API QUICK START

HYCU Data Protection as a Service for Azure

February 2023

HYCU®

Legal notices

Copyright notice

© 2023 HYCU. All rights reserved.

This document contains proprietary information, which is protected by copyright. No part of this document may be photocopied, reproduced, distributed, transmitted, stored in a retrieval system, modified or translated to another language in any form by any means, without the prior written consent of HYCU.

Trademarks

HYCU logos, names, trademarks and/or service marks and combinations thereof are the property of HYCU or its affiliates. Other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Azure®, Microsoft®, Microsoft Edge™, and Windows® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Google Chrome™ is a trademark of Google LLC.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Mozilla and Firefox are trademarks of the Mozilla Foundation in the U.S. and other countries.

Disclaimer

The details and descriptions contained in this document are believed to have been accurate and up to date at the time the document was written. The information contained in this document is subject to change without notice.

HYCU provides this material "as is" and makes no warranty of any kind, expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. HYCU shall not be liable for errors and omissions contained herein. In no event shall HYCU be liable for any direct, indirect, consequential, punitive, special or incidental damages, including, without limitation, damages for loss and profits, loss of anticipated savings, business interruption, or loss of information arising out of the use or inability to use this document, or any action taken based on the information contained herein, even if it has been advised of the possibility of such damages, whether based on warranty, contract, or any other legal theory.

The only warranties for HYCU products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty.

Notice

This document is provided in connection with HYCU products. HYCU may have copyright, patents, patent applications, trademark, or other intellectual property rights covering the subject matter of this document.

Except as expressly provided in any written license agreement from HYCU, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property on HYCU products. Use of underlying HYCU product(s) is governed by their respective Software License and Support Terms.

Important: Please read Software License and Support Terms before using the accompanying software product(s).

HYCU
www.hycu.com

Using the HYCU for Azure REST API Explorer

HYCU for Azure enables you to automate your data protection activities by using its REST-based application programming interface (API). The REST API Explorer is integrated into the HYCU for Azure web user interface and allows you to quickly test and make API requests.

Prerequisites

- A service principal is configured in HYCU for Azure. For details on how to do this, see *HYCU for Azure Help*.
- The API requests that you plan to make are authorized. For details on how to do this, see "Authorizing API requests" below.

Accessing the HYCU for Azure REST API Explorer

1. Click the user session information (👤) at the upper right of the screen, and then select **REST API Explorer**. The REST API Explorer dialog box opens.
2. Depending on whether you want to access the central service REST API Explorer or the service subscription REST API Explorer, click one of the following:
   - **Central REST API Explorer**
   - **Subscription REST API Explorer**

The corresponding HYCU for Azure REST API Explorer opens, allowing you to make requests after your access is authorized.

Authorizing API requests

Authorizing API requests includes acquiring and applying an access token. For general information on access tokens, see Azure documentation.

Procedure

1. Acquire a valid OAuth 2.0 access token. For instructions, see Azure documentation.

   Example This example includes a program in Python that you can use to acquire a new access token for the HYCU for Azure REST API. Make sure to replace <*TenantId*>, <*ClientId*>, and <*ClientSecret*> with the values of the service principal.

   ```
   from azure.common.credentials import ServicePrincipalCredentials

   TENANT_ID = '<TenantId>'
   CLIENT_ID = '<ClientId>'
   CLIENT_SECRET = '<ClientSecret>'
   ```

```
RESOURCE = 'https://management.azure.com'

# GENERATE SERVICE PRINCIPAL ID TOKEN
credential = ServicePrincipalCredentials(tenant=TENANT_ID,
                                         client_id=CLIENT_ID,
                                         secret=CLIENT_SECRET,
                                         resource=RESOURCE)
id_token = "Bearer %s" % credential.token.get("access_token")

print("Token:\n%s" % id_token)
```

2. Apply the access token to both the Central REST API Explorer and the Subscription REST API Explorer to authorize all API requests and take full advantage of the HYCU for Azure REST API. To authorize the requests that you will make, do the following:
   a. Copy the access token that you acquired.
   b. Access the Central REST API Explorer or the Subscription REST API Explorer.
   c. In the upper-right part of the webpage, click **Authorize**.
   d. In the Available authorizations dialog box, paste the access token, and then click **Authorize**.

Example

This example shows how to retrieve a list of virtual machines that belong to the default protection set:

a. Copy the access token that you acquired.
b. Access the Subscription REST API Explorer.
c. In the upper-right part of the webpage, click **Authorize**.
d. In the Available authorizations dialog box, paste the access token, and then click **Authorize**.
e. Make the following request:

```
GET /subscriptions/<SubscriptionUUID>/protectionSets/
<ProtectionSetUUID>/virtualMachines
```

Example program: assigning a policy to a virtual machine

The following Python program is an example of how you can assign a policy to a virtual machine in your Azure subscription by using the HYCU for Azure REST API.

Make sure to replace the following in the source code:

- *<TenantId>*, *<ClientId>*, and *<ClientSecret>* with the values of the service principal.
- *<SubscriptionUUID>* with the UUID of your Azure subscription.
- *<ProtectionSetUUID>* with the UUID of your HYCU for Azure protection set.
- *<PolicyUUID>* with the UUID of the policy that you want to assign.
- *<VirtualMachineUUID>* with the UUID of the virtual machine to which you want to assign the policy.

```
from azure.common.credentials import ServicePrincipalCredentials
from json import JSONDecodeError
from http import client
import json
import http

TENANT_ID = '<TenantId>'
CLIENT_ID = '<ClientId>'
CLIENT_SECRET = '<ClientSecret>'
RESOURCE = 'https://management.azure.com'

REGISTRY_ENDPOINT = 'registry.azure.hycu.com'

# Virtual Machine unique identifier:
SUBSCRIPTION = '<SubscriptionUUID>'
VIRTUAL_MACHINE = '<VirtualMachineUUID>'

# Policy unique identifier:
PROTECTION_SET = '<ProtectionSetUUID>'
POLICY = '<PolicyUUID>'

def request_id_token(tenant_id, client_id, client_secret, resource):
      credential = ServicePrincipalCredentials(tenant=tenant_id,
                   client_id=client_id, secret=client_secret,
                   resource=resource)
      return "Bearer %s" % credential.token.get("access_token")
```

```
def get_manager_url(connection, header):
      url = "/api/v1/scope/subscriptions"
      connection.request(method="GET", url=url, body={}, headers=header)
      r = connection.getresponse()
      output = json.loads(r.read())
      return output['entities'][0]['subscriptions'][0]
      ['managerEndpointUrl'].split('/')[2]

def print_response(response):
      print('Response status: %d' % response.status)
      temp = response.read()
      try:
           print_data = json.loads(temp)
           print(json.dumps(print_data, indent=4, sort_keys=True))
           return print_data
      except JSONDecodeError:
           print(temp)

def policy_assign(subscription_uuid, protection_set_uuid,
                  virtual_machine_uuid, policy_uuid, connection, header):
      # Define the body
      body = [virtual_machine_uuid]
      json_body = json.dumps(body)
      url = "/api/v1/subscriptions/{}/protectionSets/{}/policies/{}/
            assign".format(subscription_uuid, protection_set_uuid,
            policy_uuid)

     # Creating the request: 'POST /api/v1/subscriptions/{}/
                             protectionSets/{}/policies/{}/assign'
     connection.request(method="POST", url=url,
                        body=json_body, headers=header)
     return connection.getresponse()

# Establish connection to Registry
registry_endpoint_connection = http.client.HTTPSConnection(REGISTRY_
ENDPOINT)

# Generate credentials
id_token = request_id_token(TENANT_ID, CLIENT_ID, CLIENT_SECRET,
RESOURCE)
print("Token:\n%s" % id_token)
headers = {'Content-type': 'application/json', 'Authorization': id_token}

# Establish connection to Manager
MANAGER_ENDPOINT = get_manager_url(registry_endpoint_connection, headers)
print("Manager URL: " + MANAGER_ENDPOINT)
```

```
manager_endpoint_connection = http.client.HTTPSConnection(MANAGER_
ENDPOINT)

# Assign the policy
r = policy_assign(SUBSCRIPTION, PROTECTION_SET, VIRTUAL_MACHINE,
                  POLICY, manager_endpoint_connection, headers)
print_response(r)
```

Provide feedback

For any suggestions and comments regarding this product or its documentation, send us an e-mail to:

info@hycu.com

We will be glad to hear from you!

www.hycu.com

HYCU®