devops-consulting.link

KUBERNETES CLUSTER AUDIT

production-demo

Security • Reliability • Networking • Operations

Audit Date
September 03, 2026 17:31 UTC

Report Status
Read-Only Assessment

Questions About This Report?
email@thetigran.com

Generated by DevOps Consulting · devops-consulting.link

Table of Contents

Executive Summary

This audit identified 2 CRITICAL and 11 HIGH severity finding(s) that require immediate attention, alongside 13 MEDIUM, 10 LOW, and 6 informational finding(s).

Metric	Count
Total findings	42
Critical	2
High	11
Medium	13
Low	10
Info	6

Areas Requiring Attention

- Identity & RBAC (1 CRITICAL, 3 HIGH, 2 MEDIUM)
- Workload Security (1 CRITICAL, 1 HIGH, 2 MEDIUM)
- Workloads (4 HIGH, 1 MEDIUM)
- Networking (5 MEDIUM)
- Secrets (1 HIGH)
- Cluster / Control Plane (1 HIGH)
- Resilience & Availability (1 MEDIUM)
- Storage (1 HIGH)

Findings Summary

Severity	Count
CRITICAL	2
HIGH	11
MEDIUM	13
LOW	10
INFO	6

Findings by Category

Category	CRITICAL	HIGH	MEDIUM	LOW	INFO	Total
Backup & Recovery	0	0	1	0	0	1
Cloud Provider	0	0	0	0	1	1
Cluster / Control Plane	0	1	0	0	1	2
Governance	0	0	0	2	0	2
Identity & RBAC	1	3	2	1	0	7
Networking	0	0	5	1	1	7
Observability	0	0	0	0	2	2
Pod Security Standards	0	0	1	0	0	1
Resilience & Availability	0	0	1	1	1	3
Secrets	0	1	0	1	0	2
Workload Security	1	1	2	3	0	7
Storage	0	1	0	0	0	1
Supply Chain	0	0	0	1	0	1
Workloads	0	4	1	0	0	5

Priority Remediation

The findings below are grouped by severity. Address CRITICAL findings first, then HIGH, then MEDIUM.

CRITICAL (2)

CRITICAL **K8S-SEC-001: Privileged containers detected**

Category: Workload Security

Affected resources: payments/payments-api-7d9f8c5b6-k2p9x:payments-api

Impact: The audit found resources that require review.

Recommended action: Remove privileged mode or document and isolate the exception with least-privilege controls.

CRITICAL **K8S-RBAC-004: RBAC privilege-escalation verbs granted**

Category: Identity & RBAC

Affected resources: payments-ci-escalator

Impact: The `escalate`, `bind`, or `impersonate` verbs let a subject grant itself permissions beyond what it directly holds, or act as a more privileged identity -- these are privilege-escalation primitives, not routine access.

Recommended action: Remove these verbs unless a specific, documented automation (e.g. a controller that legitimately needs to impersonate) requires them, and scope narrowly with resourceNames where possible.

HIGH (11)

HIGH **K8S-CLUSTER-003: Kubernetes version is likely past upstream support**

Category: Cluster / Control Plane

Affected resources: cluster (detected v1.27)

Impact: Detected version 1.27 is 7 minor releases behind the reference version 1.34 used by this check. Upstream Kubernetes typically supports only the latest 3 minor releases.

Recommended action: Plan an upgrade path to a supported minor version. Verify the exact support window at https://kubernetes.io/releases/ before treating this as confirmed -- this check's reference version is a hand-maintained approximation.

HIGH **K8S-RES-001: Containers missing CPU or memory requests**

Category: Workloads

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker

Impact: The audit found resources that require review.

Recommended action: Define CPU and memory requests for every container; validate with LimitRange or admission policy.

HIGH **K8S-RES-002: Containers missing CPU or memory limits**

Category: Workloads

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker

Impact: The audit found resources that require review.

Recommended action: Define appropriate CPU and memory limits and test workload behavior under load.

HIGH **K8S-SEC-002: Containers may run as root**

Category: Workload Security

Affected resources: payments/payments-api-7d9f8c5b6-k2p9x:payments-api, checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker, platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend, data/analytics-etl-9f6d8c5b4-t8u9v:analytics-etl, platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service

Impact: The audit found resources that require review.

Recommended action: Set runAsNonRoot: true and a non-zero runAsUser where the image supports it.

HIGH **K8S-WORKLOAD-002: Unhealthy container state detected**

Category: Workloads

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker (CrashLoopBackOff), data/analytics-etl-9f6d8c5b4-t8u9v:analytics-etl (ImagePullBackOff)

Impact: The audit found resources that require review.

Recommended action: Investigate image availability, application logs, configuration, and restart causes.

HIGH **K8S-WORKLOAD-003: Workloads have unavailable replicas**

Category: Workloads

Affected resources: checkout/checkout-worker, data/analytics-etl

Impact: Desired replicas are not currently available.

Recommended action: Review events, scheduling, probes, image pulls, dependencies, and rollout status.

HIGH **K8S-RBAC-001: cluster-admin bindings require review**

Category: Identity & RBAC

Affected resources: legacy-cluster-admin-binding

Impact: cluster-admin grants unrestricted cluster access.

Recommended action: Replace broad bindings with scoped roles where possible and document justified break-glass access.

HIGH

K8S-RBAC-003: cluster-admin granted through a namespaced RoleBinding

Category: Identity & RBAC

Affected resources: data/data-namespace-admin

Impact: A RoleBinding to the cluster-admin ClusterRole grants cluster-wide admin access to its subjects, not just within the binding's namespace -- this is easy to miss during a namespace-scoped review.

Recommended action: Bind to a namespace-scoped Role instead, or use a ClusterRoleBinding deliberately with documented justification.

HIGH

K8S-RBAC-005: Broad read access to Secrets granted

Category: Identity & RBAC

Affected resources: platform-wildcard-admin, secret-reader-broad

Impact: Roles that can get/list/watch Secrets (directly or via a wildcard resource) can read every credential the Secret objects contain.

Recommended action: Scope Secret access with `resourceNames` to the specific Secrets a workload needs, or use a projected/CSI-driven secret store instead of broad namespace-wide read access.

HIGH

K8S-SECRET-001: Possible credential stored in a plaintext environment variable

Category: Secrets

Affected resources: platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service:NOTIFICATIONS_API_KEY

Impact: A container defines an environment variable whose name looks credential-shaped (password/token/key/secret) using a literal `value:` instead of `valueFrom.secretKeyRef`. The value itself is never read or reported by this check -- only the variable name and location.

Recommended action: Move the value into a Secret and reference it with `valueFrom.secretKeyRef`, then remove the plaintext value from the manifest (and rotate the credential, since it may already be in version control or CI logs).

HIGH

K8S-STORAGE-001: PersistentVolumeClaims are not bound

Category: Storage

Affected resources: data/analytics-data

Impact: Unbound PVCs can leave stateful workloads pending or without durable storage.

Recommended action: Check StorageClass, provisioner health, capacity, access mode, topology, and PVC events.

MEDIUM (13)

MEDIUM **K8S-SEC-004: allowPrivilegeEscalation is not explicitly disabled**

Category: Workload Security

Affected resources: payments/payments-api-7d9f8c5b6-k2p9x:payments-api, checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker, platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service

Impact: The audit found resources that require review.

Recommended action: Set `allowPrivilegeEscalation: false` unless a container specifically requires privilege escalation.

MEDIUM **K8S-SEC-008: Containers bind a hostPort**

Category: Workload Security

Affected resources: platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend (hostPort 8080)

Impact: The audit found resources that require review.

Recommended action: Avoid hostPort; use a Service (ClusterIP/NodePort/LoadBalancer) to expose the workload instead.

MEDIUM **K8S-WORKLOAD-001: Mutable or latest image tags detected**

Category: Workloads

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker (checkout-worker:latest)

Impact: The audit found resources that require review.

Recommended action: Use immutable image digests or version tags and promote tested artifacts.

MEDIUM **K8S-HA-001: Workloads run a single replica**

Category: Resilience & Availability

Affected resources: platform/web-frontend, data/analytics-etl

Impact: A single replica is a single point of failure: any node drain, crash, or rolling update causes a full outage of the workload.

Recommended action: Run at least 2 replicas for workloads that need availability during node maintenance or rollout, combined with a PodDisruptionBudget and anti-affinity/topology spread.

MEDIUM **K8S-RBAC-002: Wildcard RBAC permissions detected**

Category: Identity & RBAC

Affected resources: platform-wildcard-admin

Impact: Wildcard permissions can grant more access than intended.

Recommended action: Replace wildcard rules with the smallest required API groups, resources, and verbs.

MEDIUM

K8S-RBAC-007: The default ServiceAccount is bound to a Role or ClusterRole

Category: Identity & RBAC

Affected resources: legacy-cluster-admin-binding, checkout/checkout-default-binding

Impact: The `default` ServiceAccount is automatically mounted into any pod that does not specify one; binding it to permissions means every such pod inherits those permissions, even unintentionally.

Recommended action: Create a dedicated ServiceAccount per workload with only the permissions it needs, and leave the default ServiceAccount unbound (and set `automountServiceAccountToken: false` on it).

MEDIUM

K8S-NET-001: Namespaces without NetworkPolicy

Category: Networking

Affected resources: data, payments, platform

Impact: Namespaces with workloads have no NetworkPolicy resource; pod traffic may be unrestricted depending on the CNI.

Recommended action: Introduce a default-deny baseline, then add explicit ingress and egress rules required by each workload.

MEDIUM

K8S-NET-003: NetworkPolicy exists but allows all traffic on one or more rules

Category: Networking

Affected resources: checkout/checkout-allow-all (ingress allows all sources)

Impact: A NetworkPolicy with an ingress/egress rule that omits `from`/`to` matches all sources/destinations, which gives a false sense of restriction while allowing everything.

Recommended action: Add explicit `from`/`to` selectors (podSelector/namespaceSelector/ipBlock) to the rule, or remove it if a default-deny is intended.

MEDIUM

K8S-NET-004: NetworkPolicy allows 0.0.0.0/0 without exceptions

Category: Networking

Affected resources: checkout/checkout-allow-all

Impact: An ipBlock of 0.0.0.0/0 with no `except` entries permits traffic from/to any IPv4 address.

Recommended action: Narrow the CIDR to the specific ranges required, or add `except` entries to carve out what must remain restricted.

MEDIUM

K8S-NET-002: Ingress resources without TLS

Category: Networking

Affected resources: platform/web-frontend

Impact: Internet-facing services may be served without TLS if the ingress controller does not enforce it elsewhere.

Recommended action: Configure TLS certificates and confirm HTTPS redirect behavior at the ingress layer.

MEDIUM **K8S-NET-006: LoadBalancer service without source IP restriction**

Category: Networking

Affected resources: checkout/checkout-worker-lb

Impact: Without `loadBalancerSourceRanges`, a cloud load balancer typically accepts traffic from anywhere on the internet.

Recommended action: Set `spec.loadBalancerSourceRanges` to the known client CIDRs, or front the service with an authenticated ingress/VPN if it does not need to be public.

MEDIUM **K8S-PSS-001: Namespaces have no Pod Security Standard enforce label**

Category: Pod Security Standards

Affected resources: payments, checkout, platform, data

Impact: Without `pod-security.kubernetes.io/enforce`, Pod Security Admission defaults to `privileged` (no restriction) for the namespace.

Recommended action: Add `pod-security.kubernetes.io/enforce: baseline` or `restricted` to the namespace, after confirming existing workloads comply (use `audit`/`warn` labels first to check without blocking).

MEDIUM **K8S-BACKUP-001: No backup automation was detected**

Category: Backup & Recovery

Affected resources: None

Impact: This discovery check cannot prove backups are absent, but no common Kubernetes backup signal was found.

Recommended action: Confirm etcd, application, database, and persistent-volume backup ownership; test a restore and document the procedure.

Detailed Findings

Every finding identified during this audit is listed below, grouped by severity, with the exact category, evidence, and recommendation captured at audit time.

CRITICAL Findings (2)

CRITICAL

K8S-SEC-001: Privileged containers detected

Category: Workload Security

Affected resources: payments/payments-api-7d9f8c5b6-k2p9x:payments-api

Finding

The audit found resources that require review.

Evidence

```
payments/payments-api-7d9f8c5b6-k2p9x:payments-api
```

Recommendation

Remove privileged mode or document and isolate the exception with least-privilege controls.

Safe remediation guidance

Set `securityContext.privileged: false` and grant only the specific `capabilities.add` the workload needs instead of full privileged mode. Requires a rollout of the affected Deployment/StatefulSet; not safe to apply automatically.

CRITICAL

K8S-RBAC-004: RBAC privilege-escalation verbs granted

Category: Identity & RBAC

Affected resources: payments-ci-escalator

Finding

The `escalate`, `bind`, or `impersonate` verbs let a subject grant itself permissions beyond what it directly holds, or act as a more privileged identity -- these are privilege-escalation primitives, not routine access.

Evidence

```
Role or ClusterRole rule includes escalate/bind/impersonate.
```

Recommendation

Remove these verbs unless a specific, documented automation (e.g. a controller that legitimately needs to impersonate) requires them, and scope narrowly with resourceNames where possible.

HIGH Findings (11)

HIGH

K8S-CLUSTER-003: Kubernetes version is likely past upstream support

Category: Cluster / Control Plane

Affected resources: cluster (detected v1.27)

Finding

Detected version 1.27 is 7 minor releases behind the reference version 1.34 used by this check. Upstream Kubernetes typically supports only the latest 3 minor releases.

Evidence

```
Detected minor version: 1.27
```

Recommendation

Plan an upgrade path to a supported minor version. Verify the exact support window at https://kubernetes.io/releases/ before treating this as confirmed -- this check's reference version is a hand-maintained approximation.

HIGH

K8S-RES-001: Containers missing CPU or memory requests

Category: Workloads

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker

Finding

The audit found resources that require review.

Evidence

```
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker
```

Recommendation

Define CPU and memory requests for every container; validate with LimitRange or admission policy.

Safe remediation guidance

```
Add `resources.requests.cpu`/`resources.requests.memory` to the container spec, or enforce a
namespace LimitRange:
  kubectl create -f - <<'EOF'
  apiVersion: v1
  kind: LimitRange
  metadata: {name: default-requests, namespace: <ns>}
  spec: {limits: [{type: Container, defaultRequest: {cpu: 100m, memory: 128Mi}}]}
  EOF
```

HIGH

K8S-RES-002: Containers missing CPU or memory limits

Category: Workloads

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker

Finding

The audit found resources that require review.

Evidence

```
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker
```

Recommendation

Define appropriate CPU and memory limits and test workload behavior under load.

Safe remediation guidance

Add `resources.limits.cpu`/`resources.limits.memory` to the container spec after load-testing the workload; do not apply blind limits to production without validation.

HIGH

K8S-SEC-002: Containers may run as root

Category: Workload Security

Affected resources: payments/payments-api-7d9f8c5b6-k2p9x:payments-api, checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker, platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend, data/analytics-etl-9f6d8c5b4-t8u9v:analytics-etl, platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service

Finding

The audit found resources that require review.

Evidence

```
payments/payments-api-7d9f8c5b6-k2p9x:payments-api
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker
platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend
data/analytics-etl-9f6d8c5b4-t8u9v:analytics-etl
platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service
```

Recommendation

Set runAsNonRoot: true and a non-zero runAsUser where the image supports it.

Safe remediation guidance

Set `securityContext: {runAsNonRoot: true, runAsUser: <uid>}` on the pod or container spec, then verify the image can run as that UID before rolling out.

HIGH

K8S-WORKLOAD-002: Unhealthy container state detected

Category: Workloads

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker (CrashLoopBackOff), data/analytics-etl-9f6d8c5b4-t8u9v:analytics-etl (ImagePullBackOff)

Finding

The audit found resources that require review.

Evidence

```
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker (CrashLoopBackOff)
data/analytics-etl-9f6d8c5b4-t8u9v:analytics-etl (ImagePullBackOff)
```

Recommendation

Investigate image availability, application logs, configuration, and restart causes.

Safe remediation guidance

Inspect the failure before restarting anything: `kubectl describe pod <ns>/<pod>` and `kubectl logs <ns>/<pod> -c <container> --previous`. Do not delete or restart the pod automatically as part of an audit.

HIGH K8S-WORKLOAD-003: Workloads have unavailable replicas

Category: Workloads

Affected resources: checkout/checkout-worker, data/analytics-etl

Finding

Desired replicas are not currently available.

Evidence

```
Available replicas are lower than desired replicas.
```

Recommendation

Review events, scheduling, probes, image pulls, dependencies, and rollout status.

HIGH K8S-RBAC-001: cluster-admin bindings require review

Category: Identity & RBAC

Affected resources: legacy-cluster-admin-binding

Finding

cluster-admin grants unrestricted cluster access.

Evidence

```
ClusterRoleBinding roleRef is cluster-admin.
```

Recommendation

Replace broad bindings with scoped roles where possible and document justified break-glass access.

HIGH K8S-RBAC-003: cluster-admin granted through a namespaced RoleBinding

Category: Identity & RBAC

Affected resources: data/data-namespace-admin

Finding

A RoleBinding to the cluster-admin ClusterRole grants cluster-wide admin access to its subjects, not just within the binding's namespace -- this is easy to miss during a namespace-scoped review.

Evidence

```
RoleBinding roleRef is cluster-admin.
```

Recommendation

Bind to a namespace-scoped Role instead, or use a ClusterRoleBinding deliberately with documented justification.

HIGH

K8S-RBAC-005: Broad read access to Secrets granted

Category: Identity & RBAC

Affected resources: platform-wildcard-admin, secret-reader-broad

Finding

Roles that can get/list/watch Secrets (directly or via a wildcard resource) can read every credential the Secret objects contain.

Evidence

```
Role or ClusterRole rule grants get/list/watch on secrets or all resources.
```

Recommendation

Scope Secret access with `resourceNames` to the specific Secrets a workload needs, or use a projected/CSI-driven secret store instead of broad namespace-wide read access.

HIGH

K8S-SECRET-001: Possible credential stored in a plaintext environment variable

Category: Secrets

Affected resources: platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service:NOTIFICATIONS_API_KEY

Finding

A container defines an environment variable whose name looks credential-shaped (password/token/key/secret) using a literal `value:` instead of `valueFrom.secretKeyRef`. The value itself is never read or reported by this check -- only the variable name and location.

Evidence

```
platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service:NOTIFICATIONS_API_KEY
```

Recommendation

Move the value into a Secret and reference it with `valueFrom.secretKeyRef`, then remove the plaintext value from the manifest (and rotate the credential, since it may already be in version control or CI logs).

HIGH

K8S-STORAGE-001: PersistentVolumeClaims are not bound

Category: Storage

Affected resources: data/analytics-data

Finding

Unbound PVCs can leave stateful workloads pending or without durable storage.

Evidence

```
PVC phase is not Bound.
```

Recommendation

Check StorageClass, provisioner health, capacity, access mode, topology, and PVC events.

Safe remediation guidance

Investigate first, do not delete: `kubectl describe pvc <ns>/<name>` and `kubectl get storageclass`. A stuck PVC is commonly a missing/misnamed StorageClass, an unavailable provisioner, or a topology mismatch between the PVC and available nodes.

MEDIUM Findings (13)

MEDIUM K8S-SEC-004: allowPrivilegeEscalation is not explicitly disabled

Category: Workload Security

Affected resources: payments/payments-api-7d9f8c5b6-k2p9x:payments-api, checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker, platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service

Finding

The audit found resources that require review.

Evidence

```
payments/payments-api-7d9f8c5b6-k2p9x:payments-api
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker
platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service
```

Recommendation

Set `allowPrivilegeEscalation: false` unless a container specifically requires privilege escalation.

Safe remediation guidance

Add `securityContext.allowPrivilegeEscalation: false` to the container spec and confirm the workload still functions.

MEDIUM K8S-SEC-008: Containers bind a hostPort

Category: Workload Security

Affected resources: platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend (hostPort 8080)

Finding

The audit found resources that require review.

Evidence

```
platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend (hostPort 8080)
```

Recommendation

Avoid hostPort; use a Service (ClusterIP/NodePort/LoadBalancer) to expose the workload instead.

Safe remediation guidance

Remove `hostPort` from the container port definition and expose the workload via a Kubernetes Service.

MEDIUM

K8S-WORKLOAD-001: Mutable or latest image tags detected

Category: Workloads

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker (checkout-worker:latest)

Finding

The audit found resources that require review.

Evidence

```
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker (checkout-worker:latest)
```

Recommendation

Use immutable image digests or version tags and promote tested artifacts.

Safe remediation guidance

Pin the image reference to a digest, e.g. `image: registry/app@sha256:<digest>`, and update the CI pipeline to publish and reference digests instead of `:latest`.

MEDIUM

K8S-HA-001: Workloads run a single replica

Category: Resilience & Availability

Affected resources: platform/web-frontend, data/analytics-etl

Finding

A single replica is a single point of failure: any node drain, crash, or rolling update causes a full outage of the workload.

Evidence

```
spec.replicas == 1.
```

Recommendation

Run at least 2 replicas for workloads that need availability during node maintenance or rollout, combined with a PodDisruptionBudget and anti-affinity/topology spread.

MEDIUM

K8S-RBAC-002: Wildcard RBAC permissions detected

Category: Identity & RBAC

Affected resources: platform-wildcard-admin

Finding

Wildcard permissions can grant more access than intended.

Evidence

```
Role or ClusterRole contains wildcard verbs or resources.
```

Recommendation

Replace wildcard rules with the smallest required API groups, resources, and verbs.

MEDIUM

K8S-RBAC-007: The default ServiceAccount is bound to a Role or ClusterRole

Category: Identity & RBAC

Affected resources: legacy-cluster-admin-binding, checkout/checkout-default-binding

Finding

The `default` ServiceAccount is automatically mounted into any pod that does not specify one; binding it to permissions means every such pod inherits those permissions, even unintentionally.

Evidence

```
A RoleBinding or ClusterRoleBinding subject is ServiceAccount 'default'.
```

Recommendation

Create a dedicated ServiceAccount per workload with only the permissions it needs, and leave the default ServiceAccount unbound (and set `automountServiceAccountToken: false` on it).

MEDIUM

K8S-NET-001: Namespaces without NetworkPolicy

Category: Networking

Affected resources: data, payments, platform

Finding

Namespaces with workloads have no NetworkPolicy resource; pod traffic may be unrestricted depending on the CNI.

Evidence

```
No NetworkPolicy objects found in these workload namespaces.
```

Recommendation

Introduce a default-deny baseline, then add explicit ingress and egress rules required by each workload.

Safe remediation guidance

```
Apply a default-deny baseline per namespace before adding allow rules:
  kubectl apply -n <ns> -f - <<'EOF'
  apiVersion: networking.k8s.io/v1
  kind: NetworkPolicy
  metadata: {name: default-deny}
  spec: {podSelector: {}, policyTypes: [Ingress, Egress]}
  EOF
Test connectivity before rolling out namespace-wide to avoid breaking required traffic.
```

MEDIUM

K8S-NET-003: NetworkPolicy exists but allows all traffic on one or more rules

Category: Networking

Affected resources: checkout/checkout-allow-all (ingress allows all sources)

Finding

A NetworkPolicy with an ingress/egress rule that omits `from`/`to` matches all sources/destinations, which gives a false sense of restriction while allowing everything.

Evidence

```
checkout/checkout-allow-all (ingress allows all sources)
```

Recommendation

Add explicit `from`/`to` selectors (podSelector/namespaceSelector/ipBlock) to the rule, or remove it if a default-deny is intended.

MEDIUM K8S-NET-004: NetworkPolicy allows 0.0.0.0/0 without exceptions

Category: Networking

Affected resources: checkout/checkout-allow-all

Finding

An ipBlock of 0.0.0.0/0 with no `except` entries permits traffic from/to any IPv4 address.

Evidence

```
checkout/checkout-allow-all
```

Recommendation

Narrow the CIDR to the specific ranges required, or add `except` entries to carve out what must remain restricted.

MEDIUM K8S-NET-002: Ingress resources without TLS

Category: Networking

Affected resources: platform/web-frontend

Finding

Internet-facing services may be served without TLS if the ingress controller does not enforce it elsewhere.

Evidence

```
Ingress spec has no tls section.
```

Recommendation

Configure TLS certificates and confirm HTTPS redirect behavior at the ingress layer.

Safe remediation guidance

Issue a certificate with cert-manager and reference it in the ingress `spec.tls`, e.g. a `Certificate` resource plus `tls: [{hosts: [<host>], secretName: <name>-tls}]`. Requires cert-manager and an Issuer/ClusterIssuer already deployed (see `automation/gitops` and docs/kubernetes-setup.md).

MEDIUM K8S-NET-006: LoadBalancer service without source IP restriction

Category: Networking

Affected resources: checkout/checkout-worker-lb

Finding

Without `loadBalancerSourceRanges`, a cloud load balancer typically accepts traffic from anywhere on the internet.

Evidence

```
spec.type == LoadBalancer with no loadBalancerSourceRanges set.
```

Recommendation

Set `spec.loadBalancerSourceRanges` to the known client CIDRs, or front the service with an authenticated ingress/VPN if it does not need to be public.

MEDIUM

K8S-PSS-001: Namespaces have no Pod Security Standard enforce label

Category: Pod Security Standards

Affected resources: payments, checkout, platform, data

Finding

Without `pod-security.kubernetes.io/enforce`, Pod Security Admission defaults to `privileged` (no restriction) for the namespace.

Evidence

```
Namespace metadata.labels has no pod-security.kubernetes.io/enforce key.
```

Recommendation

Add `pod-security.kubernetes.io/enforce: baseline` or `restricted` to the namespace, after confirming existing workloads comply (use `audit`/`warn` labels first to check without blocking).

MEDIUM

K8S-BACKUP-001: No backup automation was detected

Category: Backup & Recovery

Affected resources: None

Finding

This discovery check cannot prove backups are absent, but no common Kubernetes backup signal was found.

Evidence

```
No Velero namespace and no CronJob with backup in its name were detected.
```

Recommendation

Confirm etcd, application, database, and persistent-volume backup ownership; test a restore and document the procedure.

LOW Findings (10)

LOW

K8S-SEC-006: Containers do not drop all Linux capabilities

Category: Workload Security

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker, platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service

Finding

The audit found resources that require review.

Evidence

```
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker
platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service
```

Recommendation

Set `capabilities.drop: [ALL]` and add back only what is required.

Safe remediation guidance

Add `securityContext.capabilities.drop: [ALL]` to the container spec, then add back only the specific capabilities the workload needs.

LOW K8S-SEC-007: Container root filesystem is writable

Category: Workload Security

Affected resources: payments/payments-api-7d9f8c5b6-k2p9x:payments-api, checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker, platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service

Finding

The audit found resources that require review.

Evidence

```
payments/payments-api-7d9f8c5b6-k2p9x:payments-api
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker
platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service
```

Recommendation

Set `readOnlyRootFilesystem: true` and mount writable emptyDir volumes only where the application needs to write.

Safe remediation guidance

Add `securityContext.readOnlyRootFilesystem: true`; add an `emptyDir` volume mount for any path the application must write to (e.g. `/tmp`).

LOW K8S-SEC-009: No seccomp profile configured

Category: Workload Security

Affected resources: payments/payments-api-7d9f8c5b6-k2p9x:payments-api, checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker, platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend, data/analytics-etl-9f6d8c5b4-t8u9v:analytics-etl, platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service

Finding

The audit found resources that require review.

Evidence

```
payments/payments-api-7d9f8c5b6-k2p9x:payments-api
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker
platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend
```

```
data/analytics-etl-9f6d8c5b4-t8u9v:analytics-etl
platform/notifications-service-8b6c7d9e5-w1x2y:notifications-service
```

Recommendation

Set a seccompProfile at the pod or container level (RuntimeDefault at minimum).

Safe remediation guidance

Add `securityContext.seccompProfile: {type: RuntimeDefault}` at the pod level, or per-container if profiles differ.

LOW K8S-IMG-001: Images reference the default registry implicitly

Category: Supply Chain

Affected resources: checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker (checkout-worker:latest), platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend (nginx:1.25)

Finding

The audit found resources that require review.

Evidence

```
checkout/checkout-worker-6b8d4f9c7-m4n2p:checkout-worker (checkout-worker:latest)
platform/web-frontend-5c7b9d8f6-q7r8s:web-frontend (nginx:1.25)
```

Recommendation

Confirm images sourced from the default registry (Docker Hub) are from a vetted publisher, or mirror them into an approved registry.

Safe remediation guidance

Reference an explicit, approved registry host in the image string (e.g. `ghcr.io/org/app:tag`) and confirm pull-through or mirroring policy for third-party images.

LOW K8S-HA-002: Multi-replica workloads have no PodDisruptionBudget in their namespace

Category: Resilience & Availability

Affected resources: payments/payments-api, checkout/checkout-worker, platform/notifications-service

Finding

Without a PodDisruptionBudget, voluntary disruptions (node drains, cluster upgrades) can evict all replicas of a workload at once.

Evidence

```
No PodDisruptionBudget object exists in the workload's namespace.
```

Recommendation

Add a PodDisruptionBudget with `minAvailable` or `maxUnavailable` matching the workload's availability requirement.

LOW

K8S-RBAC-008: Possibly unused ServiceAccounts

Category: Identity & RBAC

Affected resources: payments/payments-ci, checkout/legacy-deploy-bot

Finding

No Pod currently references these ServiceAccounts. This is a discovery signal only: other controllers, CronJobs that are not currently running, or external tooling may still use them.

Evidence

```
No Pod's spec.serviceAccountName matches this ServiceAccount.
```

Recommendation

Confirm each ServiceAccount is still required before removing it or its RBAC bindings; stale service accounts with standing permissions are unnecessary attack surface.

LOW

K8S-NET-005: NodePort services detected

Category: Networking

Affected resources: platform/web-frontend-nodeport

Finding

NodePort services expose a port on every node's IP, which is often broader network exposure than intended.

Evidence

```
spec.type == NodePort.
```

Recommendation

Confirm the exposure is intentional; prefer a LoadBalancer with source-range restrictions or an Ingress with TLS for anything meant to be reachable from outside the cluster.

LOW

K8S-GOV-001: Namespaces without a ResourceQuota

Category: Governance

Affected resources: checkout, data, payments, platform

Finding

Without a ResourceQuota, a namespace has no cap on aggregate compute/object consumption, which can let one tenant exhaust cluster capacity.

Evidence

```
No ResourceQuota object found in these namespaces.
```

Recommendation

Define a ResourceQuota per namespace sized to the tenant's expected usage.

LOW

K8S-GOV-002: Namespaces without a LimitRange

Category: Governance

Affected resources: checkout, data, payments, platform

Finding

Without a LimitRange, containers can be created with no resource requests/limits at all, which this audit already flags per-container (K8S-RES-001/002); a namespace default closes the gap for future workloads too.

Evidence

```
No LimitRange object found in these namespaces.
```

Recommendation

Define a LimitRange with sane default request/limit values for the namespace.

LOW

K8S-SECRET-002: Legacy long-lived ServiceAccount token Secrets present

Category: Secrets

Affected resources: checkout/checkout-legacy-sa-token

Finding

Kubernetes 1.24+ no longer auto-creates these for every ServiceAccount; long-lived, unrotated tokens are more valuable to an attacker than the short-lived, audience-bound tokens the TokenRequest API issues.

Evidence

```
1 Secret(s) of type kubernetes.io/service-account-token found.
```

Recommendation

Confirm whether each is still required; migrate workloads to projected, auto-rotated ServiceAccount tokens where possible and delete unused legacy token Secrets.

INFO Findings (6)

INFO

K8S-CLUSTER-004: Control-plane internals were not verified

Category: Cluster / Control Plane

Affected resources: None

Finding

This audit runs `kubectl get <resource> -o json` against the API server with the credentials it was given. It cannot observe API server authentication/authorization mode, anonymous-access configuration, admission controller enablement, audit log configuration/retention/integrity, etcd authentication/encryption/exposure/backups, kube-scheduler/kube-controller-manager flags, or kubelet TLS/authentication/authorization/insecure-port configuration -- none of these are exposed by the Kubernetes API itself.

Evidence

```
No API call can answer these from outside the control plane.
```

Recommendation

Verify these directly: for a managed cluster, through the cloud provider's console/API (e.g. EKS/GKE/AKS control-plane configuration and logging settings); for self-managed control planes, by inspecting the kube-apiserver/kube-controller-manager/kube-scheduler/etcd flags and kubelet configuration on the relevant hosts.

INFO

K8S-HA-003: No HorizontalPodAutoscaler resources were found

Category: Resilience & Availability

Affected resources: None

Finding

This is a discovery signal, not proof that autoscaling is unnecessary or absent at the infrastructure level.

Evidence

```
No HorizontalPodAutoscaler objects were returned by the read-only discovery check.
```

Recommendation

Confirm whether workloads with variable load have an autoscaling strategy (HPA, KEDA, or a documented fixed-capacity decision).

INFO

K8S-NET-007: ExternalName services detected

Category: Networking

Affected resources: platform/legacy-billing-external

Finding

ExternalName services are a DNS CNAME alias; if the target hostname is ever taken over externally, in-cluster traffic silently follows it (a DNS rebinding / dependency-confusion risk).

Evidence

```
spec.type == ExternalName.
```

Recommendation

Confirm the external hostname is trusted and controlled, and that losing control of it is an acceptable/monitored risk.

INFO

K8S-CLOUD-001-GCP: GCP-managed nodes detected -- provider controls need separate review

Category: Cloud Provider

Affected resources: None

Finding

This audit only inspects Kubernetes API objects; it cannot verify cloud IAM, KMS, network, or logging configuration.

Evidence

```
Node spec.providerID matches the gcp prefix.
```

Recommendation

Manually verify: Workload Identity instead of node service-account permissions, GKE private cluster/private endpoint configuration, Cloud KMS envelope encryption for Secrets (application-layer encryption), VPC firewall rules, and Cloud Logging/Monitoring plus Security Command Center coverage.

INFO

K8S-OBS-001: Prometheus was not detected

Category: Observability

Affected resources: None

Finding

This is a discovery signal, not proof that the capability is absent.

Evidence

```
No matching deployment name was found by the read-only discovery check.
```

Recommendation

Confirm the observability platform, metrics coverage, alert routing, and runbooks with the platform owner.

INFO

K8S-OBS-002: Grafana was not detected

Category: Observability

Affected resources: None

Finding

This is a discovery signal, not proof that the capability is absent.

Evidence

```
No matching deployment name was found by the read-only discovery check.
```

Recommendation

Confirm the observability platform, metrics coverage, alert routing, and runbooks with the platform owner.