

```

1  oc login -u admin -p redhatocp https://api.ocp4.example.com:6443
2
3  #oc describe is httpd -n openshift
4
5  Answer 1)
6
7  armstrong indionce
8  collins veraster
9  aldrin roonkere
10 jobs sestiver
11 wozniak glegunge
12
13 #rpm -qa httpd*
14 #sudo yum install httpd-tools -y
15 #htpasswd -c -b -B userfile deepak redhat
16 #htpasswd -b -B userfile rahul redhat
17 #htpasswd -b -B userfile ankita redhat
18 #htpasswd -b -B userfile shavya redhat
19 #htpasswd -b -B userfile divyanshi redhat
20 #cat userfile
21 #oc create secret generic ex280-idp-secret --from-file htpasswd=userfile -n
22 openshift-config
23 #oc get secrets -n openshift-config
24 #oc get oauths.config.openshift.io -o yaml > oauth_file.yaml
25 #vim oauth_file.yaml
26 #cat oauth_file.yaml
27     apiVersion: v1
28     items:
29         - apiVersion: config.openshift.io/v1
30             kind: OAuth
31             metadata:
32                 name: cluster
33             spec:
34                 identityProviders:
35                     - name: ex280-htpasswd
36                         htpasswd:
37                             fileData:
38                                 name: ex280-ldp-secret
39                         mappingMethod: claim
40                         type: HTPasswd
41
42 #oc replace -f oauth_file.yaml
43 #watch oc get pods -n openshift-authentication
44 #oc login -u deepak -p redhat
45 #oc login -u rahul -p redhat
46 #oc login -u ankita -p redhat
47 #oc login -u shavya -p rehat
48 #oc login -u shavya -p redhat
49 #oc login -u divyanshi -p redhat
50 #oc whoami
51 #cp .auth/ocp4-kubeconfig .kube/config
52 #oc whoami
53
54 Answer 2)
55 #oc adm policy remove-cluster-role-from-group self-provisioner
56 system:authenticated:oauth      (for lab only)
57 #oc adm policy add-cluster-role-to-user cluster-admin jobs
58 #oc login -u jobs -p sestiver
59 #oc adm policy add-cluster-role-to-user self-provisioner wozniak
60 #oc login -u wozniak -p glegunge
61
62 Answer 3)
63 #oc new-project apollo
64 #oc new-project manhattan
65 #oc new-project gemini
66 #oc new-project bluebook
67 #oc new-project titan
68 #oc adm policy add-role-to-user admin armstrong -n apollo
69 #oc adm policy add-role-to-user admin armstrong -n gemini

```

```

68 #oc adm policy add-role-to-user view wozniak -n titan
69
70 Answer 4)
71 #oc adm groups new commander
72 #oc adm groups new pilot
73 #oc adm groups add-users commander armstrong
74 #oc adm groups add-users pilot collins aldrin
75 #oc get groups
76 #oc adm policy add-role-to-group edit commander -n apollo
77 #oc adm policy add-role-to-group view pilot -n apollo
78
79 Answer 5)
80 #oc new-project manhattan
81 #oc create quota --help | grep hard
82 #oc create quota ex280-quota
--hard(cpu=2,memory=160Mi,pods=3,services=6,replicationcontrollers=3) --dry-run -o yaml >
quota-file.yaml
83 #vim quota-file.yaml           #update the request and limits in yaml
84     apiVersion: v1
85     kind: ResourceQuota
86     metadata:
87         creationTimestamp: null
88         name: ex280-quota
89     spec:
90         hard:
91             requests.cpu: "2"
92             limits.cpu: "2"
93             requests.memory: 16Mi
94             limits.memory: 16Mi
95             pods: "3"
96             replicationcontrollers: "3"
97             services: "6"
98         status: {}
99
100 #oc create -f quota-file.yaml
101 #oc describe quota
102
103 Answer 6)
104 #oc new-project gru
105 #oc new-app --name=minion --image=registry.ocp4.example.com:8443/ubi9/httpd-24:latest
106 #oc scale -h | grep repli
107 #oc scale --replicas=5 deployment.apps/minion
108 #oc get all
109
110 Answer 7)
111
112 kubectl run -i --tty load-generator --rm --image=busybox:1.28 --restart=Never -- /bin/sh
-c "while sleep 0.01; do wget -q -O- http://php-apache; done"
113
114 #oc new-project lerna
115 #wget https://k8s.io/examples/application/php-apache.yaml
116 #oc create -f php-apache.yaml
117 #oc get pods          # Should be in error stage
118 #oc adm policy add-scc-to-user anyuid -z default
119 #oc get all
120 #oc delete pod/php-apache-5bdbb8dbf8-5kmmk
121 #oc set resources -h | grep deploy
122 #oc set resources deploy php-apache --requests=cpu=25mi --limits=cpu=180mi
123 #oc autoscale -h | grep min
124 #oc autoscale deployment php-apache --min=3 --max=9 --cpu-percent=60
125 #oc get all
126
127 Answer 8)
128
129 #oc new-project area51
130 #oc new-app --name oxcart --image registry.ocp4.example.com:8443/ubi9/httpd-24:latest
131 #openssl req -x509 -sha256 -nodes 365 -newkey rsa:4096 -keyout private.key -out
certificate.crt  # Learn
132 #oc get service

```

```

133 #oc create route edge --service oxcart --hostname=classified.lab.example.com
134 --cert=certificate.crt --key=private.key # Learn
135 #curl -kv https://classified.apps.ocp4.lab.example.com
136
137 Answer 9)
138 lab only
139 -----
140 podman search docker.io/hello-openshift
141 podman pull docker.io/openshift/hello-openshift
142 podman login registry.ocp4.example.com:8443 -u developer -p developer
143 podman tag docker.io/openshift/hello-openshift
144 registry.ocp4.example.com:8443/openshift/hello-openshift
145 podman push registry.ocp4.example.com:8443/openshift/hello-openshift
146
147 #oc new-project acid
148 #oc new-app --name=phosphoric
149 --image=registry.ocp4.example.com:8443/openshift/hello-openshift or
150 registry.ocp4.example.com:8443/ubi9/httpd-24:latest
151 #oc get all
152 #oc expose service phosphoric --hostname phosphoric.apps.ocp4.example.com
153 #curl http://phosphoric.apps.ocp4.example.com
154 #oc create configmap sedicen --from-literal RESPONSE="Soda pop won't stop"
155 #oc set env deployment phosphoric --from=cm/sedicen
156 #oc get pods
157 #curl http://phosphoric.apps.ocp4.example.com
158
159 Answer 10)
160
161 #oc new-project ascii-movie
162 #helm repo add ascii-movie-repo http://helm.ocp4.example.com/charts      # lab only
163 #helm search repo ascii-movie-repo
164 #helm install <application-name> ascii-movie-repo/etherpad
165 #oc get all
166 #telnet ascii-movie-etherpad-ascii-movie.apps.ocp4.example.com
167
168 Answer 11)
169
170 #oc new-project math
171 #echo redhat-password | base64
172 #oc create secret generic magic --from-literal decoder_ring=cmVkaGF0LXBhc3N3b3JkCg==
173 #oc get secret
174
175 Answer 12)
176
177 #oc new-project math
178 #oc new-app --name=qed --image=httpd
179 #oc expose service qed
180 #oc get routes
181 #oc get secrets
182 #oc set env deployment qed --from=secret/magic
183 #oc get pods
184 #oc get routes
185 #curl http://qed-math.apps.ocp4.example.com
186
187 Answer 13)
188
189 #oc new-project apples
190 #oc create serviceaccount ex280sa
191 #oc adm policy add-scc-to-user anyuid -z ex280sa
192 #oc get serviceaccounts
193
194 Answer 14)
195
196 #oc new-app --name=orange
197 --image=registry.ocp4.example.com:8443/redhattraining/gitlab-ce:8.4.3-ce.0
198 #oc get all      # pod is in error stage
199 #oc set serviceaccount deployment/orange ex280sa      # Pod start running
200 #oc expose service orange --port 80      # Still application not access, and might already
201 configured
202 #oc get pod --show-labels

```

```

196 #oc describe service orage      # Endpoints are <none>
197 #oc edit  service/orange          #Updated selector on service accrout same as pod
198 label
199
200 #oc describe service/orange
201
202 Answer 15)
203
204 #oc new-project mercury
205 #oc new-app --name atles --image registry.ocp4.example.com:8443/ubi9/httpd-24:latest
206 --as-deployment-config --dry-run -o yaml > myfile.yaml
207 #vim myfile.yaml
208     : Update the below lines in container session
209     resources:
210         requests:
211             memory: 800Gi
212
213 #oc create -f myfile.yaml
214 #oc get pods      # should be in pending stage
215 #oc edit dc/atles
216     : Remove the resources details
217 #oc get pods      # Pod start running
218
219 Answer 16)
220
221 #oc new-project database
222 #oc new-project checker
223 #oc create deployment mysql
224 --image=registry.ocp4.example.com:8443/rhscl/mysql-80-rhel7:latest -o yaml > mysql.yaml
225
226 [student@workstation ~]$ cat mysql.yaml
227     apiVersion: apps/v1
228     kind: Deployment
229     metadata:
230         creationTimestamp: null
231         labels:
232             app: mysql
233             name: mysqlsource
234         spec:
235             replicas: 1
236             selector:
237                 matchLabels:
238                     network.openshift.io/policy-group: source
239             strategy: {}
240             template:
241                 metadata:
242                     creationTimestamp: null
243                     labels:
244                         network.openshift.io/policy-group: source
245             spec:
246                 containers:
247                     - image: registry.ocp4.example.com:8443/rhscl/mysql-80-rhel7:latest
248                     name: mysql-80-rhel7
249                     env:
250                         - name: MYSQL_ROOT_PASSWORD
251                         value: redhat
252
253 [student@workstation ~]$
254
255 #oc create -f mysql.yaml -n database
256 #oc get all -n database
257 #oc label ns checker team=devsecops
258 #oc get ns checker --show-labels
259
260 [student@workstation ~]$ cat mysql.yaml
261     apiVersion: apps/v1
262     kind: Deployment
263     metadata:
264         creationTimestamp: null
265         labels:
266             app: mysql

```

```

262     name: mysqlclient
263   spec:
264     replicas: 1
265     selector:
266       matchLabels:
267         deploymentb-sql: target
268     strategy: {}
269   template:
270     metadata:
271       creationTimestamp: null
272     labels:
273       deploymentb-sql: target
274   spec:
275     containers:
276       - image: registry.ocp4.example.com:8443/rhscl/mysql-80-rhel7:latest
277         name: mysql-80-rhel7
278         env:
279           - name: MYSQL_ROOT_PASSWORD
280             value: redhat
281 [student@workstation ~]$

282 #oc create -f mysql.yaml -n checker
283 #oc get all -n checker
284
285
286 Network pdf -> seach "NetworkPolicy" and patch "ingress" -> create net.yaml file and
287 paste the yaml content
288
289 [student@workstation ~]$ cat net.yaml
290 kind: NetworkPolicy
291 apiVersion: networking.k8s.io/v1
292 metadata:
293   name: db-allow-sysql-conn    # Network Policy Name, given in exam
294   namespace: database          # Project Name
295 spec:
296   podSelector:
297     matchLabels:
298       network.openshift.io/policy-group: source  # Source Pod label (oc get pod
299         -n database --show-labels)
300   ingress:
301     - from:
302       - namespaceSelector:
303         matchLabels:
304           team: devsecops      # Client Project label (oc get checker
305             --show-labels)
306         podSelector:
307           matchLabels:
308             deploymentb-sql: target  # Client Pod label (oc get pod -n checker
309               --show-labels)
310   ports:
311     - port: 8080
312       protocol: TCP
313 [student@workstation ~]$

314 #oc create -f net.yaml          => After validate pod running on checker namespace
315 start working in Exam,
316
317 Test:
318 #oc get pods -n database -o wide    ==> Note down IP address
319 #oc project checker
320 #oc rsh <Client Pod Name>
321   mysql -u root -h <IP Address> -p      ==> connection successful
322
323 Create Deploy one more application with same mysql.yaml, to change matchLabels and
324 labels valumes
325 #vim mysql.yaml
326   spec:
327     replicas: 1
328     selector:

```

```

325     matchLabels:
326         sqlclient: other
327     strategy: {}
328     template:
329         metadata:
330             creationTimestamp: null
331         labels:
332             sqlclient: other
333
334 #oc create -f mysql.yaml -n checker
335 #oc rsh <Other Pod Name>
336     mysql -u root -h <IP Address> -p          ==> connection Failed
337
338 Answer 17)
339
340 #oc adm create-bootstrap-project-template -o yaml > tempfile.yaml
341 #vim tempfile.yaml
342 #oc create -f tempfile.yaml -n openshift-config      # Search LimitRange in documents
and update in last
343     - apiVersion: v1
344         kind: LimitRange
345         metadata:
346             name: ${PROJECT_NAME}-Limits
347             namespace: ${PROJECT_NAME}
348         spec:
349             limits:
350                 - default:
351                     memory: 512Mi
352                 defaultRequest:
353                     memory: 256Mi
354                 max:
355                     memory: 512Mi
356                 min:
357                     memory: 128Mi
358             type: Container
359 #oc edit projects.config.openshift.io cluster           # Update projectRequestTemplate,
refer documents. update in spec
360         spec:
361             projectRequestTemplate:
362                 name: project-request
363 #watch oc get pod -n openshift-apiserver
364 #oc new-project test
365 #oc new-project test
366 #oc get limitranges
367 #oc describe limitranges
368
369 Answer 18)
370 Install : "file integrity Operator" via GUI
371
372 Answer 19)
373 #oc new-project elementum
374 #oc create sa magna
375 #oc create cronjob -h | grep -i create
376 #oc create cronjob job-runner
--image=registry.ocp4.example.com:8443/rhscl/mysql-80-rhel7:latest --schedule="05 04 02
* * "
377 #oc edit cronjobs.batch job-runner      #Update Successful Job History Limit"
378 #oc describe cronjobs.batch job-runner
379
380 Answer 20)
381
382 #mkdir dump
383 #cd dump/
384 #oc adm must-gather --dest-dir=.
385 #tar -czvf ex280-ocp-b1d661ca-7fb3-42e2-a62a-968b80672189.tar.gz .
386 #/usr/local/bin/upload ~/dump/ex280-ocp-b1d661ca-7fb3-42e2-a62a-968b80672189.tar.gz
387
388 Answer 21)
389

```

```

390 #oc get sc
391 #oc describe sc <StorageClass Name> ==> Note Down ReclaimPolicy, NFS_PATH and NFS_SERVER
392
393 #oc get pod -A | grep nfs ==> Lab only
394 #oc describe pod/nfs-client-provisioner-b4cc59f59-h7pht -n nfs-client-provisioner ==>
Lab only, noted NFS_SERVER:192.168.50.254, NFS_PATH:/exports-ocp4, ReclaimPolicy:Delete
395
396 Search "PersistentVolumes" in document and create pv.yaml fileData
397
398 [student@workstation ~]$ cat pv.yaml
399     apiVersion: v1
400     kind: PersistentVolume
401     metadata:
402         name: landing-pv      #Given in Exam
403     spec:
404         capacity:
405             storage: 16Mi      #Given in Exam
406         volumeMode: Filesystem
407         accessModes:
408             - ReadOnlyMany      #Given in Exam
409         persistentVolumeReclaimPolicy: Delete      #Get is from (oc describe sc sc_name)
410         nfs:
411             path: /exports-ocp4      #Get is from (oc describe sc sc_name)
412             server: 192.168.50.254    #Get is from (oc describe sc sc_name)
413 [student@workstation ~]$
414
415 #oc create -f pv.yaml
416 #oc get pv
417 #oc new-project page
418 #oc new-app --name=landing
--image=registry.ocp4.example.com:8443/ubi9/httpd-24:latest      (Replace image name
given in exam)
419 #oc get deployment
420 #oc scale --replicas 3 deployment landing
421 #oc create route edge --service landing --hostname
https://.....domain20.example.com      (In Exam only)
422 #oc create route edge --service
landing                                (In Lab only)
423 #oc set volume deployment/landing --add --name anyname --type pvc --claim-mode rwm
--claim-size 16Mi --mount-path /usr/share/nginx/html --claim-name landing-pvc
424
425 #oc rsh pod/landing-<podname>      (Validate)
426     df -h /usr/share/nginx/html      (should be mount through NFS)
427 #GUI firefox Web page should be open through https://
428
429 Answer 22)
430
431 #oc new-project mercury
432 #oc new-app --name=atlas
--image=registry.ocp4.example.com:8443/ubi9/httpd-24:latest      # till here for
exam
433 login GUI -> workload -> Deployment -> Action -> addhealthcheck -> update "Liveness" ->
Change Type : "TCP Socket & 8080"
    -> initial delay : "10" -> TimeOut : "30" -> Validate and Done
434
435
436 End of Exam
437 #oc get secret | grep kubeadmin
438 #oc delete secret kubeadmin -n kube-system

```