Voice Microservices Private Edition Guide

Config Service metrics and alerts

5/6/2026

Contents

Find the metrics Config Service exposes and the alerts defined for Config Service.

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Config Service	Supports both CRD and annotations	9100	http://:9100/metrics	30 seconds

See details about:

- Config Service metrics
- Config Service alerts

Metrics

You can query Prometheus directly to see all the metrics that the Voice Config Service exposes. The following metrics are likely to be particularly useful. Genesys does not commit to maintain other currently available Config Service metrics not documented on this page.

Metric and description	Metric details	Indicator of
config_device_response Number of device responses for each request.	**Unit:** N/A **Type:** counter **Label:** location, tenant, request_type, status **Sample value:** 2	Traffic
config_tenant_response Number of Tenant responses for each request.	**Unit:** N/A **Type:** counter **Label:** location, request_type, status **Sample value:** 2	Traffic
config_node_get_response Number of Get responses for each request.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
config_node_agent_response Number of agent responses for each request.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
config_redis_state	**Unit:** N/A	Errors

Metric and description	Metric details	Indicator of
Current Redis connection state: -1 – error 0 – disconnected 1 – connected 2 – ready	**Type:** gauge **Label:** location, redis_cluster_name **Sample value:** 2	
service_version_info Displays the version of Voice Config Service that is currently running. In the case of this metric, the labels provide the important information. The metric value is always 1 and does not provide any information.	**Unit:** N/A **Type:** gauge **Label:** version **Sample value:** service_version_info{version="100.0.1000006"} 1	
config_health_level Health level of the config node: -1 – error 0 – fail 1 – degraded 2 – pass	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 2	Errors
config_healthcheck_generic_exception Generic error during health check.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 0	

Alerts

The following alerts are defined for Config Service.

Alert	Severity	Description	Based on	Threshold
Redis disconnected for 5 minutes	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, then restart Redis. • If the alarm is triggered only for the pod {{ $labels.pod }}, check to see if there is an issue with the	redis_state	Redis is not available for pod {{ $labels.pod }} for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		pod.		
Redis disconnected for 10 minutes	Critical	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, then restart Redis. • If the alarm is triggered only for the pod {{ $labels.pod }}, check to see if there is an issue with the pod.	redis_state	Redis is not available for the pod {{ $labels.pod }} for 10 minutes.
Pod Failed	Warning	Actions: • One of the containers in the pod has entered a failed state. Check the Kibana logs for the reason.	kube_pod_status_phase	Pod failed {{ $labels.pod }}.
Pod Unknown state	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with the Kubernetes cluster. • If the alarm is triggered only for the pod {{ $labels.pod }}, check to see whether the image is correct and if the container is starting up.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Unknown state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
Pod Pending state	Warning	Actions: • If the alarm is triggered for multiple services, make sure the Kubernetes nodes where the pod is running are alive in the cluster. • If the alarm is triggered only for the pod {{ $labels.pod }}, check the health of the pod.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Pending state for 5 minutes.
Pod Not ready for 10 minutes	Critical	Actions: • If this alarm is triggered, check whether the CPU is available for the pods. • Check whether the port of the pod is running and serving the request.	kube_pod_status_ready	Pod {{ $labels.pod }} is in NotReady state for 10 minutes.
Container restarted repeatedly	Critical	Actions: • One of the containers in the pod has entered a failed state. Check the Kibana logs for the reason.	kube_pod_container_status_restarts_total	Container {{ $labels.container }} was restarted 5 or more times within 15 minutes.
Pod memory greater than 65%	Warning	High memory usage for pod {{ $labels.pod }}. Actions: • Check whether	container_memory_working_set_bytes, kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 65% for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached. • Check Grafana for abnormal load. • Collect the service logs; raise an investigation ticket.		
Pod memory greater than 80%	Critical	Critical memory usage for pod {{ $labels.pod }}. Actions: • Check whether the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached. • Check Grafana for abnormal load. • Restart the service. • Collect the service logs; raise an investigation ticket.	container_memory_working_set_bytes kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 80% for 5 minutes.
Pod CPU greater than 65%	Warning	High CPU load for pod {{ $labels.pod }}. Actions: • Check whether the horizontal	container_cpu_usage_seconds_total container_spec_cpu_period	Container {{ $labels.container }} CPU usage exceeded 65% for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		pod autoscaler has triggered and if the maximum number of pods has been reached. • Check Grafana for abnormal load. • Collect the service logs; raise an investigation ticket.		
Pod CPU greater than 80%	Critical	Critical CPU load for pod {{ $labels.pod }}. Actions: • Check whether the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached. • Check Grafana for abnormal load. • Restart the service. • Collect the service logs; raise an investigation ticket.	container_cpu_usage_seconds_total, container_spec_cpu_period	Container {{ $labels.container }} CPU usage exceeded 80% for 5 minutes.