**EKS Monitoring with New Relic - Review and Insights**

**Objective**

To evaluate the comprehensiveness of EKS monitoring using New Relic, validate key metrics, identify gaps in event and alert coverage, and propose actionable recommendations to improve observability.

**Metrics Monitored**

Highlight key metrics New Relic monitors for EKS:

**Cluster-Level Metrics**

* Total CPU/Memory utilization (Request/Limit).
* Node availability and health (Ready/NotReady states).
* Network I/O (Bytes In/Out per second).

**Pod-Level Metrics**

* Pod status (Running, Pending, Failed).
* Container restarts.
* CPU throttling percentage.

**Control Plane Monitoring**

The Kubernetes API Server and Control Plane component details are not accessible due to the EKS-managed Kubernetes cluster. Please refer to the New Relic documentation for more information. [Configure control plane monitoring | New Relic Documentation](https://docs.newrelic.com/docs/kubernetes-pixie/kubernetes-integration/advanced-configuration/configure-control-plane-monitoring/)

**Event Coverage**

The following event types were reviewed against the EKS cluster (ukrs-landp-eks-dev-001):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | **Event Type** |  |  | | --- | |  | | **Availability** | | **Observations** | | --- |  |  | | --- | |  | |
| |  | | --- | | **K8sClusterSample** |  |  | | --- | |  | | Available | Provides high-level metrics such as cluster health, node counts, and resource utilization. |
| |  | | --- | | **K8sContainerSample** |  |  | | --- | |  | | Available | |  | | --- | | Captures container-specific metrics, including CPU, memory usage, and restart counts. |  |  | | --- | |  | |
| **K8sDaemonsetSample** | Available | |  | | --- | | Monitors DaemonSet status, deployment status, and pod counts. |  |  | | --- | |  | |
| **K8sDeploymentSample** | Available | |  | | --- | | Tracks Deployment health, replica status, and update progress. |  |  | | --- | |  | |
| |  | | --- | | **K8sEndpointSample** |  |  | | --- | |  | | Available | |  | | --- | | Captures Kubernetes endpoints, useful for service discovery and troubleshooting. |  |  | | --- | |  | |
| **K8sNamespaceSample** | Available | |  | | --- | | Provides namespace-level metrics, including resource usage and event counts. | |  |  |  | | --- | |  | |
| **K8sNodeSample** | Available | |  | | --- | | Monitors node-level metrics such as readiness, resource utilization, and status events. |  |  | | --- | |  | |
| **K8sPodSample** | Available | |  | | --- | | Tracks pod-level metrics like status (Running, Pending, Failed) and resource consumption. |  |  | | --- | |  | |
| **K8sReplicasetSample** | Available | |  | | --- | | Monitors ReplicaSet health, replica counts, and deployment issues. |  |  | | --- | |  | |
| **K8sServiceSample** | Available | Provides data on Kubernetes services, such as endpoint configurations and service health. |
| **K8sVolumeSample** | Available | |  | | --- | | Captures persistent volume metrics, including usage and binding status. |  |  | | --- | |  | |

**Missing Events:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | **Event Type** |  |  | | --- | |  | | **Availability** | | **Observations** | | --- |  |  | | --- | |  | |
| |  | | --- | | **K8sApiServerSample** |  |  | | --- | |  | | Missing | |  | | --- | | Critical for monitoring API server latency, error rates, and certificate expiration. |  |  | | --- | |  | |
| |  | | --- | | **K8sCronJobSample** |  |  | | --- | |  | | Missing | |  |  |  | | --- | --- | --- | | |  | | --- | | Metrics for scheduled jobs are absent, limiting visibility into CronJob success and failure rates. |  |  | | --- | |  | |  |  | | --- | |  | |
| **K8sEtcdSample** | Missing | |  |  |  | | --- | --- | --- | | |  | | --- | | Required for etcd health and consistency monitoring; essential for control plane operations. |  |  | | --- | |  | |  |  | | --- | |  | |
| **K8sHpaSample** | Missing | |  |  |  | | --- | --- | --- | | |  | | --- | | Horizontal Pod Autoscaler (HPA) metrics such as scaling activity and metric availability are missing. |  |  | | --- | |  | |  |  | | --- | |  | |
| |  |  | | --- | --- | | |  | | --- | |  | |  |  |  | | --- | --- | | |  | | --- | | **K8sJobSample** | | | Missing | |  | | --- | | Tracks Job execution metrics; absence limits visibility into batch job performance. |  |  | | --- | |  | |
| |  | | --- | | **K8sSchedulerSample** |  |  | | --- | |  | | Missing | |  | | --- | | Metrics for pod scheduling delays and resource allocation are unavailable. |  |  | | --- | |  | |
| **K8sStatefulsetSample** | Missing | StatefulSet-specific metrics like pod readiness and update progress are missing. |

**Recommendations for Missing Events:**

* 1. **Workload-Specific Events**:
* **K8sCronJobSample**, **K8sHpaSample**, **K8sJobSample**, **K8sStatefulsetSample**:  
  These events are not shown in New Relic because the EKS cluster does not have corresponding workloads or configurations deployed.
  + **Recommendation**: Deploy workloads like CronJobs, Horizontal Pod Autoscalers (HPA), StatefulSets, and Jobs in the EKS cluster if these metrics are critical for your use case. Ensure New Relic monitoring agents are correctly configured to capture them after deployment.
  1. **Control Plane Metrics**:
* **K8sApiServerSample**, **K8sEtcdSample**, **K8sSchedulerSample**:  
  These details are unavailable because Amazon EKS is a managed Kubernetes service, and its control plane is abstracted and managed by AWS. [Configure control plane monitoring | New Relic Documentation](https://docs.newrelic.com/docs/kubernetes-pixie/kubernetes-integration/advanced-configuration/configure-control-plane-monitoring/)
  + **Recommendation**:
    - Use **CloudWatch** for control plane monitoring in EKS, as AWS provides metrics for API server latency, etcd health, and other control plane events.

**Reviewed Queries and Their Alerts**

It evaluates the current coverage of metrics and queries used to monitor various Kubernetes components and provides recommendations for improvement. The review identifies gaps in monitoring coverage and suggests actionable steps to enhance observability.

**Scope**

* Evaluates the effectiveness of New Relic in monitoring the EKS cluster.
* Lists available and unavailable queries for key Kubernetes components.
* Highlights missing events and metrics due to the nature of EKS being a managed Kubernetes service.
* Recommends improvements to address identified gaps.

|  |  |  |  |
| --- | --- | --- | --- |
| **SNO** | **Custom Events or Alerts** | **New Relic Query** | **Status** |
| 1 | |  | | --- | | **Container OOM Killed** |  |  | | --- | |  | | SELECT count(container) FROM Metric WHERE reason = 'OOMKilled' FACET container, pod | **No Event**  (No data available) |
| |  | | --- | |  |   2 | |  | | --- | | **Pod Job Failed** |  |  | | --- | |  | | **SELECT count(\*) FROM K8sPodSample WHERE status = 'Failed' FACET podName, namespace SINCE this quarter** | Working |
| 3 | **StatefulSet Down** | **N/A** | **No Event** |
| 4 | **HPA Scaling Ability** | **SELECT latest(AbleToScale) FROM K8sHpaSample WHERE AbleToScale = 'false' FACET hpaName** | **No Event**  (No data available) |
| 5 | **HPA Metric Availability** | **SELECT latest(ScalingActive) FROM K8sHpaSample WHERE ScalingActive = 'false' FACET hpaName** | **No Event**  (No data available) |
| 6 | |  | | --- | |  |  |  | | --- | | **Pod Not Ready** | | **SELECT uniqueCount(podName) AS 'Not Running Pod' FROM K8sPodSample WHERE isReady = 0** | Working |
| 7 | |  | | --- | |  |  |  | | --- | | **Pod Crash Looping** | | |  | | --- | | **SELECT latest(restartCount) FROM K8sContainerSample FACET podName, namespace** |  |  | | --- | |  | | Working |
| 8 | **Kubernetes API Errors** | |  | | --- | | **SELECT rate(count(\*), 1 minute) AS 'Error Rate' FROM K8sApiServerSample WHERE statusCode >= 400** |  |  | | --- | |  | | **No Event**  (No data available) |
| 9 | |  | | --- | | **Kubernetes API Latency** |  |  | | --- | |  | | |  | | --- | | **SELECT percentile(requestDuration, 99) AS '99th Percentile Latency' FROM K8sApiServerSample** |  |  | | --- | |  | | **No Event**  (No data available) |
| 10 | |  | | --- | | **Kubernetes Client Cert Expiring** |  |  | | --- | |  | | |  | | --- | | **SELECT latest(clientCertificateExpiration) AS 'Certificate Expiration Time' FROM K8sApiServerSample** |  |  | | --- | |  | | **No Event**  (No data available) |
| 11 | |  | | --- | | **Node Not Ready** |  |  | | --- | |  | | **SELECT uniqueCount(nodeName) AS 'Node Not Ready' FROM K8sNodeSample WHERE condition.Ready = 0** | Working |
| 12 | **Node High CPU** | |  | | --- | | **SELECT latest(allocatableCpuCoresUtilization) AS 'CPU Usage Percentage' FROM K8sNodeSample FACET nodeName** |  |  | | --- | |  | | Working |
| 13 | **Node OOM Kill Detected** | |  | | --- | | **SELECT count(\*) FROM K8sContainerSample WHERE reason = 'OOMKilled' FACET nodeName** |  |  | | --- | |  | | **No Event**  (No data available) |
| 14 | **Node Clock Not in Sync** |  |  |
| 15 | |  | | --- | |  |  |  | | --- | | **Pod High CPU** | | |  | | --- | | **SELECT latest(cpuCoresUtilization) FROM K8sContainerSample FACET podName** |  |  | | --- | |  | | Working |
| 16 | |  | | --- | |  |  |  | | --- | | **Pod High Memory** | | |  | | --- | | **SELECT latest(requestedMemoryUtilization) FROM K8sContainerSample FACET podName** |  |  | | --- | |  | | Working |
| 17 | **CoreDNS Down** | |  | | --- | | **SELECT count(pod) FROM Metric WHERE metricName = 'kube\_pod\_info' AND pod LIKE '%coredns%' FACET pod** |  |  | | --- | |  | | Working |
| 18 | **CoreDNS Errors** | |  | | --- | | **SELECT percentage(count(\*), WHERE dnsError = 'SERVFAIL') AS 'SERVFAIL Error Percentage' FROM K8sContainerSample WHERE containerName = 'coredns'** |  |  | | --- | |  | | Working |
| 19 | **CoreDNS Latency** | |  | | --- | | **SELECT percentile(requestDuration, 99) AS '99th Percentile Request Duration' FROM K8sContainerSample WHERE containerName = 'coredns'** |  |  | | --- | |  | | Working |
| 20 | **CoreDNS Forwarding Latency** | |  | | --- | | **SELECT percentile(forwardRequestDuration, 99) AS '99th Percentile Forward Request Duration' FROM K8sContainerSample WHERE containerName = 'coredns' AND forwardTarget = 'kube-dns'** |  |  | | --- | |  | | Working |
| 21 | **CoreDNS Forwarding Error** | |  | | --- | | **SELECT percentage(count(\*), WHERE forwardError = 'true') AS 'Forwarding Error Rate' FROM K8sContainerSample WHERE containerName = 'coredns'** |  |  | | --- | |  | | Working |

**Dashboard Enhancement**

**Kubernetes Events Page - Cluster Events Widget**

* **Issue:** The query in the Cluster Events widget is incorrect due to a misconfiguration of the cluster name.
* Resolution: Update the query to use the correct cluster name (ukrs-landp-eks-dev-001) to accurately reflect the cluster-specific events.

**Suggested Dashboard Name**

* Rename the dashboard to "**EKS Cluster Monitoring - ukrs-landp-eks-dev-001**" for clarity and better alignment with its purpose.

**Remaining Dashboard Enhancements**

* **Recommendation:** Assign the remaining enhancements to **Kanika** for implementation. This includes refining other widgets, updating queries, and ensuring all key metrics and events are captured effectively.