



ONAP

OPEN NETWORK AUTOMATION PLATFORM

ONAP Benchmark Project Progress

Oct. 17th, 2018

We created a mock openstack service that can interact with vcpe autotest script and ONAP,
 The mock server overcomes the difficulty of consuming resources and creating delay in real openstack .
 Test under different concurrent pressures in mock openstack environment(creat vBRG concurrently), here is the result:

Benchmark Performance and Stability Test Result								
No.	log No.	Concurrent interval	Execution count	success count in vcpe log	creat success in openstack	failed count	failed error tpye	performance (JProfiler)
1	13	0s	10	10	20	0	no	
2	15	0s	20	20	20	0	no	
3	16	0s	30	30	29	1	type1	
4	17	0s	40	40	39	1	type1	
5	18	0s	50	50	35	15	type2, type3	
6	14	0s	50	50	20	30	type1 , type2, type3	

Error type specification

<p style="text-align: center;">error type1 (Error caused by concurrent reading and writing files in vcpe autotest script)</p>	<pre>Traceback (most recent call last): File "vcpe.py", line 233, in <module> deploy_brg_only() File "vcpe.py", line 71, in deploy_brg_only vcpecommon.increase_ip_address_or_vni_in_template(vnf_template_file, ['vbrgemu_private_ip_0']) File "/home/ubuntu/autotest-test/vcpecommon.py", line 484, in increase_ip_address_or_vni_in_template json_data = json.load(json_input) File "/usr/lib/python2.7/json/__init__.py", line 291, in load **kw) File "/usr/lib/python2.7/json/__init__.py", line 339, in loads return _default_decoder.decode(s) File "/usr/lib/python2.7/json/decoder.py", line 364, in decode obj, end = self.raw_decode(s, idx=_w(s, 0).end()) File "/usr/lib/python2.7/json/decoder.py", line 382, in raw_decode raise ValueError("No JSON object could be decoded") ValueError: No JSON object could be decoded raise ValueError("No JSON object could be decoded") ValueError: No JSON object could be decoded</pre>
<p style="text-align: center;">error type2</p>	<pre>"requestId": "d9e05cf6-7cae-49be-a831-37508387505b", "requestScope": "vfModule", "requestStatus": { "finishTime": "Tue, 16 Oct 2018 06:51:18 GMT", "percentProgress": 100, "requestState": "FAILED", "statusMessage": "SDNC Callback Timeout Error" }, "requestType": "createInstance", "startTime": "Tue, 16 Oct 2018 06:44:47 GMT"</pre>
<p style="text-align: center;">error type3</p>	<pre>2018-10-16 06:45:25,477 : req_id of vfmodule is: 2018-10-16 06:45:25,477 : instance_id of vfmodule is: 2018-10-16 06:45:25,477 : Error when checking SO request progress, invalid request ID: 2018-10-16 06:45:25,477 : Failed to create VF Module vcpe_vfmodule_vcpevspvbrg20180927a_201810160644269119. 2018-10-16 06:45:25,478 : response-code of request.post() in submit_create_req() is: 2018-10-16 06:45:25,478 : 0 2018-10-16 06:45:25,478 : response from requests.post() in submit_create_req() is: 2018-10-16 06:45:25,478 : { "serviceException": { "messageId": "SVC2000", "text": "Request Failed due to BPEL error with HTTP Status= %1 \nRequest timedout, request id:8a0d4931-f88d-45b3-93b1-438c543dc6dd", "variables": ["502"] } }</pre>

Request for help

1. In both autotest script (vcpe) and robot script (vfw) , we found the problem of naming the service instance by creation time, which can lead to a lot of problems in concurrent testing. We suggest who maintain the two usecase scripts can improve the naming notations to ensure uniqueness in concurrent creation.