FM-KED-006 — Kubernetes Cluster Certificate Expiration

Severity: S1 — Critical
Recovery Class: B — Standard Recovery
Covered by Monthly Support: Yes

Description

Internal Kubernetes certificates expire, leading to partial or complete cluster malfunction. This may affect control plane communication, node registration, API access, or workload scheduling.

This issue typically appears in long-running clusters where certificate rotation was not automated or monitored.

Typical Symptoms

- `kubectl` commands failing with TLS or x509 errors
- Nodes switching to `NotReady` state
- Control plane components restarting or failing
- Ingress, networking, or admission controllers malfunctioning

---

Diagnostic Checklist

Verify Certificate Expiration

On control plane node:

```
sudo kubeadm certs check-expiration
```

If `kubeadm` is not available, inspect certificates directly:

```
openssl x509 -in /etc/kubernetes/pki/apiserver.crt -noout -dates
```

Recovery Procedure

⚠ Perform these steps on the **control plane node**
⚠ Requires administrative access

---

1. Renew Kubernetes Certificates

```
sudo kubeadm certs renew all
```

This renews all cluster certificates managed by kubeadm.

---

2. Restart Control Plane Components

```
sudo systemctl restart kubelet
```

Kubernetes will automatically recreate static pods for:

- kube-apiserver
- kube-controller-manager
- kube-scheduler

---

3. Refresh Local kubeconfig Files

```
sudo cp /etc/kubernetes/admin.conf ~/.kube/config
sudo chown $(id -u):$(id -g) ~/.kube/config
```

Repeat for any other kubeconfig files in use.

---

4. Verify Cluster Health

```
kubectl get nodes
kubectl get pods -A
```

Ensure all nodes return to `Ready` state and system pods stabilize.

Preventive Notes

- Monitor certificate expiration dates regularly
- Schedule certificate renewal before expiration
- Prefer automated rotation where supported

- Avoid running clusters indefinitely without maintenance

Responsibility Boundary

Finmars SCSA provides best-effort operational guidance.
Clusters not managed via kubeadm or heavily customized may require additional investigation beyond standard support scope.

Revision #1
Created 2026-01-07 14:36:42 UTC by Sergei Zhitenev
Updated 2026-01-07 14:37:14 UTC by Sergei Zhitenev