200-901^{Q&As}

Developing Applications and Automating Workflows using Cisco Platforms (DEVASC)

Pass Cisco 200-901 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.leads4pass.com/200-901.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Cisco Official Exam Center

⚙ **Instant Download** After Purchase

⚙ **100% Money Back** Guarantee

⚙ **365 Days** Free Update

⚙ **800,000+** Satisfied Customers

QUESTION 1

DRAG DROP

Drag and drop the code snippets from the bottom into the Python script to write API output to a csv file. Not all options are used.

Select and Place:

```
#!/usr/bin/env python3
import csv
import json
import [          ]
#Create a request to grab info in api
response = requests. [          ] ('http://api.zippopotam.us/us/80301')

try:
    with open('80301.json', 'w') as outfile:
            json.dump(response.json(), outfile)
    print( [          ] .json())
except Exception as e:
    print(e)

with open('80301.csv', 'w') as outfile:
            json.dump(response.json(), outfile)
print(response. [          ] )
```

response	requests	request
get	status_code	status

Correct Answer:

```
#!/usr/bin/env python3
import csv
import json
import   requests
#Create a request to grab info in api
response = requests.   get      ('http://api.zippopotam.us/us/80301')

try:
    with open('80301.json', 'w') as outfile:
            json.dump(response.json(), outfile)
    print(   response   .json())
except Exception as e:
    print(e)

with open('80301.csv', 'w') as outfile:
            json.dump(response.json(), outfile)
print(response.   status_code  )
```

```
request
status
```

QUESTION 2

A developer created a new secure API and is testing the functionality locally. The API was added to a container for further testing, and other team members and applications need to connect to the API. Which command exposes the API in the container as port 8443 and allows connections external to the host?

A. docker run -p 127.0.0.1:80:8443/tcp myapiservice

B. docker run -p 0.0.0.0:443:8443/tcp myapiservice

C. docker run -p 0.0.0.0:8443:443/tcp myapiservice

D. docker run -p 127.0.0.1:443:8443/tcp myapiservice

Correct Answer: B

To expose the API in a container on port 8443 and allow external connections to the host on port 443, you can use the -p flag with docker run to publish a specific port from the container to the host.

This will expose the API running in the container on port 8443, and allow external connections to the host on port 443. The port number on the left side of the colon (443) is the port on the host, and the port number on the right side (8443) is the port inside the container.
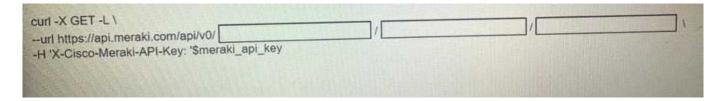
QUESTION 3

FILL IN THE BLANK

Refer to the exhibit.

```
List the networks in a Meraki organization:

GET https://api.meraki.com/api/v0/organizations/<org_id>/networks
Response code: 200
Response body: [ {
    "id": <network_id>,
    "name": "Test Network",
    "organization_id": <new_org_id>,
    "type": "wireless",
    "timeZone": "America/Los_Angeles",
    "tags": " test "
} ]
```

Fill in the blanks to complete the cURL command to the list of networks in the Meraki organization with an id of
384279060

```
curl -X GET -L \
--url https://api.meraki.com/api/v0/ [          ] / [          ] / [          ] \
-H 'X-Cisco-Meraki-API-Key: '$meraki_api_key
```

A. Check the answer in explanation.

Correct Answer: A

organizations , 384279060 , networks

QUESTION 4

Refer to the exhibit.

```
 1 import requests
 2 import json
 3
 4 BASE_URL = "https://<IP Address>"
 5 FIRST_URL = "/dna/intent/api/v1/network-device"
 6 SECOND_URL = "/dna/intent/api/v1/network-device-poller/cli/read-request"
 7 THIRD_URL = "/dna/intent/api/v1/task/{task_id}"
 8 FOURTH_URL = "/dna/intent/api/v1/file/{file_id}"
 9
10 headers = {"X-Auth-Token": "TOKEN", "Content-Type": "application/json"}
11
12 params = {"platformId": "C9500-40X"}
13 response = requests.get(BASE_URL + FIRST_URL, headers=headers, params=params)
14 devices = []
15 for device in response.json()["response"]:
16     devices.append(device["id"])
17
18     payload = {
19         "commands": ["show version", "show ip int brief"],
20         "deviceUuids": devices,
21         "timeout": 0,
22     }
23 response = requests.post(BASE_URL + SECOND_URL, data=json.dumps(payload),
   · headers=headers)
24 task_id = response.json()["response"]["taskId"]
25
26 response = requests.get(BASE_URL + THIRD_URL.format(task_id=task_id),
   · headers=headers)
27 progress_json = json.loads(response.json()["response"]["progress"])
28 file_id = progress_json("fileId")
29
30 response = requests.get(BASE_URL + FOURTH_URL.format(file_id=file_id),
   · headers=headers)
31 file_json = response.json()
32
33 for cmd in file_json:
34     print(file_json[cmd]["commandResponses"]["SUCCESS"]["show ip int brief"])
```

A network engineer must collect information from the network using Cisco DNA Center APIs. The script must query for the devices with \\'platformId\\' \\'C9540-x\\' and then send commands to be executed. Which process is automated by the Python script?

A. results gathering of the show version and show ip int brief commands

B. output of show ip int brief command on devices of type C9500-40X

C. execution information of the show version and show ip int brief commands

D. list of executed commands on C9500-40X and the information of show ip int brief command

Correct Answer: B

QUESTION 5

Refer to the exhibit.

Generating Client ID and API Key:

• Log in to your AMP for Endpoints Console.

• Go to Accounts > Business Settings.

• Click Configure API Credentials under Features to generate the Client ID and secure API Key.

Once you have the API client ID and API key, you can make the API calls as follows:

```
https://<your_client_id>:<your_api_key>@<api_endpoint>
```

Alternatively you can use Basic HTTP Authentication. Base 64 encode the string ":", and send that prefixed with the string "Basic" as the authorization header. For instance, if your client_id was 1234, and your api_key was "atest", then it would be base64 encoded to "MTIzNDphdGVzdA==", and your header would be:

```
Authorization: Basic MTIzNDphdGVzdA==
```

The documentation outlines how to use credentials for the AMP API. Which script successfully performs an API call using basic authentication?

A.
```
import requests
amp_auth_token = 'a1b2c3d4e5f6g7h8i9j0 k1l2m3n4o5p6'
url = 'https://api.amp.cisco.com/v1/version'
request = requests.get(url,
    auth=(amp_auth_token))
print(request.json())
```

B.
```
import requests
amp_client_id = 'a1b2c3d4e5f6g7h8i9j0'
amp_api_key = 'a1b2c3d4-e5f6-g7h8-i9j0-k1l2m3n4o5p6'
url = 'https://api.amp.cisco.com/v1/version'
request = requests.get(url,
    auth=(amp_client_id, amp_api_key))
print(request.json())
```

C.
```
import requests
amp_client_id = 'a1b2c3d4e5f6g7h8i9j0'
amp_api_key = 'a1b2c3d4-e5f6-g7h8-i9j0-k1l2m3n4o5p6'
url = 'https://api.amp.cisco.com/v1/version'
request = requests.get(url)),
print(request.json())
```

D.
```
import requests
amp_client_id = 'a1b2c3d4e5f6g7h8i9j0'
amp_login_password = '10g1n_p@ssw0rd'
url = 'https://api.amp.cisco.com/v1/version'
request = requests.get(url,
    auth=(amp_client_id, amp_login_password))
print(request.json())
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: B

[200-901 PDF Dumps](#)   [200-901 Practice Test](#)   [200-901 Exam Questions](#)