1 CCA

Generated on: 2026-01-13 15:04:50 **Version:** 1.0

1.1 Table of Contents

1.2 Overview

The **Cilium Certified Associate (CCA)** exam demonstrates knowledge of eBPF-based networking, security, and observability with Cilium.

1.3 Exam Overview

Detail	Information
Exam Format	Multiple Choice
Number of Questions	60
Duration	90 minutes
Passing Score	75%
Certification Validity	3 years
Cost	$250 USD
Retake Policy	1 free retake

1.4 Exam Domains & Weights

Domain	Weight
Architecture	10%
Network Policy	20%

Domain	Weight
Service Mesh	10%
Network Observability	15%
Cluster Mesh	10%
eBPF	10%
BGP and External Networking	10%
Cilium CLI and Hubble	15%

1.5 Key Topics

1.5.1 Architecture

- Cilium components (agent, operator, CNI)
- eBPF datapath
- Identity-based security

1.5.2 Network Policy

- Cilium Network Policies
- L3/L4 and L7 policies
- Host policies

1.5.3 Observability

- Hubble for network visibility
- Flow logs and metrics
- Service maps

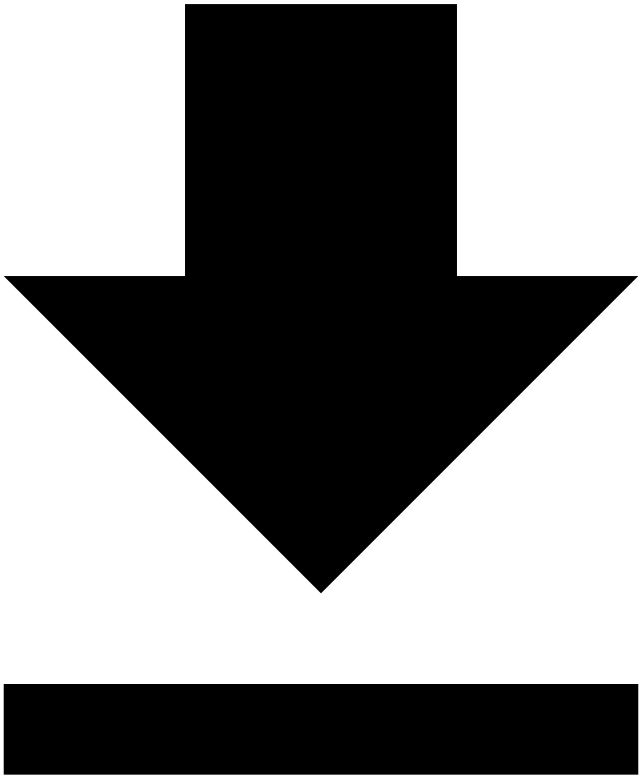
1.6 Study Resources

- Cilium Documentation
- CCA Curriculum
- Isovalent Labs

1.7 Navigation

- Next: Sample Questions →

1.8 Cilium Architecture



Download PDF Version

Comprehensive guide to Cilium networking for CCA certification.

1.9 Overview

Cilium is a CNI plugin that provides:

- **Networking** - Pod-to-pod connectivity using eBPF
- **Security** - Network policies with L3-L7 filtering
- **Observability** - Deep visibility with Hubble
- **Load Balancing** - Kubernetes service implementation

1.10 Architecture Components

1.10.1 Cilium Agent

- Runs on every node as DaemonSet
- Manages eBPF programs
- Implements network policies
- Handles service load balancing

1.10.2 Cilium Operator

- Cluster-wide operations
- IP address management (IPAM)
- Garbage collection

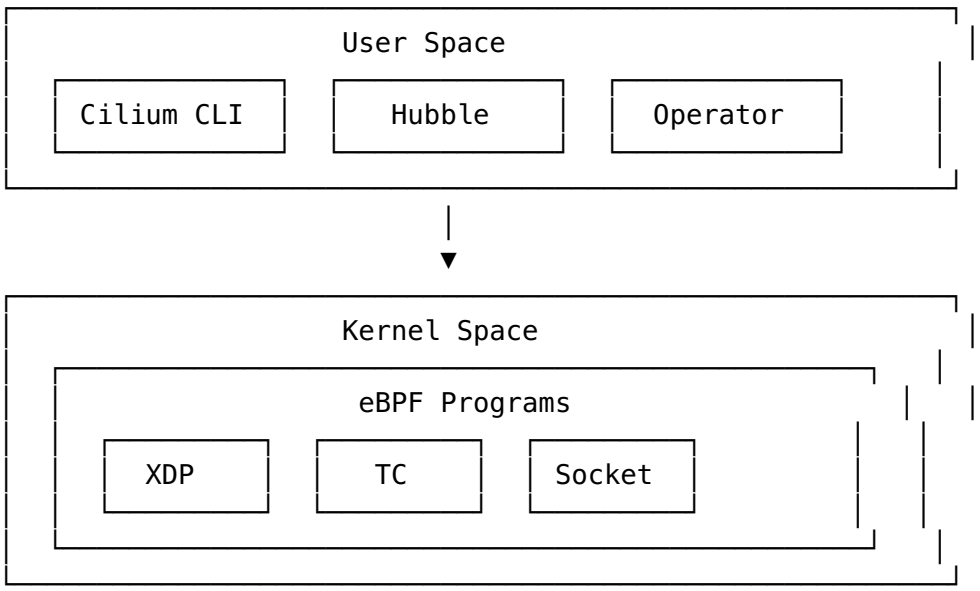
1.10.3 Hubble

- Observability platform
- Network flow visibility
- Service dependency maps

1.11 eBPF Technology

eBPF (extended Berkeley Packet Filter) enables:

- High-performance packet processing
- Programmable network datapath
- No kernel modifications needed
- Real-time observability



1.12 Installation

1.12.1 Using Cilium CLI

```
# Install Cilium CLI
curl -L --remote-name-all https://github.com/cilium/cilium-cli/
        releases/latest/download/cilium-linux-amd64.tar.gz
tar xzvf cilium-linux-amd64.tar.gz
sudo mv cilium /usr/local/bin

# Install Cilium
cilium install

# Check status
cilium status

# Enable Hubble
cilium hubble enable --ui
```

1.12.2 Using Helm

```
helm repo add cilium https://helm.cilium.io/
helm install cilium cilium/cilium --namespace kube-system
```

1.13 Network Policies

1.13.1 CiliumNetworkPolicy

```
apiVersion: cilium.io/v2
kind: CiliumNetworkPolicy
metadata:
  name: allow-frontend
  namespace: default
spec:
  endpointSelector:
    matchLabels:
      app: backend
  ingress:
  - fromEndpoints:
    - matchLabels:
        app: frontend
    toPorts:
    - ports:
      - port: "80"
        protocol: TCP
```

1.13.2 L7 Policy (HTTP)

```yaml
apiVersion: cilium.io/v2
kind: CiliumNetworkPolicy
metadata:
  name: l7-policy
spec:
  endpointSelector:
    matchLabels:
      app: api
  ingress:
  - fromEndpoints:
    - matchLabels:
        app: frontend
    toPorts:
    - ports:
      - port: "80"
        protocol: TCP
      rules:
        http:
        - method: GET
          path: "/api/v1/.*"
```

1.13.3 Cluster-wide Policy

```yaml
apiVersion: cilium.io/v2
kind: CiliumClusterwideNetworkPolicy
metadata:
  name: default-deny
spec:
  endpointSelector: {}
  ingress:
  - fromEntities:
    - cluster
```

1.14 Hubble Observability

1.14.1 Enable Hubble

```
cilium hubble enable
cilium hubble enable --ui
```

1.14.2 Hubble CLI

```
# Install Hubble CLI
export HUBBLE_VERSION=$(curl -s https://raw.githubusercontent.com/
        cilium/hubble/master/stable.txt)
```

```
curl -L --remote-name-all https://github.com/cilium/hubble/
        releases/download/$HUBBLE_VERSION/hubble-linux-
        amd64.tar.gz
tar xzvf hubble-linux-amd64.tar.gz
sudo mv hubble /usr/local/bin

# Port forward
cilium hubble port-forward &

# Observe flows
hubble observe
hubble observe --namespace default
hubble observe --pod frontend
hubble observe --verdict DROPPED
```

1.14.3 Hubble UI

```
cilium hubble ui
```

1.15 Service Load Balancing

Cilium replaces kube-proxy with eBPF-based load balancing:

```
# Install without kube-proxy
cilium install --set kubeProxyReplacement=strict

# Verify
cilium status | grep KubeProxyReplacement
```

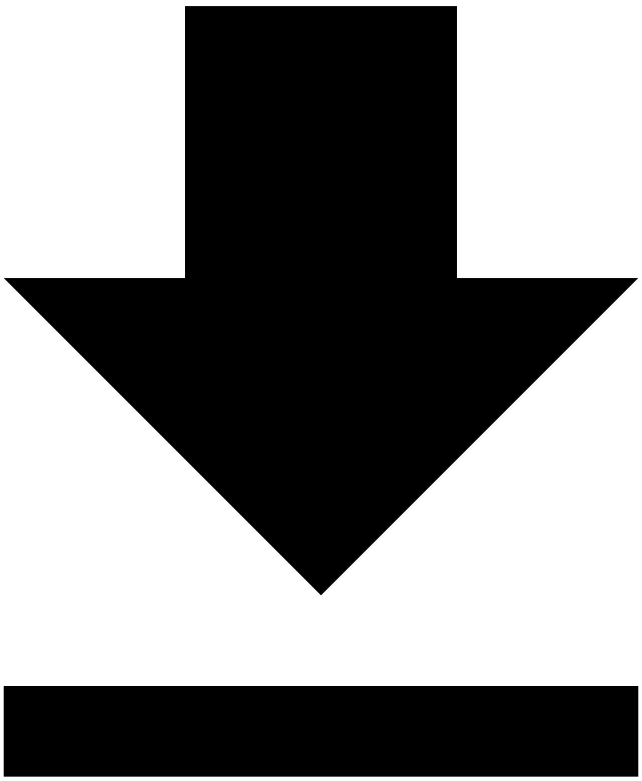
1.16 Useful Commands

```
# Status
cilium status
cilium status --verbose

# Connectivity test
cilium connectivity test

# Endpoint list
cilium endpoint list

# Service list
cilium service list

# BPF maps
cilium bpf lb list
cilium bpf ct list global
cilium bpf policy get
```

```
# Debug
cilium debuginfo
```

← Back to CCA Overview

1.17 Sample Practice Questions



Download PDF Version

1.18 Practice Resources

- Cilium Documentation
- Isovalent Labs

1.19 Network Policy (20%)

1.19.1 Question 1

Create a CiliumNetworkPolicy that allows only HTTP traffic from pods with label app=frontend to pods with label app=backend.

Show Solution

```yaml
apiVersion: cilium.io/v2
kind: CiliumNetworkPolicy
metadata:
  name: allow-frontend-http
spec:
  endpointSelector:
    matchLabels:
      app: backend
  ingress:
  - fromEndpoints:
    - matchLabels:
        app: frontend
    toPorts:
    - ports:
      - port: "80"
        protocol: TCP
      rules:
        http:
        - method: "GET"
```

1.19.2 Question 2

Create a policy that denies all egress traffic except DNS.

Show Solution

```yaml
apiVersion: cilium.io/v2
kind: CiliumNetworkPolicy
metadata:
  name: deny-egress-except-dns
spec:
  endpointSelector: {}
  egress:
  - toEndpoints:
    - matchLabels:
        k8s:io.kubernetes.pod.namespace: kube-system
        k8s-app: kube-dns
    toPorts:
    - ports:
      - port: "53"
        protocol: UDP
```

1.19.3 Question 3

Create an L7 policy that only allows GET requests to /api/v1/*.

Show Solution

```yaml
apiVersion: cilium.io/v2
kind: CiliumNetworkPolicy
metadata:
  name: l7-api-policy
spec:
  endpointSelector:
    matchLabels:
      app: api-server
  ingress:
  - toPorts:
    - ports:
      - port: "80"
        protocol: TCP
      rules:
        http:
        - method: "GET"
          path: "/api/v1/.*"
```

1.20 Hubble & Observability (15%)

1.20.1 Question 4

How do you enable Hubble in a Cilium installation?

Show Solution

```
# Using Helm
helm upgrade cilium cilium/cilium \
  --namespace kube-system \
  --set hubble.enabled=true \
  --set hubble.relay.enabled=true \
  --set hubble.ui.enabled=true

# Or using cilium CLI
cilium hubble enable --ui
```

1.20.2 Question 5

Use Hubble CLI to observe flows from namespace “production”.

Show Solution

```
hubble observe --namespace production
```

```
# With filters
hubble observe --namespace production --verdict DROPPED
hubble observe --namespace production --protocol TCP --port 80
```

1.20.3 Question 6

How do you view the service dependency map?

Show Solution

```
# Access Hubble UI
kubectl port-forward -n kube-system svc/hubble-ui 12000:80

# Or use CLI for service map data
hubble observe --namespace default -o json | jq
```

1.21 Cilium CLI (15%)

1.21.1 Question 7

Check the status of Cilium in the cluster.

Show Solution

```
cilium status

# Detailed status
cilium status --verbose
```

1.21.2 Question 8

Run connectivity tests to verify Cilium is working correctly.

Show Solution

```
cilium connectivity test
```

1.21.3 Question 9

List all Cilium endpoints in the cluster.

Show Solution

```
cilium endpoint list

# Or using kubectl
kubectl get ciliumendpoints -A
```

1.22 eBPF (10%)

1.22.1 Question 10

What is eBPF and why does Cilium use it?

Show Solution

eBPF (extended Berkeley Packet Filter) is a technology that allows running sandboxed programs in the Linux kernel without changing kernel source code.

Cilium uses eBPF for: - High-performance packet processing - Network policy enforcement at kernel level - Load balancing without kube-proxy - Deep observability without overhead - Security enforcement

1.22.2 Question 11

How does Cilium replace kube-proxy?

Show Solution

```
# Install Cilium with kube-proxy replacement
helm install cilium cilium/cilium \
  --namespace kube-system \
  --set kubeProxyReplacement=true \
  --set k8sServiceHost=<API_SERVER_IP> \
  --set k8sServicePort=<API_SERVER_PORT>
```

Cilium uses eBPF to implement: - ClusterIP services - NodePort services - LoadBalancer services - ExternalIPs

---

1.23 Cluster Mesh (10%)

1.23.1 Question 12

What is Cilium Cluster Mesh used for?

Show Solution

Cilium Cluster Mesh enables: - Multi-cluster connectivity - Cross-cluster service discovery - Global network policies - Shared services across clusters - High availability across clusters

1.23.2 Question 13

Enable Cluster Mesh on a Cilium installation.

Show Solution

```
# Enable Cluster Mesh
cilium clustermesh enable

# Connect to another cluster
cilium clustermesh connect --destination-context <other-cluster-
        context>

# Check status
cilium clustermesh status
```

1.24 BGP & External Networking (10%)

1.24.1 Question 14

Configure Cilium BGP peering.

Show Solution

```yaml
apiVersion: cilium.io/v2alpha1
kind: CiliumBGPPeeringPolicy
metadata:
  name: bgp-peering
spec:
  nodeSelector:
    matchLabels:
      bgp: enabled
  virtualRouters:
  - localASN: 65000
    exportPodCIDR: true
    neighbors:
    - peerAddress: "10.0.0.1/32"
      peerASN: 65001
```

1.25 Architecture (10%)

1.25.1 Question 15

What are the main components of Cilium?

Show Solution

1. **Cilium Agent** - Runs on each node, manages eBPF programs
2. **Cilium Operator** - Manages cluster-wide operations
3. **Cilium CNI** - Container Network Interface plugin
4. **Hubble** - Observability platform
5. **Cilium CLI** - Command-line tool

1.26 Exam Tips

1. **Know CiliumNetworkPolicy syntax** - L3/L4 and L7 policies
2. **Practice with Hubble** - Observing flows and troubleshooting
3. **Understand eBPF basics** - Why it's used, benefits
4. **Know cilium CLI commands** - status, connectivity test, endpoint list
5. **Understand identity-based security** - How Cilium identifies pods

---

← Back to CCA Overview
