IBM Cloud Object Storage System™
Version 3.14.12

Container Mode Credentials Management API Guide

IBM

This edition applies to IBM Cloud Object Storage System and is valid until replaced by new editions.

Contents

Chapter 1. Overview

This specific section covers the interfaces relating to the management of access keys. While these interfaces are heavily influenced by the interface provided by AWS IAM and OpenStack Identity API's, they are not intended to replace these interfaces but complement them.

The Storage as a Service (STaaS) feature will deliver a base set of Service APIs that are intended for deployment, system management, and service operator usage.

This interface is based off of the Keystone credentials API.

Additional notes to be reviewed and considered for the implementation of the API:

1. In the absence of an external Keystone authentication server for end users, the project ID referenced throughout this document will be the storage account ID. This API continues to use project ID in order to remain faithful to the original Keystone credentials API, and to support future migration to Keystone.
2. The storage account ID used should be generated by the client creating the credentials with this API, to facilitate migration to Keystone server eventually
3. The credentials ID throughout this document is the AWS Key, including when the client making the requests with this API is generating the AWS Key

Roles and permissions

As stated before, the Container Mode Credentials Management API is intended to be used by development operations, system management, and service operators. Therefore, the authenticated user that is leveraging this interface must be assigned a role that has been configured with permissions that allow for Container Mode Credentials Management API and access key management.

Note: Because the permission system within STaaS is under design, role guidance is provided in each section in order to help shape expectations for such design. This guidance will be removed at a later time.

Interface details

Table 1. Command Summary

Interface	Command	Description
Create credential	**`POST <accesser>:8338/credentials`**	Create Access Key for a given user, domain and project
List credentials	**`GET <accesser>:8338/credentials`**	Returns a list of all credentials or credentials for a user
Show credential details	**`GET <accesser>:8338/credentials/[credential id]`**	Returns information about a specific credential ID
Delete credential	**`DELETE <accesser>:8338/credentials/[credential id]`**	Deletes Access Key for a given user, domain and project
Update credential	**`PATCH <accesser>:8338/credentials/[credential id]`**	Updates an existing credential for status of credential (Active/Inactive)

Chapter 2. Create credential

Create a new secret access key and corresponding access key ID for the specified user. The default status for new keys generated by the IBM Cloud Object Storage System™ is Active.

If a "blob" is not sent in the request, then the system will generate keys for the request and send as part of the response. Note that if a "blob" is sent, it must contain an access key but is not required to have a secret key. In this case, the system will generate the secret and send this in the response along with the access key provided in the "blob".

See the following section for the recommended way of generating AWS keys in the case where this will be provided as part of "blob" in the request (for example, if the system is not expected to generate the AWS keys for the request, but is expected to store and use what is provided by the clients).

Impersonate permission is required in order to create a key when *project_id* does not match the authenticated user.

Common request parameters

Table 2. Common request parameters

Request Parameter	**Style**	**Type**	**Description**
credential	body	Object	A credential object
blob	body	String	Required: No (the system will generate keys if blob is not present)If "blob" is present, it must contain an access key but is not required to have a secret key. Only the secret key will be generated by the system in this case)
project_id	body	String	The storage account ID.
type	body	String	Required: Yes (ec2)
subject_ibm_id	body	String	Identity of the user/service within IAM. Empty string will not be allowed and will be rejected with a 400 Required: No

Common response parameters

Table 3. Common response parameters

Response Parameter	**Style**	**Type**	**Description**
credential	body	Object	A credential object Required: Yes

Table 3. Common response parameters (continued)

Response Parameter	Style	Type	Description

`blob`	body	String	The credential itself, as a serialized blob
			`"blob": {`
			`"access": "<access key>",`
			`"secret": "<secret key>",`
			`"status": "<Active/Inactive>"`
			`}`
			Required: Yes (if the system generates, the status will be Active by default)

`project_id`	body	String	The storage account ID. Required: Yes
`type`	body	String	Required: Yes (ec2)
`subject_ibm_id`	body	String	Identity of the user/service within IAM. Required: No. Only present if **`subject_ibm_id`** was included in the request.
`id`	body	String	This will be the Access Key Required: Yes

Table 4. HTTP response codes

HTTP Response Code	Description
`201 Created`	Request is successful
`400 Bad Request`	Request does not comply with specification
`401 Unauthorized`	The provided token is not valid or cannot be verified
`403 Forbidden`	The provided token although valid, does not provide appropriate permissions to the user
`404 Not Found`	No such user or storage account exists
`405 Method Not Allowed`	Although the user may be valid, the user does not have privileges to access storage account
`409 Conflict`	The user has reached maximum number of keys that can be generated for the user
`503 Service Unavailable`	The credential API service is currently unavailable

Table 5. Role Guidance

Role Guidance	Description
`Key Management Admin Permission`	Generic permission to allow a admin to manage access keys
`Key Management User Permission`	Generic permission to allow a user to manage access keys

Example Output

```
Request
-------
POST <accesser>:8338/credentials/
{
    "credential": {
        "project_id": "731fc6f265cd486d900f16e84c5cb594",
        "type": "ec2",
        "subject_ibm_id": "IBMid-61KR43CAFF",
        "user_id": "bb5476fd12884539b41d5a88f838d773"
    }
}

Response
--------
HTTP/1.1 201 CREATED
Content-Length: 342
Content-Type: application/json; charset=utf-8
X-Trans-Id: tx8ea13a3a835544d8bebf1-0056eb522a
Date: Fri, 18 Mar 2016 00:56:10 GMT
X-Timestamp: 1458262564.22774
{
    "credential": {
        "user_id": "bb5476fd12884539b41d5a88f838d773",
        "blob": {"access": "7da79ff0aa364e1396f067e352b9b79a",
                 "secret": "secretKey",
                 "status": "Inactive"
        },
        "project_id": "731fc6f265cd486d900f16e84c5cb594",
        "type": "ec2",
        "subject_ibm_id": "IBMid-61KR43CAFF",
        "id": "7da79ff0aa364e1396f067e352b9b79a"
    }
}
```

Chapter 3. List credentials

Lists all secret access keys and corresponding access key IDs for the specified user and storage account name.

Query parameters successively filter a request:

- No parameter: Keys for the authenticated user will be returned
- Project only: All root account keys for that project will be returned

Impersonate permission is required in order to list a key when *project_id* (storage account ID) does not match the authenticated user.

Common request parameters

Table 6. Common request parameters

Request Parameter	**Style**	**Type**	**Description**
Common Request Parameters : See Storage Account Management API Common Request and Response Headers			
project_id	query (within URI)	String	The storage account ID.
limit (Optional)	query (within URI)	Integer	For an integer value n, limits the number of results to n. Maximum limit is 1000 and the default limit is 1000 if none is provided.
marker(Optional)	query (within URI)	String	For a string value x, returns account names that are greater than the marker value.
end_marker (Optional)	query (within URI)	String	For a string value x, returns account names that are less than the marker value.

Common response parameters

Table 7. Common response parameters

Response Parameter	**Style**	**Type**	**Description**
credentials	body	List	A credential object list
blob	body	String	The credential itself, as a serialized blob Required: Yes
project_id	body	String	The storage account ID. Required: Yes
type	body	String	Required: Yes (ec2)

Table 7. Common response parameters (continued)

Response Parameter	Style	Type	Description
subject_ibm_id	body	String	Identity of the user/service within IAM. Required: No. Only present if **subject_ibm_id** was configured for this access key.
id	body	String	This will be the Access Key Required: Yes

Table 8. HTTP response codes

HTTP Response Code	Description
200 OK	The specified user or storage account exists.
400 Bad Request	Request does not comply with specification.
401 Unauthorized	The provided token is not valid or cannot be verified.
403 Forbidden	The provided token although valid, does not provide appropriate permissions to the user.
404 Not Found	The specified user or storage account does not exist.
405 Method Not Allowed	Although the user may be valid, the user does not have privileges to access storage account.
503 Service Unavailable	The credential API service is currently unavailable.

Table 9. Role Guidance

Role Guidance	Description
Key Listing Admin Permission	Generic permission to allow an admin to list access keys
Key Listing User Permission	Generic permission to allow a user to list access keys

Example Output

```
Request
-------
GET <accesser>:8338/credentials/?project_id=6e01855f345f4c59812999b5e459137d

Response
--------
HTTP/1.1 200 OK
Content-Length: 786
Accept-Ranges: bytes
Content-Type: application/json; charset=utf-8
X-Timestamp: 1458262564.22774
X-Trans-Id: tx8ea13a3a835544d8bebf1-0056eb522a
Date: Fri, 18 Mar 2016 00:56:10 GMT
{
    "credentials": [
        {
            "user_id": "bb5476fd12884539b41d5a88f838d773",
            "blob": {"access": "a42a27755ce6442596b049bd7dd8a563",
                     "secret": "71faf1d40bb24c82b479b1c6fbbd9f0c",
                     "status": "Active"
            },
            "project_id": "6e01855f345f4c59812999b5e459137d",
            "type": "ec2",
            "subject_ibm_id": "IBMid-61KR43CAFF",
            "id": "a42a27755ce6442596b049bd7dd8a563"
        },
        {
            "user_id": "6f556708d04b4ea6bc72d7df2296b71a",
            "blob": {"access": "7da79ff0aa364e1396f067e352b9b79a",
                     "secret": "7a18d68ba8834b799d396f3ff6f1e98c",
                     "status": "Active"
            },
            "project_id": "1a1d14690f3c4ec5bf5f321c5fde3c16",
            "type": "ec2",
            "id": "7da79ff0aa364e1396f067e352b9b79a"
        }
    ]
}
```

Chapter 4. Show credential details

Lists secret access key and corresponding access key ID for the specified credential identity. Can be disabled globally on a system (honors Manager UI settings disallowing showing secret access keys after initial creation).

Impersonate permission is required in order to show a secret key when *project_id* (storage account ID) does not match the authenticated user.

Common request parameters

Table 10. Request Parameter

Request Parameter	**Style**	**Type**	**Description**
Common Request Parameters : See Storage Account Management API Common Request and Response Headers			
id	URI	String	The id for the credential

Common response parameters

Table 11. Response Parameter

Response Parameter	**Style**	**Type**	**Description**
`credential`	body	Object	A credential object
`blob`	body	String	The credential itself, as a serialized blob Required: Yes
`project_id`	body	String	The storage account ID. Required: Yes
`type`	body	String	Required: Yes (ec2)
`subject_ibm_id`	body	String	Identity of the user/service within IAM. Required: No. Only present if **`subject_ibm_id`** was configured for this access key.
`id`	body	String	This will be the Access Key Required: Yes

Table 12. HTTP response code

HTTP Response Code	**Description**
`200 OK`	Request was successful
`400 Bad Request`	Request does not comply with specification
`401 Unauthorized`	The provided token is not valid or cannot be verified

Table 12. HTTP response code (continued)

HTTP Response Code	Description
`403 Forbidden`	The provided token although valid, does not provide appropriate permissions to the user
`404 Not Found`	No such user or storage account exists
`405 Method Not Allowed`	Although the user may be valid, the user does not have privileges to access storage account
`503 Service Unavailable`	The credential API service is currently unavailable

Table 13. Role Guidance

Role Guidance	Description
`Key Listing Admin Permission`	Generic permission to allow an admin to list access keys
`Key Listing User Permission`	Generic permission to allow a user to list access keys

Example Output

```
Request
-------
GET <accesser>:8338/credentials/a42a27755ce6442596b049bd7dd8a563

Response
--------
HTTP/1.1 200 OK
Content-Length: 368
Accept-Ranges: bytes
Content-Type: application/json; charset=utf-8
X-Timestamp: 1458262564.22774
X-Trans-Id: tx8ea13a3a835544d8bebf1-0056eb522a
Date: Fri, 18 Mar 2016 00:56:10 GMT
{
    "credential": {
        "user_id": "bb5476fd12884539b41d5a88f838d773",
        "blob": {"access": "a42a27755ce6442596b049bd7dd8a563",
                 "secret": "71faf1d40bb24c82b479b1c6fbbd9f0c",
                 "status": "Active"
        },
        "project_id": "6e01855f345f4c59812999b5e459137d",
        "type": "ec2",
        "subject_ibm_id": "IBMid-61KR43CAFF",
        "id": "a42a27755ce6442596b049bd7dd8a563"
    }
}
```

Chapter 5. Delete credential

Delete the specified credential identity.

Impersonate permission is required in order to show a secret key when *project_id* (storage account ID) does not match the authenticated user.

Common request parameters

Table 14. Request Parameter

Request Parameter	**Style**	**Type**	**Description**
Common request headers: See Storage Account Management API Common Request and Response Headers			
id	URI	String	The id for the credential

Common response parameters

Table 15. Response Parameter

Response Parameter	**Style**	**Type**	**Description**
None			

Table 16. HTTP Response Code

HTTP Response Code	**Description**
204 No Content	Successful request
400 Bad Request	Request does not comply with specification
401 Unauthorized	The provided token is not valid or cannot be verified
403 Forbidden	The provided token although valid, does not provide appropriate permissions to the user
404 Not Found	No such user or storage account exists
405 Method Not Allowed	Although the user may be valid, the user does not have privileges to access storage account
503 Service Unavailable	The credential API service is currently unavailable

Table 17. Role Guidance

Role Guidance	**Description**
Key Management Admin Permission	Generic permission to allow an admin to manage access keys
Key Management User Permission	Generic permission to allow a user to manage access keys

Example Output

```
Request
---------
```

```
DELETE <accesser>:8338/credentials/a42a27755ce6442596b049bd7dd8a563

Response
---------
HTTP/1.1 204 OK
Content-Length: 0
Accept-Ranges: bytes
Content-Type: application/json; charset=utf-8
X-Timestamp: 1458262564.22774
X-Trans-Id: tx8ea13a3a835544d8bebf1-0056eb522a

This operation does not send a request body and does not return a response body
```

Chapter 6. Update credential

Update the status of a specified credential as requested for the credential. Impersonate permission is required in order to show a secret key when *project_id* (storage account ID) does not match the authenticated user.

Common request parameters

Table 18. Request Parameters

Request Parameter	Style	Type	Description
Common Request Parameters : See Storage Account Management API Common Request and Response Headers			
`credential`	body	Object	A credential object
`blob`	body	String	The credential itself, as a serialized blob Required: Yes
`project_id`	body	String	The storage account ID. Required: Yes
`type`	body	String	Required: Yes (ec2)

Common response parameters

Table 19. Response Parameters

Response Parameter	Style	Type	Description
`credential`	body	Object	A credential object
`blob`	body	String	The credential itself, as a serialized blob Required: Yes
`project_id`	body	String	The storage account ID Required: Yes
`type`	body	String	Required: Yes (ec2)
`subject_ibm_id`	body	String	Identity of the user/service within IAM. Required: No. Only present if **`subject_ibm_id`** was configured for this access key
`id`	body	String	The Access Key Required: Yes, only if being requested on behalf of another user

Table 20. HTTP Response Code	
HTTP Response Code	**Description**
200 OK	Success request
400 Bad Request	Request does not comply with specification
401 Unauthorized	The provided token is not valid or cannot be verified
403 Forbidden	The provided token although valid, does not provide appropriate permissions to the user
404 Not Found	No such user or storage account exists
405 Method Not Allowed	Although the user may be valid, the user does not have privileges to access storage account
503 Service Unavailable	The credential API service is currently unavailable

Table 21. Role Guidance	
Role Guidance	**Description**
Key Management Permission	Generic permission to allow a user to manage access keys

Example Output

```
Request
-------
PATCH <accesser>:8338/credentials/181920
Content-Length: 263
Accept-Ranges: bytes
Content-Type: application/json; charset=utf-8
X-Timestamp: 1458262564.22774
X-Trans-Id: tx8ea13a3a835544d8bebf1-0056eb522a
{
    "credential": {
        "blob": {"access": "181920",
                 "secret": "secretKey",
                 "status": "Inactive"
        },
        "project_id": "731fc6f265cd486d900f16e84c5cb594",
        "type": "ec2",
        "user_id": "bb5476fd12884539b41d5a88f838d773"
    }
}

Response
--------
HTTP/1.1 200 OK
Content-Length: 341
Accept-Ranges: bytes
Content-Type: application/json; charset=utf-8
X-Timestamp: 1458262564.22774
X-Trans-Id: tx8ea13a3a835544d8bebf1-0056eb522a
{
    "credential": {
        "user_id": "bb5476fd12884539b41d5a88f838d773",
        "blob": {"access": "a42a27755ce6442596b049bd7dd8a563",
                 "secrete": "secretKey",
                 "status": "Inactive"
        },
        "project_id": "731fc6f265cd486d900f16e84c5cb594",
        "type": "ec2",
        "subject_ibm_id": "IBMid-61KR43CAFF",
        "id": "a42a27755ce6442596b049bd7dd8a563"
    }
}
```

Chapter 7. Method to generate AWS access keys

The recommended method to generate AWS Access Keys.

The generated AWS Access Keys provided to the system should be randomly generated alphanumeric strings of 20 and 40 characters. The **secretAccessKey** one should use a **SecureRandom** whereas the **accessKeyId** should use a normal Random.

The Access Key ID should be globally unique, since this will be used as the credential ID by the system

Code Sample

```
String accessKeyId = Util.randomAlphanumericString(20);
String secretAccessKey = Util.randomAlphanumericString(secureRandom, 40);

private static final char[] ALPHANUMERIC = new char[]{'0', '1', '2', '3', '4', '5', '6', '7',
'8', '9', 'a',
 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't',
'u', 'v', 'w',
 'x', 'y', 'z', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P',
'Q', 'R', 'S',
 'T', 'U', 'V', 'W', 'X', 'Y', 'Z'};

public static char randomAlphanumericCharacter(final Random random)
   {
      return ALPHANUMERIC[random.nextInt(ALPHANUMERIC.length)];
   }

   public static String randomAlphanumericString(final Random random, final int length)
   {
      final StringBuilder sb = new StringBuilder(length);

      for (int i = 0; i < length; i++)
         sb.append(randomAlphanumericCharacter(random));

      return sb.toString();
   }
```

Notices

This information was developed for products and services offered in the US. This material might be available from IBM® in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785

US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Accesser®, Cleversafe®, ClevOS™, Dispersed Storage®, dsNet®, IBM Cloud Object Storage Accesser®, IBM Cloud Object Storage Dedicated™, IBM Cloud Object Storage Insight™, IBM Cloud Object Storage Manager™, IBM Cloud Object Storage Slicestor®, IBM Cloud Object Storage Standard™, IBM Cloud Object Storage System™, IBM Cloud Object Storage Vault™, SecureSlice™, and Slicestor® are trademarks or registered trademarks of Cleversafe, an IBM Company and/or International Business Machines Corp.

Other product and service names might be trademarks of IBM or other companies.

Homologation statement

This product may not be certified in your country for connection by any means whatsoever to interfaces of public telecommunications networks. Further certification may be required by law prior to making any such connection. Contact an IBM representative or reseller for any questions.

IBM®