</p> <p>Defined values are:</p> <ul style="list-style-type: none">• Minimum length: 1• Maximum length: 127

Name	Type	Required	Description
keyType	string	False	<p>JSON string containing an optional user-specified data validation type descriptor for the contents of the keyStore. Extended update constraints:</p> <ul style="list-style-type: none"> • If a keyType does not exist in the resource and is not specified in an update, the keyType must not continue to exist in the updated resource. • If a keyType does not exist in the resource and is specified in an update, the keyType must be added to the resource and validation must be performed. If the validation fails, the update will fail. • If a keyType exists in the resource and is not specified in an update, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified matches the existing keyType, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified is different from the existing keyType, the update will fail. <p>Defined values are:</p> <p>"generic" - The credential is a generic collection of data. This is</p>

Name	Type	Required	Description
valid	string	True	<p>JSON string describing whether the credential is valid for use. Default value, if omitted, is "true". Defined values are:</p> <ul style="list-style-type: none"> • "true" - The credential is eligible for use. • "false" - The credential must not be used.
validFromTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema
validUntilTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is no longer valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema

must exist within the keyStore with a value containing a base64-encoded cleartext password:

- The password cleartext must conform to the account's password policy.
- The password is stored as a hash.
- A "change" string must exist within the keyStore with a value containing a

Name	Type	Required	Description
metadata	type_astra_metadata_update	True	Client and service-specified metadata associated with the resource. Defined values are: <ul style="list-style-type: none"> • Conforms to the Astra Metadata Schema. If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

[type_astra_metadata_list](#)

Name	Type	Required	Description
continue	string	False	
count	number	False	
labels	array[type_astra_label]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
createdBy	string	False	
modifiedBy	string	False	

[invalidParams](#)

Name	Type	Required	Description
name	string	True	Name of the invalid query parameter
reason	string	True	Reason why the query parameter is invalid

- Extended validation is performed as follows:
- A "certificate" string must exist within the keyStore, with a value containing a base64-encoded

Retrieve a credential

GET /accounts/{account_id}/core/v1/credentials/{credential_id}

Returns the stored Credential API resource in the JSON response body.

Parameters

Name	Type	In	Required	Description
account_id	string	path	True	ID of the containing account resource <ul style="list-style-type: none">example: {{.Account}}
credential_id	string	path	True	ID of the credential collection to list

Response

Status: 200, Returns the stored Credential API resource in the JSON response body.

Name	Type	Required	Description
type	string	True	Media type of the resource. Defined values are: <ul style="list-style-type: none">"application/astra-credential"
version	string	True	Version of the resource. Defined values are: <ul style="list-style-type: none">"1.0""1.1"
id	string	True	Globally unique identifier of the resource. Defined values are: <ul style="list-style-type: none">Conforms to the UUIDv4 Schema

Name	Type	Required	Description
name	string	True	<p>JSON string containing a user-specified name. When a credential is created for a user, the name will be set to the UUID of that user. Defined values are:</p> <ul style="list-style-type: none"> • Minimum length: 1 • Maximum length: 127

value containing a base64-encoded PEM private key.

- "s3" - Indicates the credentials are for Simple Storage Service (S3) access.
- Extended validation is performed as follows:
 - An "accessKey" string must exist within the keyStore with a value containing a base64-encoded S3 access key.
 - An "accessSecret" string must exist within the keyStore with a value containing a base64-encoded S3 access secret.

Name	Type	Required	Description
keyType	string	False	<p>JSON string containing an optional user-specified data validation type descriptor for the contents of the keyStore. Extended update constraints:</p> <ul style="list-style-type: none"> • If a keyType does not exist in the resource and is not specified in an update, the keyType must not continue to exist in the updated resource. • If a keyType does not exist in the resource and is specified in an update, the keyType must be added to the resource and validation must be performed. If the validation fails, the update will fail. • If a keyType exists in the resource and is not specified in an update, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified matches the existing keyType, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified is different from the existing keyType, the update will fail. Defined values are: <ul style="list-style-type: none"> • "generic" - The credential is a generic collection of data. This is equivalent to not including a keyType. <p>Extended validation is performed as follows?</p>

Name	Type	Required	Description
valid	string	True	<p>JSON string describing whether the credential is valid for use. Default value, if omitted, is "true". Defined values are:</p> <ul style="list-style-type: none"> • "true" - The credential is eligible for use. • "false" - The credential must not be used.
validFromTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema
validUntilTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is no longer valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema

password.

- The password cleartext must conform to the account's password policy.
- The password is stored as a hash.
- A "change" string must exist within the keyStore with a value containing a base64-encoded string "true" or "false".
- A credential of type passwordHash can only be deleted if the user ID indicated by the name does not

Name	Type	Required	Description
metadata	type_astra_metadata_update	True	<p>Client and service-specified metadata associated with the resource. Defined values are:</p> <ul style="list-style-type: none"> Conforms to the Astra Metadata Schema. If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

Example response

```

{
  "type": "application/astra-credential",
  "version": "1.1",
  "id": "58e26f6a-a631-5122-8e0e-7afe306675c0",
  "name": "myCert",
  "valid": "true",
  "metadata": {
    "labels": [],
    "creationTimestamp": "2022-10-06T20:58:16.305662Z",
    "modificationTimestamp": "2022-10-06T20:58:16.305662Z",
    "createdBy": "8f84cf09-8036-51e4-b579-bd30cb07b269"
  }
}

```

Response

Status: 401, Unauthorized

Name	Type	Required	Description
type	string	True	
title	string	True	

keyStore, with a value containing a base64-encoded PEM certificate.

encoded private key.

keyStore, with a value containing a base64-encoded PEM private

Name	Type	Required	Description
detail	string	True	
status	string	True	
correlationID	string	False	

Service (S3) access.

Example response

- Extended validation is

```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
}
```

with a value containing a base64-encoded S3 access

Response

Status: 400, Bad request

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	
invalidParams	array[invalidParams]	False	List of invalid query parameters

Example response

```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```


Response

Status: 403, Forbidden

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```

Error

Status: 404, Not found

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example error response

```
{
  "type": "https://astra.netapp.io/problems/2",
  "title": "Collection not found",
  "detail": "The collection specified in the request URI wasn't
found.",
  "status": "404"
}
```

Definitions

See Definitions

type_astra_label

Name	Type	Required	Description
name	string	True	
value	string	True	

type_astra_metadata_update

Client and service-specified metadata associated with the resource. Defined values are:

- Conforms to the Astra Metadata Schema If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

Name	Type	Required	Description
labels	array[type_astra_label]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
createdBy	string	False	
modifiedBy	string	False	

invalidParams

Name	Type	Required	Description
name	string	True	Name of the invalid query parameter
reason	string	True	Reason why the query parameter is invalid

Modify a credential

PUT /accounts/{account_id}/core/v1/credentials/{credential_id}

Replaces the entire stored resource with the supplied JSON request body while preserving the key values that are not user-modifiable.

Parameters

Name	Type	In	Required	Description
account_id	string	path	True	ID of the containing account resource <ul style="list-style-type: none"> example: <code>{{.Account}}</code>
credential_id	string	path	True	ID of the credential collection to list

Request Body

Replaces the entire stored resource with the supplied JSON request body while preserving the key values that are not user-modifiable.

Name	Type	Required	Description
type	string	True	Media type of the resource. Defined values are: <ul style="list-style-type: none"> "application/astra-credential"
version	string	True	Version of the resource. Defined values are: <ul style="list-style-type: none"> "1.0" "1.1"
id	string	False	Globally unique identifier of the resource. Defined values are: <ul style="list-style-type: none"> Conforms to the UUIDv4 Schema
name	string	True	JSON string containing a user-specified name. When a credential is created for a user, the name will be set to the UUID of that user. Defined values are: <ul style="list-style-type: none"> Minimum length: 1 Maximum length: 127

Name	Type	Required	Description
keyType	string	False	<p>JSON string containing an optional user-specified data validation type descriptor for the contents of the keyStore. Extended update constraints:</p> <ul style="list-style-type: none"> • If a keyType does not exist in the resource and is not specified in an update, the keyType must not continue to exist in the updated resource. • If a keyType does not exist in the resource and is specified in an update, the keyType must be added to the resource and validation must be performed. If the validation fails, the update will fail. • If a keyType exists in the resource and is not specified in an update, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified matches the existing keyType, the keyType must be preserved. • If a keyType exists in the resource and is specified in an update, and the keyType specified is different from the existing keyType, the update will fail. Defined values are: • "generic" - The credential is a generic collection of data. This is equivalent to not including a keyType. <p>Extended validation is performed as follows³⁵</p>

Name	Type	Required	Description
keyStore	keyStore	True	<p>JSON object containing JSON strings. Each string in the keyStore represents part of the credential. The name of each string represents the credential part, and the corresponding key-value stores the value of that part of the credential in base64 format. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the base64 Encoded Data Schema No limits are placed on the lengths of the strings.
valid	string	False	<p>JSON string describing whether the credential is valid for use. Default value, if omitted, is "true". Defined values are:</p> <ul style="list-style-type: none"> • "true" - The credential is eligible for use. • "false" - The credential must not be used.
validFromTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema

- The password is stored as a hash.
- A "change" string must exist within the keyStore with a value containing a base64-encoded string "true" or "false".
- A credential of type passwordHash can only be deleted if the user ID indicated by the name does not

Name	Type	Required	Description
validUntilTimestamp	string	False	<p>JSON string containing a timestamp indicating when the credential is no longer valid. These are used for credential rotation. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the ISO-8601 Date Time Schema
metadata	type_astra_metadata_update	False	<p>Client and service-specified metadata associated with the resource. Defined values are:</p> <ul style="list-style-type: none"> • Conforms to the Astra Metadata Schema. If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

Example request

- "certificate" - The credential is a PEM-

```
{
  "type": "application/astra-credential",
  "version": "1.1",
  "name": "oldCert",
  "keyStore": {
    "privKey": "SGkh",
    "pubKey": "VGhpcyBpcyBhbiBleGFtcGxlLg=="
  }
}
```

- Extended validation is performed as follows:
- A "privkey" string must exist within the keyStore, with a value containing a base64-₃₇ encoded PEM private

Response

Status: 204, Indicates if the credential resource was updated.

Response

Status: 403, Forbidden

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```

Response

Status: 406, Not acceptable

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/32",
  "title": "Unsupported content type",
  "detail": "The response can't be returned in the requested format.",
  "status": "406"
}
```

keyStore with a value containing a base64-encoded S3 access

Response

Status: 500, Internal server error

string must exist within the keyStore

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/34",
  "title": "Internal server error",
  "detail": "The server was unable to process this request.",
  "status": "500"
}
```

Response

Status: 412, Precondition failed

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	

Name	Type	Required	Description
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/38",
  "title": "Precondition not met",
  "detail": "The conditional headers aren't satisfied.",
  "status": "412"
}
```

Response

Status: 409, Conflict

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/39",
  "title": "Credential exists",
  "detail": "A credential of this type already exists.",
  "status": "409"
}
```

Response

Status: 502, Bad gateway

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/40",
  "title": "Communication failed",
  "detail": "Communication wasn't made with downstream services.",
  "status": "502"
}
```

Response

Status: 503, Service unavailable

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/41",
  "title": "Service not ready",
  "detail": "Currently, the service can't respond to this request.",
  "status": "503"
}
```

Error

Status: 400, Bad request

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	
invalidParams	array[invalidParams]	False	List of invalid query parameters

Example error response

```
{
  "type": "https://astra.netapp.io/problems/7",
  "title": "Invalid JSON payload",
  "detail": "The request body is not valid JSON.",
  "status": "400"
}
```

Definitions

See Definitions

keyStore

JSON object containing JSON strings. Each string in the keyStore represents part of the credential. The name of each string represents the credential part, and the corresponding key-value stores the value of that part of the credential in base64 format. Defined values are:

- Conforms to the base64 Encoded Data Schema No limits are placed on the lengths of the strings.

Hash mapping strings to string

type_astra_label

Name	Type	Required	Description
name	string	True	
value	string	True	

type_astra_metadata_update

Client and service-specified metadata associated with the resource. Defined values are:

- Conforms to the Astra Metadata Schema If not specified on create, a metadata object will be created with no labels. If not specified on update, the metadata object's labels, creationTimestamp and createdBy, will be preserved without modification.

Name	Type	Required	Description
labels	array[type_astra_label]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
createdBy	string	False	
modifiedBy	string	False	

invalidFields

Name	Type	Required	Description
name	string	True	Name of the invalid request body field
reason	string	True	Reason why the request body field is invalid

invalidParams

Name	Type	Required	Description
name	string	True	Name of the invalid query parameter
reason	string	True	Reason why the query parameter is invalid

Delete a credential

DELETE /accounts/{account_id}/core/v1/credentials/{credential_id}

Removes the specified Credential API resource.

Side Effects

- If the credential being deleted is relied on by another resource or API, the deletion might cause unintended side effects within other APIs or the system as a whole.

Parameters

Name	Type	In	Required	Description
account_id	string	path	True	ID of the containing account resource <ul style="list-style-type: none"> • example: <code>{{.Account}}</code>
credential_id	string	path	True	ID of the credential collection to list

Response

Status: 204, Indicates if the credential resource was deleted.

Response

Status: 401, Unauthorized

Name	Type	Required	Description
type	string	True	
title	string	True	

Name	Type	Required	Description
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
}
```

Response

Status: 400, Bad request

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	
invalidParams	array[invalidParams]	False	List of invalid query parameters

Example response

```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```

Response

Status: 403, Forbidden

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example response

```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```

Error

Status: 404, Not found

Name	Type	Required	Description
type	string	True	
title	string	True	
detail	string	True	
status	string	True	
correlationID	string	False	

Example error response

```
{
  "type": "https://astra.netapp.io/problems/1",
  "title": "Resource not found",
  "detail": "The resource specified in the request URI wasn't found.",
  "status": "404"
}
```

Definitions

See Definitions

invalidParams

Name	Type	Required	Description
name	string	True	Name of the invalid query parameter
reason	string	True	Reason why the query parameter is invalid

Copyright information

Copyright © 2024 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.