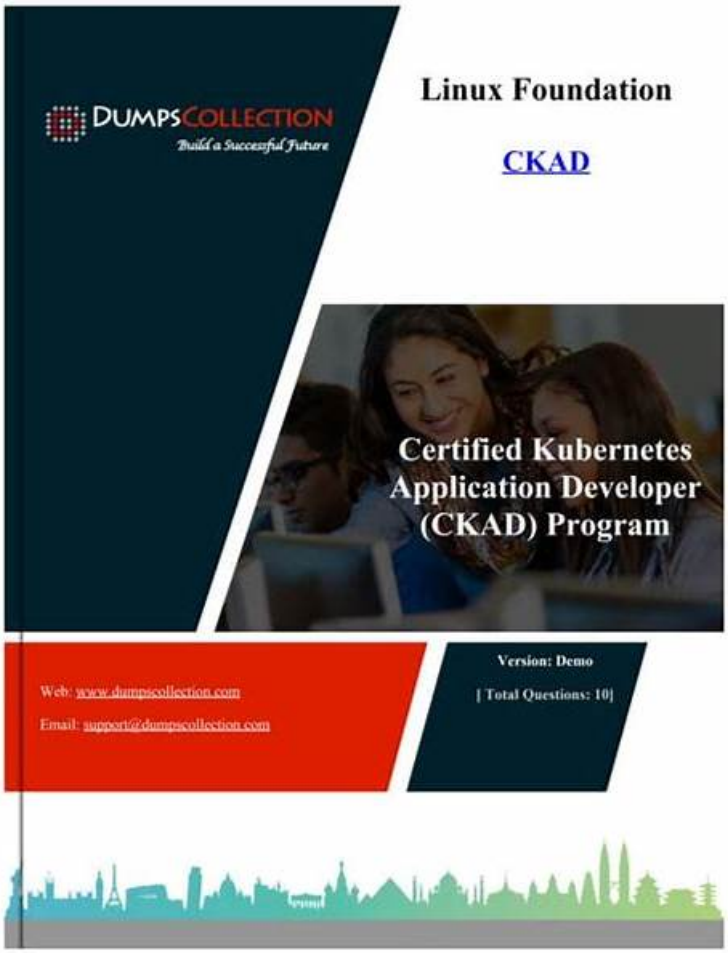
Linux Foundation CKAD시험덤프공부, CKAD퍼펙트공부문제



그리고 Itcertkr CKAD 시험 문제집의 전체 버전을 클라우드 저장소에서 다운로드할 수 있습니다: https://drive.google.com/open?id=1fkr6JUT2r7Y3X7AgFdYM2xtqrT0mWbk1

Itcertkr에서 출시한 Linux Foundation 인증 CKAD시험덤프는Itcertkr의 엘리트한 IT전문가들이 IT인증실제시험문제를 연구하여 제작한 최신버전 덤프입니다. 덤프는 실제시험의 모든 범위를 커버하고 있어 시험통과율이 거의 100%에 달합니다. 제일 빠른 시간내에 덤프에 있는 문제만 잘 이해하고 기억하신다면 시험패스는 문제없습니다.

리눅스 재단 인증 쿠버네티스 응용 프로그램 개발자(CKAD) 시험은 쿠버네티스와 함께 작업하고자 하는 개발자들의 기술과 지식을 시험하는 인증 프로그램입니다. 쿠버네티스는 컨테이너화된 응용 프로그램의 배포, 확장, 관리를 자동화하는 오픈소스 컨테이너 오케스트레이션 시스템입니다. CKAD 시험은 개발자의 쿠버네티스 응용 프로그램을 설계, 구축, 구성, 노출하는 능력을 시험하는 것을 목표로 합니다.

>> Linux Foundation CKAD시험덤프공부 <<

CKAD퍼펙트 공부문제, CKAD높은 통과율 시험덤프문제

Itcertkr선택으로Linux Foundation CKAD시험을 패스하도록 도와드리겠습니다. 우선 우리Itcertkr 사이트에서Linux Foundation CKAD관련자료의 일부 문제와 답 등 샘플을 제공함으로 여러분은 무료로 다운받아 체험해보실 수 있습니다. 체험 후 우리의Itcertkr에 신뢰감을 느끼게 됩니다. Itcertkr에서 제공하는Linux Foundation CKAD덤프로 시험 준비하세요. 만약 시험에서 떨어진다면 덤프전액환불을 약속 드립니다.

CKAD 자격증 시험은 Kubernetes 애플리케이션 개발에 대한 개발자들의 기술과 전문성을 선보이는 뛰어난 방법입

니다. 이는 또한 경력 전망을 향상시키고 취업 기회를 늘리는 좋은 방법이기도 합니다. Kubernetes는 컨테이너 오케스트레이션의 표준으로 빠르게 성장하고 있으며, 숙련된 Kubernetes 개발자에 대한 수요가 증가하고 있습니다. CKAD 자격증은 개발자의 이력서에 가치 있는 추가 되며 경쟁적인 취업 시장에서 눈에 띄게 도움이 됩니다.

리눅스 재단은 개발자들이 CKAD 자격증 시험에 대비할 수 있도록 다양한 교육 과정과 자원을 제공합니다. 이러한 자원은 온라인 강좌, 학습 지침서, 모의 시험 및 실습 랩을 포함합니다. 리눅스 재단은 또한 CKAD 자격증 시험을 신청할 수 있는 인증 시험 쿠폰을 제공합니다. 이 쿠폰은 첫 번째 시도에서 시험에 통과하지 못한 경우 한 번의 무료 재시도가 가능합니다.

최신 Kubernetes Application Developer CKAD 무료샘플문제 (Q40-Q45):

질문 # 40
You are running a critical application in Kubernetes tnat requires nign availability and IOW latency. The application uses a statefulset With 3 replicas, each consuming a large amount of memory. You need to define resource requests and limits for the pods to ensure that the application operates smoothly and doesn't get evicted due to resource constraints.

정답：

설명：
See the solution below with Step by Step Explanation.
Explanation:
Solution (Step by Step) :
1. Determine Resource Requirements:
- Analyze tne application's memory usage. Determine tne average memory consumption per pod and the peak memory usage.
- Consider the resources available on your Kubernetes nodes.
- Define realistic requests and limits based on the application's needs and available node resources.
2. Define Resource Requests and Limits in the StatefulSet:
- Update the StatefulSet YAML configuration with resource requests and limits for the container.
- requests: Specifies the minimum amount of resources the pod will request
- limits: Specifies the maximum amount of resources the pod can use.

```
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: my-critical-app
spec:
  serviceName: "my-critical-app"
  replicas: 3
  selector:
    matchLabels:
      app: my-critical-app
  template:
    metadata:
      labels:
        app: my-critical-app
    spec:
      containers:
      - name: my-critical-app
        image: my-app-image:latest
        resources:
          requests:
            memory: "2Gi"
            cpu: "1" # 1 CPU core
          limits:
            memory: "4Gi"
```

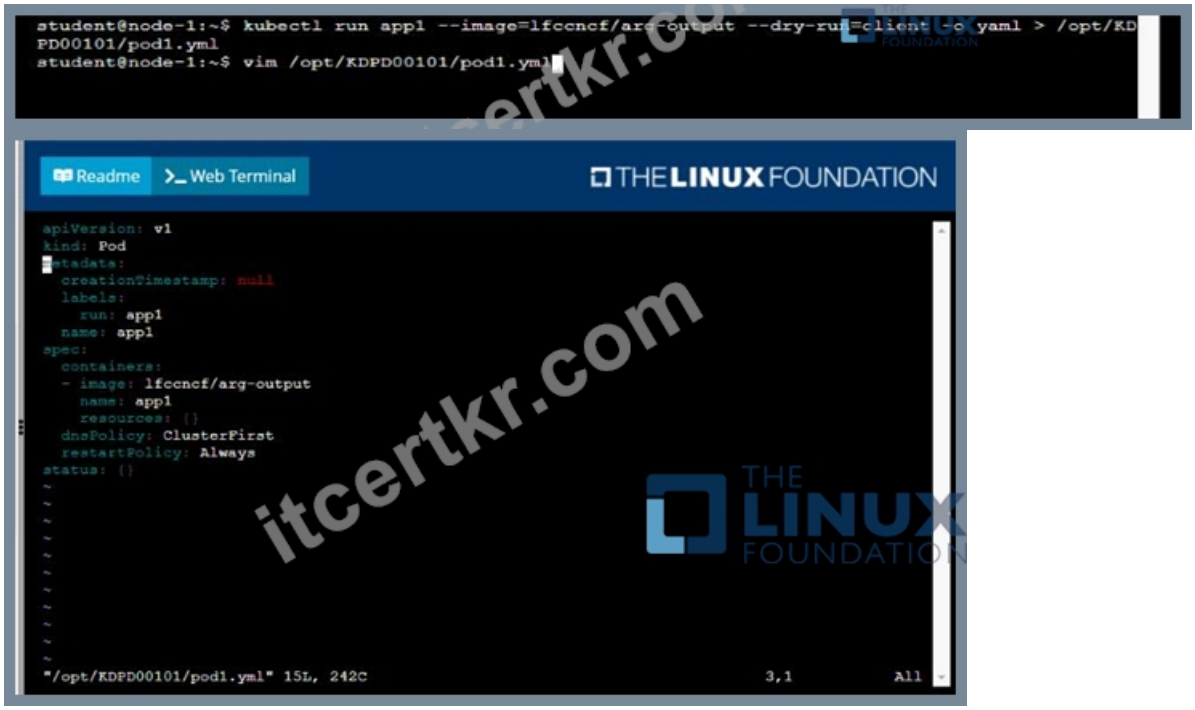
3. Apply the StatefulSet Configuration: - Apply the updated StatefulSet configuration to your Kubernetes cluster: bash kubectl apply -f my-critical-app-statefulset.yaml 4. Monitor Resource Usage: - Use 'kubectl describe pod' to monitor the resource usage of the pods. - Ensure that the pods are utilizing the requested resources and not exceeding the limits.

질문 # 41
Context
Anytime a team needs to run a container on Kubernetes they will need to define a pod within which to run the container.
Task
Please complete the following:
* Create a YAML formatted pod manifest
/opt/KDPD00101/podl.yml to create a pod named app1 that runs a container named app1cont using image lfccncf/arg-output with

these command line arguments: -lines 56 -F
* Create the pod with the kubect1 command using the YAML file created in the previous step
* When the pod is running display summary data about the pod in JSON format using the kubect1 command and redirect the output to a file named /opt/KDPD00101/out1.json
* All of the files you need to work with have been created, empty, for your convenience

정답：

설명：
Solution:



질문 # 42
You have a Deployment named 'my-app-deployment running a Flask application. You want to add a liveness probe that checks if the Flask application is responding on port '5000' and a readiness probe that checks if the application is ready to receive requests. Implement these probes using Kustomize.

정답：

설명：
See the solution below with Step by Step Explanation.
Explanation:
Solution (Step by Step) :
1. Create a base Deployment configuration:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-app-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: my-app
  template:
    metadata:
      labels:
        app: my-app
    spec:
      containers:
      - name: my-app
        image: my-app:latest
        ports:
        - containerPort: 5000
```

2. Create a 'kustomization.yamr file:

3. Create 'patcnes/liveness-probe.yaml':

```
spec:
  template:
    spec:
      containers:
      - name: my-app
        livenessProbe:
          tcpSocket:
            port: 5000
          initialDelaySeconds: 15
          periodSeconds: 20
          failureThreshold: 3
```

4. Create 'patches/readiness-probe-yaml':

```
spec:
  template:
    spec:
      containers:
      - name: my-app
        readinessProbe:
          tcpSocket:
            port: 5000
          initialDelaySeconds: 5
          periodSeconds: 10
          failureThreshold: 2
```

5. Apply the Kustomize configuration: bash kustomize . I kubectl apply -t- - Liveness probe: This probe checks if the application is still alive and running. It uses a TCP socket to connect to port ' 5000' and waits for 15 seconds before making the first cneck. It checks every 20 seconds, and if it fails 3 times in a row, the pod is restarted. - Readiness probe: This probe checks if the application is ready to receive requests. It also uses a TCP socket to connect to port '5000'. It checks every 10 seconds and waits for 5 seconds before the first check. If it fails 2 times in a row, the pod is marked as unhealthy and excluded trom receiving traffic. Note: Make sure your Flask application is actually listening on port '5000' and responding to requests. ,

질문 # 43
You are building a web application with two microservices: a frontend service ('frontend') and a backend service (' backend'). The frontend service requires access to the backend service, which iS exposed on port 8080 within the Kubernetes cluster. How would you configure an Ingress resource to direct traffic to the correct service based on the hostname, ensuring that the frontend service can access the backend service internally without exposing the backend service to the public internet?

정답：

설명：
See the solution below with Step by Step Explanation.
Explanation:

Solution (Step by Step) :
1. Create a Service tor the Backend Service:
- Define a Service for the 'backend' service, exposing it internally within the Kubernetes cluster on port 8080.

```
apiVersion: v1
kind: Service
metadata:
  name: backend-service
spec:
  ports:
  - port: 8080
    targetPort: 8080
  selector:
    app: backend
```

2. Configure the Ingress Resource: - Create an Ingress resource that directs traffic to the frontend service based on the hostname, allowing the frontend service to access the backend service internally without exposing it to the public internet - Define the Ingress rule to map the hostname 'frontend-example.com' to the 'frontend' service on port 80. - Configure an Ingress rule to enable access to the 'backend' service on port 8080 using the hostname 'internal-backend-example-com' within the Kubernetes cluster.

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: my-ingress
spec:
  rules:
  - host: frontend.example.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: frontend-service
            port:
              number: 80
  - host: internal-backend.example.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: backend-service
            port:
              number: 8080
  tls:
  - hosts:
    - frontend.example.com
    secretName: frontend-tls
```

3. Create a Secret for the Frontend TLS Certificate: - Create a Secret in Kubernetes to store the TLS certificate and key for the frontend service.

```
apiVersion: v1
kind: Secret
metadata:
  name: frontend-tls
type: tls
data:
  tls.crt:
  tls.key:
```

4. Apply the Resources: - Apply the Service, Ingress, and Secret YAML files to your Kubernetes cluster using 'kubectl apply -f 5. Access the Frontend Service: - Access the frontend service using the hostname 'frontend-example.com'. The frontend service can now access the backend service internally using the hostname 'internal-backend-example-com' without exposing the backend service to the public internet.]

질문 # 44
You are running a web application on a Kubernetes cluster, and you want to ensure that thecontainer running your application is protected from potential security vulnerabilities. You are specifically concerned about unauthorized access to the container's filesystem. Explain how you would implement AppArmor profiles to restrict access to the container's filesystem.

정답：

설명：
See the solution below with Step by Step Explanation.
Explanation:
Solution (Step by Step) :
1. Define the AppArmor Profile:
- Create a new AppArmor profile file, for example, 'nginx-apparmor.conf, within your Kubernetes configuration directory.
- Within this file, define the restrictions for the container.
- For instance, to allow access to specific directories and files:
include common AppArmor profile
include /etc/apparmor.d/abstractions/base/nginx.apparmor
Allow access to specific directories
/var/www/html r,
/etc/nginx r,
Allow access to specific files
/etc/nginx/nginx.conf r,
/usr/sbin/nginx r,
Deny access to all other files and directories
Deny
2. Load the AppArmor Profile:
- Use the create configmap' command to create a ConfigMap containing your AppArmor profile:
Bash
kubectl create configmap nginx-apparmor-profile --from-file=nginx-apparmor.conf
3. Apply the Profile to Your Deployment:
- Update your Deployment YAML file to include the AppArmor profile:

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:latest
        securityContext:
          # Enable AppArmor and specify the profile name
          appArmor: nginx-apparmor-profile
      # ... (rest of your Deployment YAML)
```


4. Restart the Pods: - Apply the updated Deployment YAML using 'kubectl apply -f nginx-deployment.yaml' - The updated deployment will restart the pods with the new AppArmor profile. 5. Verify the Profile: - Check the status of the pods with 'kubectl describe pod - Look for the "Security Context" section and verify that the AppArmor profile is correctly applied. 6. Test the

Restrictions: - Try to access files or directories that are not allowed by your AppArmor profile. - This will help you confirm that the profile is effectively restricting access.

질문 # 45
.....

CKAD퍼펙트 공부문제: https://www.itcertkr.com/CKAD_exam.html

- CKAD인증시험자료 ❤□ CKAD완벽한 시험공부자료 □ CKAD시험패스 인증공부자료 □ 「 www.pass4test.net 」에서{ CKAD }를 검색하고 무료 다운로드 받기CKAD완벽한 시험공부자료
- CKAD시험덤프공부 100%시험패스 인증덤프공부 □ ➥ www.itdumpskr.com □은⇒ CKAD ⇐무료 다운로드를 받을 수 있는 최고의 사이트입니다CKAD자격증문제
- CKAD인증덤프문제 □ CKAD시험기출문제 □ CKAD자격증문제 □ □ www.passtip.net □에서➽ CKAD □□를 검색하고 무료로 다운로드하세요CKAD최신시험
- 100% 합격보장 가능한 CKAD시험덤프공부 시험 ❤□ 오픈 웹 사이트➥ www.itdumpskr.com □검색「 CKAD 」무료 다운로드CKAD시험기출문제
- CKAD인증덤프공부문제 □ CKAD높은 통과율 시험대비자료 □ CKAD시험패스 가능 덤프문제 □ ☀ www.dumptop.com □☀□을 통해 쉽게□ CKAD □무료 다운로드 받기CKAD인증덤프문제
- CKAD퍼펙트 최신 덤프모음집 □ CKAD적중율 높은 인증시험덤프 □ CKAD퍼펙트 최신버전 덤프자료 □ □ [www.itdumpskr.com]에서 검색만 하면□ CKAD □를 무료로 다운로드할 수 있습니다CKAD최신버전 시험덤프공부
- CKAD시험덤프공부 덤프는 Linux Foundation Certified Kubernetes Application Developer Exam 시험패스의 유효 공부자료 □ ▸ CKAD ◂를 무료로 다운로드하려면[kr.fast2test.com]웹사이트를 입력하세요CKAD완벽한 공부문제
- CKAD인증덤프공부문제 □ CKAD최신버전 시험덤프공부 □ CKAD덤프문제 □ 무료로 다운로드하려면 □ www.itdumpskr.com □로 이동하여（ CKAD ）를 검색하십시오CKAD시험자료
- CKAD완벽한 시험공부자료 □ CKAD적중율 높은 인증시험덤프 □ CKAD최신시험 □ { kr.fast2test.com }에서➥ CKAD □를 검색하고 무료로 다운로드하세요CKAD최신버전 시험덤프공부
- CKAD시험패스 인증공부자료 □ CKAD인증시험자료 □ CKAD시험기출문제 □ 무료 다운로드를 위해 지금☀ www.itdumpskr.com □☀□에서➽ CKAD □검색CKAD적중율 높은 인증시험덤프
- CKAD덤프문제 □ CKAD자격증문제 □ CKAD퍼펙트 최신버전 덤프자료 □ 지금▹ www.itdumpskr.com ◃을(를) 열고 무료 다운로드를 위해▸ CKAD ◂를 검색하십시오CKAD시험자료
- agneszgtp519436.wikiparticularization.com, nellcrdb501111.life3dblog.com, wavesocialmedia.com, bookmarking1.com, bookmark-master.com, businessbookmark.com, bigboxdirectory.com, dianeywpe211619.liberty-blog.com, tayakqag513943.bloggactivo.com, safestructurecourse.com, Disposable vapes

그리고 Itcertkr CKAD 시험 문제집의 전체 버전을 클라우드 저장소에서 다운로드할 수 있습니다: https://drive.google.com/open?id=1fkr6JUT2r7Y3X7AgFdYM2xtqrT0mWbk1