



Part 20: Kubernetes Real-Time Troubleshooting

Introduction

Welcome to the world of Kubernetes troubleshooting, where every challenge is an opportunity to sharpen your skills and emerge victorious. Join us as we embark on a journey through common real-time scenarios, unraveling mysteries, and uncovering solutions along the way.



PART 20 - KUBERNETES REAL-TIME TROUBLESHOOTING

- Controller Manager Leader Election Failures
- Endpoint Controller Errors
- Secret Management Errors
- Network Policy Enforcement Issues
- API Server Connection Errors
- 2 Bonus Issues of HPA & Latency

Scenario 96: Controller Manager Leader Election Failures

rabbitmq/cluster-operator

#69 Controller manager crashes with leader election failure



 0 comments



chewymeister opened on March 31, 2020





Symptoms: Controller manager leader election fails to elect a leader, resulting in controller manager downtime.

Diagnosis: Review controller manager logs (`kubectl logs -n kube-system <controller_manager_pod_name>`) and inspect leader election status.

Solution:

1. Verify that leader election configurations are correctly set up and that etcd is accessible from all controller manager nodes.
2. Implement leader election health checks and automated failover mechanisms to recover from leader election failures.
3. Monitor leader election metrics and configure alerts for abnormal leader election behaviour.

Scenario 97: Endpoint Controller Errors

kubernetes/kubernetes

#56972 Controller-manager unable to update service...



13 comments



andor44 opened on December 8, 2017



hashicorp/consul-k8s



#2491 BUG+FIX: Endpoints controller fails to deregister...

5 comments



mr-miles opened on June 29, 2023





Symptoms: Endpoints for Kubernetes services are not being updated or synchronized correctly.

Diagnosis: Check endpoint controller logs (`kubectl logs -n kube-system <endpoint_controller_pod_name>`) and review endpoint controller configurations.

Solution:

1. Ensure that the endpoint controller is running and healthy in the cluster.
2. Check for errors or misconfigurations in endpoint controller configurations that may prevent endpoint updates.
3. Monitor endpoint controller metrics and implement retry mechanisms for endpoint synchronization to handle transient failures.

Scenario 98: Secret Management Errors

Azure/secrets-store-csi-driver-provider-azure

#90 Sync with Kubernetes Secrets not working



6 comments

 bq1756 opened on May 7, 2020



Azure/secrets-store-csi-driver-provider-azure

#224 Kubernetes secret not updated even after pod restart



4 comments

 kamilzzz opened on September 6, 2020





Symptoms: Pods fail to start due to errors accessing or decrypting secrets.

Diagnosis: Describe pod (`kubectl describe pod <pod_name>`) and review secret configurations.

Solution:

1. Verify that secrets are correctly created and referenced in pod specifications with the appropriate volume mounts or environment variables.
2. Check for permission issues or incorrect secret data that may cause decryption errors.
3. Monitor secret rotation and expiration policies to prevent disruptions to applications relying on secrets.

Scenario 99: Network Policy Enforcement Issues

k3s-io/k3s

#947 Network policy enforcement is delayed



11 comments



wilsonianb opened on October 23, 2019



aws/aws-network-policy-agent

#271 Network Policy Not Enforced on Initial Creation



9 comments



kervrosales opened on May 17, 2024





Symptoms: Network policies are not applied correctly, leading to unauthorized network access or traffic disruptions.

Diagnosis: Describe network policies (`kubectl describe networkpolicy <policy_name>`) and review network plugin configurations.

Solution:

1. Verify that the network plugin (e.g., Calico, Cilium) is properly installed and configured in the cluster.
2. Check for conflicts or overlapping rules in network policies that may be causing unintended behaviour.
3. Monitor network traffic and implement logging and auditing to detect and troubleshoot policy enforcement issues.

Scenario 100: API Server Connection Errors

kubernetes/kubernetes

#94063 Kubernetes API server response timeout



8 comments



csandiri11 opened on August 17, 2020



kubernetes/kubernetes

#45787 Connection refused for apiserver



7 comments



yasassri opened on May 14, 2017





Symptoms: Clients experience intermittent connection errors or timeouts when interacting with the Kubernetes API server.

Diagnosis: Review API server logs (`kubectl logs -n kube-system kube-apiserver`) and inspect network connectivity.

Solution:

1. Check for network issues such as packet loss, latency, or firewall rules blocking API server traffic.
2. Monitor API server performance metrics and scale the API server deployment to handle increased load.
3. Implement API request retries and circuit breakers in client applications to handle transient errors more gracefully.

Scenario 101: Horizontal Pod Autoscaler (HPA) Misbehavior

kubernetes/kubernetes

#54578 HPA failed with error "horizontal-pod-autoscaler failed to get..."



4 comments

 **kishorekumark** opened on October 25, 2017



kubernetes/kubernetes

#95237 HPA won't work: FailedGetResourceMetric confused by having tw...



10 comments

 **mpkonmbk** opened on October 1, 2020



Symptoms: HPA does not scale pods in or out appropriately based on CPU or custom metrics utilization.



Diagnosis: Describe HPAs (`kubectl describe hpa <hpa_name>`) and review metric server configurations.

Solution:

1. Verify that the metrics server is deployed and functioning correctly in the cluster to provide accurate metrics for HPA scaling decisions.
2. Adjust HPA configurations, including target CPU utilization thresholds and scaling behavior parameters, to better align with application requirements.
3. Monitor pod and cluster resource utilization over time to fine-tune HPA settings and optimize workload scalability.

Scenario 102: API Server Latency Spikes

kubernetes/kubernetes

#59049 High API Server latency



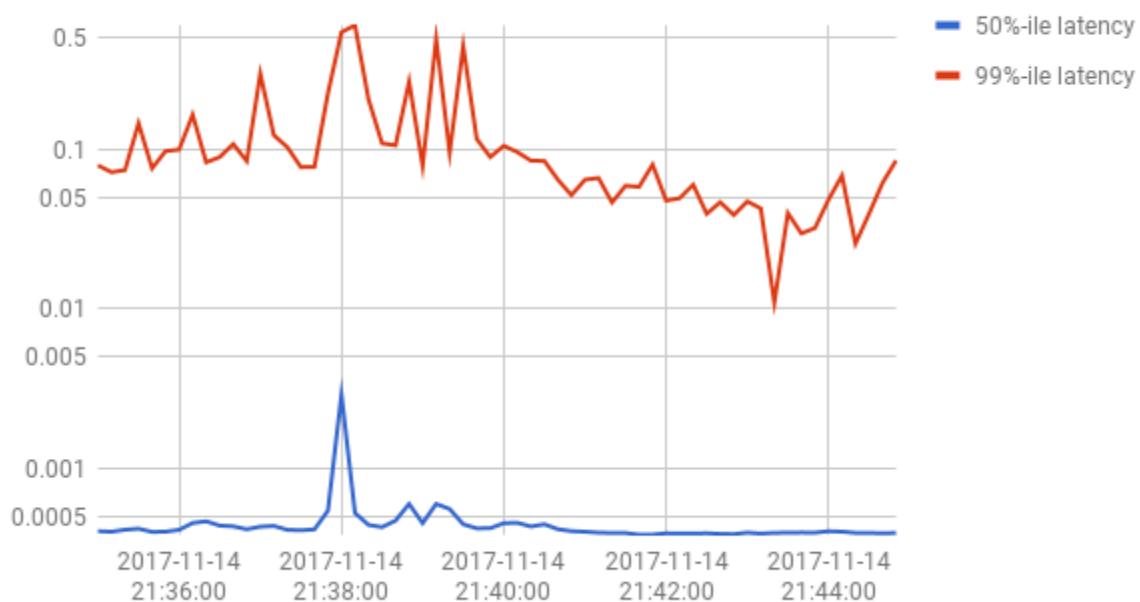
8 comments



naveensrinivasan opened on January 30, 2018



GET 50%-ile latency and 99%-ile latency





Symptoms: API server response times increase significantly, leading to delays in pod scheduling, API resource creation, or cluster management operations.

Diagnosis: Monitor API server metrics (e.g., latency, throughput) using Prometheus or Kubernetes dashboard and inspect API server logs for any errors or warnings.

Solution:

1. Investigate potential causes of API server latency spikes, such as high request volumes, resource contention, or network congestion.
 2. Scale API server replicas horizontally or vertically to handle increased workload or resource demands during peak usage periods.
 3. Optimize API server performance by tuning configuration settings, implementing caching mechanisms, or offloading heavy processing tasks to dedicated components.
-



In the up-coming parts, we will discuss on more troubleshooting steps for the different Kubernetes based scenarios. So, stay tuned for the and follow @Prasad Suman Mohan for more such posts.

