

Passcert

Higher Quality, better service!



Q&A

[Http://www.passcert.com](http://www.passcert.com)

We offer free update service for one year.

Exam : EX328

Title : Red Hat Certified Specialist
in Building Resilient
Microservices exam

Version : DEMO

1.How do you create a new VirtualService to route HTTP traffic to different versions of a service based on headers in OpenShift Service Mesh?

A. See the Explanation.

Answer: A

Explanation:

1. Create the VirtualService YAML:

```
apiVersion: networking.istio.io/v1beta1 kind: VirtualService
```

```
metadata:
```

```
name: reviews-route namespace: bookinfo
```

```
spec:
```

```
hosts:
```

```
- reviews http:
```

```
- match:
```

```
- headers: end-user:
```

```
exact: jason route:
```

```
- destination: host: reviews subset: v2
```

```
- route:
```

```
- destination: host: reviews subset: v1
```

2. Apply with:

```
oc apply -f reviews-route.yaml
```

2.How do you define a DestinationRule to enable mutual TLS between services?

A. See the Explanation.

Answer: A

Explanation:

1. Create the DestinationRule YAML:

```
apiVersion: networking.istio.io/v1beta1 kind: DestinationRule
```

```
metadata:
```

```
name: ratings-destination namespace: bookinfo
```

```
spec:
```

```
host: ratings trafficPolicy:
```

```
tls:
```

```
mode: ISTIO_MUTUAL
```

2. Apply it:

```
oc apply -f ratings-destination.yaml
```

3.How can you create a PeerAuthentication policy to enforce mTLS across a namespace?

A. See the Explanation.

Answer: A

Explanation:

1. Define the policy in a YAML file:

```
apiVersion: security.istio.io/v1beta1 kind: PeerAuthentication
```

```
metadata:
```

```
name: default namespace: bookinfo
```

```
spec:
mtls:
mode: STRICT
2. Apply the file:
oc apply -f peer-auth.yaml
```

4.How do you enable tracing using Telemetry CR to export spans to Jaeger?
A. See the Explanation.

Answer: A

Explanation:

```
1. Create the Telemetry custom resource:
apiVersion: telemetry.istio.io/v1alpha1 kind: Telemetry
metadata:
name: telemetry-default namespace: istio-system
spec:
tracing: providers:
- name: "jaeger"
2. Apply the resource:
oc apply -f telemetry.yaml
```

5.How to apply a fault injection rule with VirtualService to simulate a delay?
A. See the Explanation.

Answer: A

Explanation:

```
1. Define the rule:
apiVersion: networking.istio.io/v1beta1 kind: VirtualService
metadata:
name: ratings-fault namespace: bookinfo
spec:
hosts:
- ratings http:
- fault:
delay: percentage: value: 100 fixedDelay: 5s
route:
- destination: host: ratings
2. Apply it:
oc apply -f fault-injection.yaml
```