Voice Microservices Private Edition Guide

5/26/2022

Table of Contents

Contents

Find links to all the topics in this guide.

Related documentation:

-

Voice Microservices is a service available with the Genesys Multicloud CX private edition offering. Voice Microservices includes the Tenant Service, however there is a separate Private Edition Guide for the Tenant Service. For information about the Tenant Service, including provisioning, configuration, and deployment information, see the *Tenant Service Private Edition Guide*.

Overview

Learn more about Voice Microservices and how to get started.

- About Voice Microservices
- Architecture
- High availability and disaster recovery

Configure and deploy

Find out how to configure and deploy Voice Microservices.

- Before you begin
- Configure Voice Microservices
- Provision Voice Microservices
- Deploy Voice Microservices
- Upgrade, rollback, or uninstall Voice Microservices

Observability

Learn how to monitor Voice Microservices with metrics and logging.

- Observability in Voice Microservices

- Agent State Service metrics and alerts
- Call State Service metrics and alerts
- Config Service metrics and alerts
- Dial Plan Service metrics and alerts
- FrontEnd Service metrics and alerts
- ORS metrics and alerts
- Voice Registrar Service metrics and alerts
- Voice RQ Service metrics and alerts
- Voice SIP Cluster Service metrics and alerts
- Voice SIP Proxy Service metrics and alerts
- Voicemail metrics and alerts

About Voice Microservices

Contents

Learn about Voice Microservices and how it works in Private Edition.

Related documentation:

-

Supported Kubernetes platforms

Voice Microservices are supported on the following Kubernetes platforms:

- Google Kubernetes Engine (GKE)
- OpenShift Container Platform (OpenShift)

See the Voice Microservices Release Notes for information about when support was introduced.

Voice Microservices

Voice Microservices is an application cluster that provides the following functionality:

- Handle incoming voice (SIP) interactions
- Route voice and digital (IXN) interactions
- Support outbound interactions
- Provide events stream for reporting
- Support agents across regions

Voice Microservices comprises the following microservices:

- Voice SIP Cluster Service
- Voice SIP Proxy Service
- Voice Tenant Service
- Voice Orchestration Service
- Voice Agent State Service
- Voice Call State Service
- Voice Dial Plan Service
- Voice Config Service
- Voice Registrar Service

- Voice Front End Service
- Voice Redis (RQ) Service
- Voice Voicemail Service

Voice SIP Cluster Service

The Voice SIP Cluster Service provides the following functionality:

- Handles SIP signaling by running multiple nodes: each node is tenant-independent and uses a Voice Dial Plan Service to resolve tenant-specific information.
- N+1 scalable: Each node starts from a predefined configuration file, which is the same for every node in the cloud.
- Includes a **js** controller providing traditional services to SIP Server (LCA, HA link), as well as:
  - Publishing TLib events and user data requests for Voice Call State, Voice Orchestration, and Voice Tenant Services.
  - Providing the Rest API to handle TLib requests from a Voice Front End Service.

Voice SIP Proxy Service

The Voice SIP Proxy Service is an intermediate interface among services and the Voice SIP Cluster Service. The Voice SIP Proxy Service provides the following functionality:

- Balances load of SIP signaling across Voice SIP Cluster Service instances.
- Processes SIP REGISTER requests and relays them to Voice Registrar Service.

SIP Proxy adds the following URL into the SIP messaging sent to the SBC:

```
voice-sipproxy.{{k8s-namespace }}.svc.cluster.local
```

This is an SRV record created in the K8s DNS when the SIP Proxy Service is deployed. This FQDN depends on the name of a namespace where SIP Proxy Service is deployed.

The DNS used by an SBC is integrated with the K8s DNS service to forward .svc.cluster.local FQDNs K8s DNS.

Voice Tenant Service

The Voice Tenant Service is a core service of the Genesys Multicloud CX platform that serves as an application layer between front-end Genesys Multicloud CX solutions and shared back-end core services in a region.

The Voice Tenant Service instances are dedicated to a tenant of Genesys Multicloud CX platform and

provide these main functions: provisioning of tenant resources, such as agents and DNs; routing of interactions within a tenant; execution of outbound campaigns for a tenant; providing call control functionality; participation in authentication workflow for tenant's agents.

Voice Orchestration Service

The Voice Orchestration Service provides the following functionality:

- Interacts with each Voice Tenant Service.
- Provides routing instructions to a Voice Front End Service.
- Provides local routing session states through a storage system.
- Retrieves Route Points (RP) configuration with URLs and parameters of associated Designer SCXML Application from the Voice Config Service.
- Dynamically retrieves Applications from Designer Application Server.
- Compiles Designer Application into a javascript code to be executed with each session.
- Monitors Redis streams for new interactions from SIP Cluster Service, IXN Service or GWS. Orchestration Services retrieve triggering events in a round-robin fashion, thus new interactions are evenly distributed between Orchestration Nodes.
- Starts and executes Voice and digital Sessions when triggered by routing events.
- Reads from Voice RQ Service streams TLib events and user data requests published by Voice SIP Cluster Service.
- Reads from Voice RQ Service streams Interaction (IXN) events and user data requests published by IXN Service.
- Delivers call control and user data update requests to a proper Voice SIP Cluster Service node via the Restful API.
- Delivers new call control requests to a Voice Front End Service via the Restful API.
- Sends requests to URS via a corresponding Tenant Redis stream as a session requires.
- Reads from Voice RQ Service streams URS responses and events.
- Serializes context of sessions into Redis for HA.
- Recovers sessions from Redis in case of ORS failover and continues session execution from the last state it was serialized.
- Processes HTTP requests from MCP and sends events back.
- Provides monitoring and health metrics using the Prometheus API.

Voice Agent State Service

The Voice Agent State Service provides the following functionality:

- Maintains agent states in a storage system. Recovers agent states from failure and in case of auto-

scaling events.

- Reads agent state requests (RequestAgentLogin, RequestAgentReady, ...) from a Voice Front End Service.
- Updates agent login sessions (through a Voice Config Service) based on those requests.
- Generates agent state events according to the TLib model and provides them to a Voice Tenant Service and reporting clients.
- Reads agent-related interaction events (EventRinging, ...) from a Voice Call State Service and updates agent session accordingly. Provides those events to reporting clients.
- Reads device notifications (in service/out of service) from a Voice Registrar Service and updates agent states accordingly.
- Reads agent reservation requests (RequestReserveAgent) from a Voice Front End Service and grants agent reservation to clients.

Voice Call State Service

The Voice Call State Service provides the following functionality:

- Reads interaction events from a Voice SIP Cluster Service.
- Reads user data requests from a Voice Front End Service and updates call user data states accordingly.
- Maintains call-thread states in a storage system.
- Recovers call-thread states from failure and in case of auto-scaling events.
- Produces agent-related call events to a Voice Agent State Service.

Voice Dial Plan Service

The Voice Dial Plan Service provides the following functionality:

- Provides the HTTP interface to the Voice SIP Cluster Service for device type resolution (internal, external) and dial plan execution, including the number translation.
- Supports Voicemail scenarios.
- Provides the following information to the SIP Cluster Service:
  - Device contact
  - Agent logged in on the device
  - Options configured on the DN or at Person CME object.

Voice Config Service

The Voice Config Service provides the following functionality:

- Provides access to tenant configuration data through the Rest API.
- Provides the Rest API for services to store and access device registration and agent login information.
- The following services access the configuration:
  - Voice Orchestration Service (for obtaining SCXML application details of a Route Point).
  - Voice SIP Cluster Service (for obtaining details about a tenant trunk and softswitch).
  - Voice Dial Plan Service (for obtaining details about tenants and Dial Plan provisioning).
  - Voice SIP Proxy Service (for obtaining details about tenants).
  - Voice Registrar Service (for saving details about device registration).
  - Voice Agent State Service (for saving details about agent logins).

Voice Registrar Service

The Voice Registrar Service provides the following functionality:

- Maintains device states by processing SIP REGISTER messages.
- Stores device registrations through a Voice Config Service.
- Distributes device notifications (EventDNBackInService, EventDNOutOfService) to a Voice Tenant Service. Device notifications can also be used by a Voice Agent State Service for agent state updates.

Voice Front End Service

The Voice Front End Service provides the following functionality:

- Delivers call control, user data updates, and distribute event requests to a proper Voice SIP Cluster Service node that handles the call.
- Writes agent state, agent reservation, DND status requests to a storage system (Kafka topic), consumed by a Voice Agent Service.

Voice Redis Queue Service

The Voice Redis Queue (RQ) Service provides the following functionality:

- Distributes TLib events for each voice call or digital interaction to a Voice Orchestration Service from other services, such as a Voice SIP Cluster Service and Interaction Service.
- The Voice RQ Service works as a cluster of nodes, where each node in the cluster accepts client connections and plays primary and backup roles.

- To interact with the Voice RQ Service, the rq-client library is used by other services that take care of computing the RQ node, to which TLib events are sent.

Voice Voicemail Service

The Voice Voicemail Service is part of the multi-tenant microservice architecture. It provides the following functionality:

- Provides deposit of voicemail messages to agent and agent group mailboxes.
- Provides access to voice mailboxes by dialing to a voicemail access number.
- Uses the Voice Config Service to retrieve agent configuration and states.
- Stores voicemail recordings and metadata in a storage system.
- Provisioning is done through Agent Setup.

Architecture

Contents

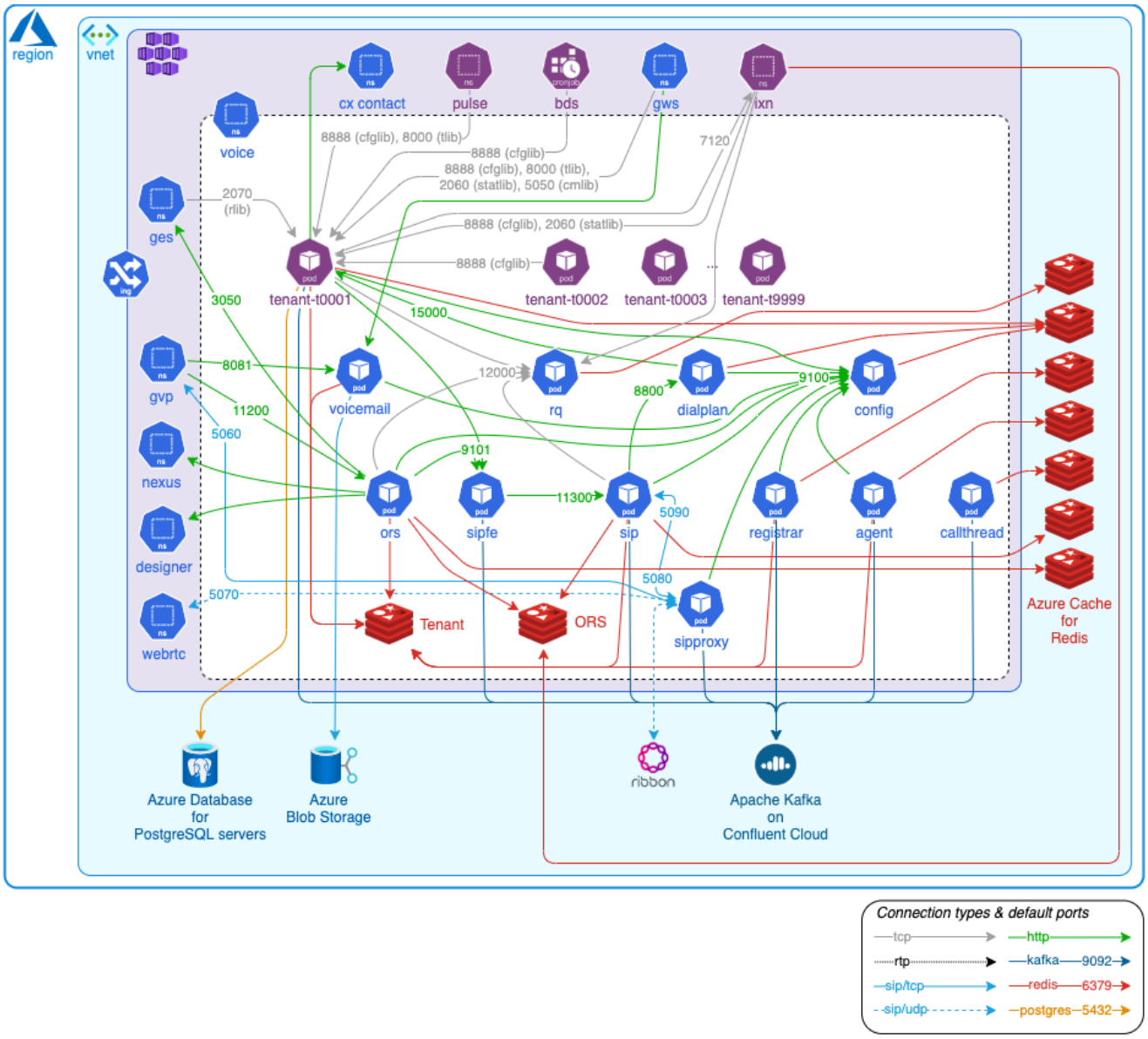
Learn about Voice Microservices architecture.

Related documentation:

-

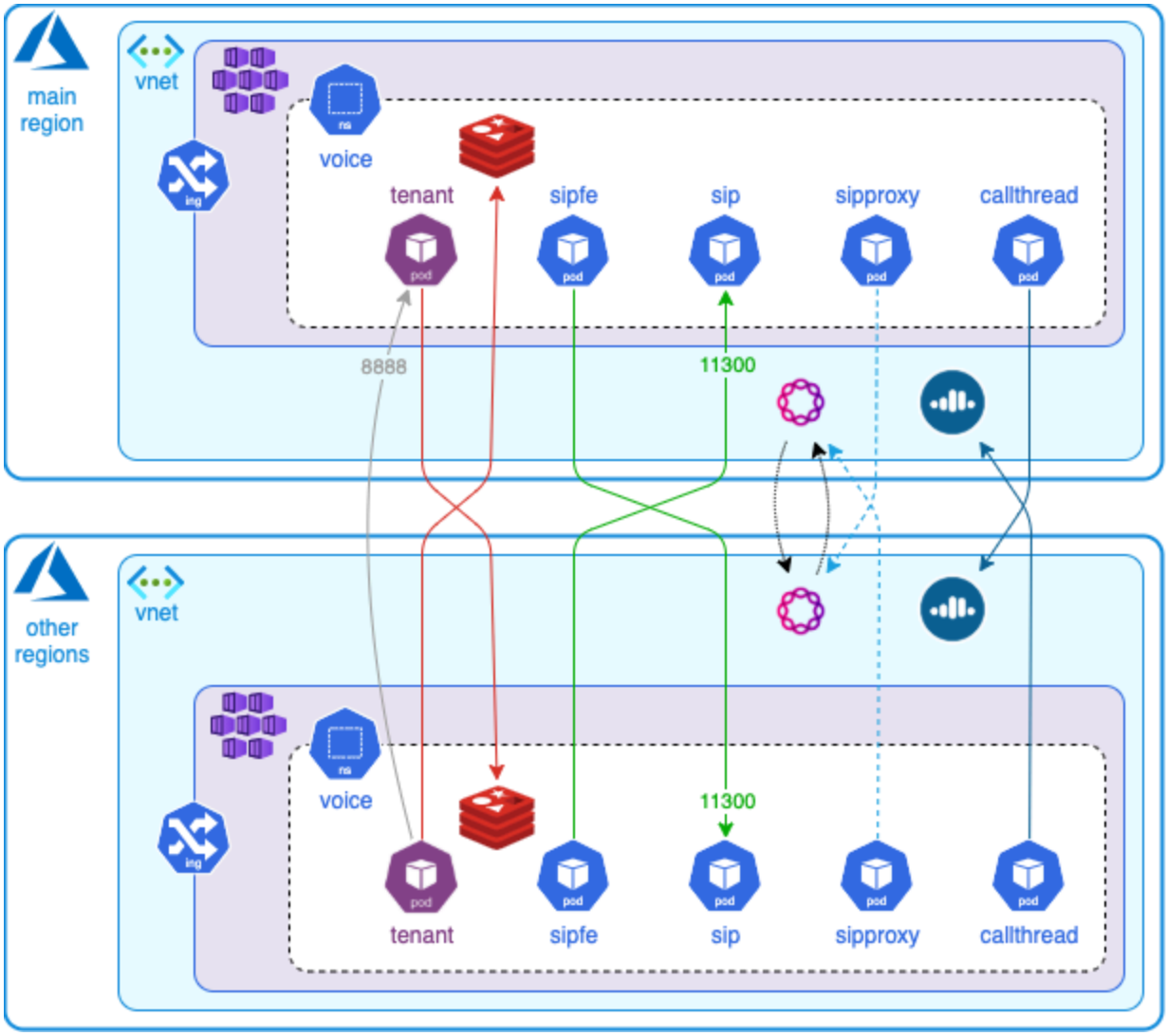
For information about the overall Genesys Multicloud CX Private Edition architecture, see Architecture.

The following diagram shows an example of the high-level architecture for Voice Microservices.



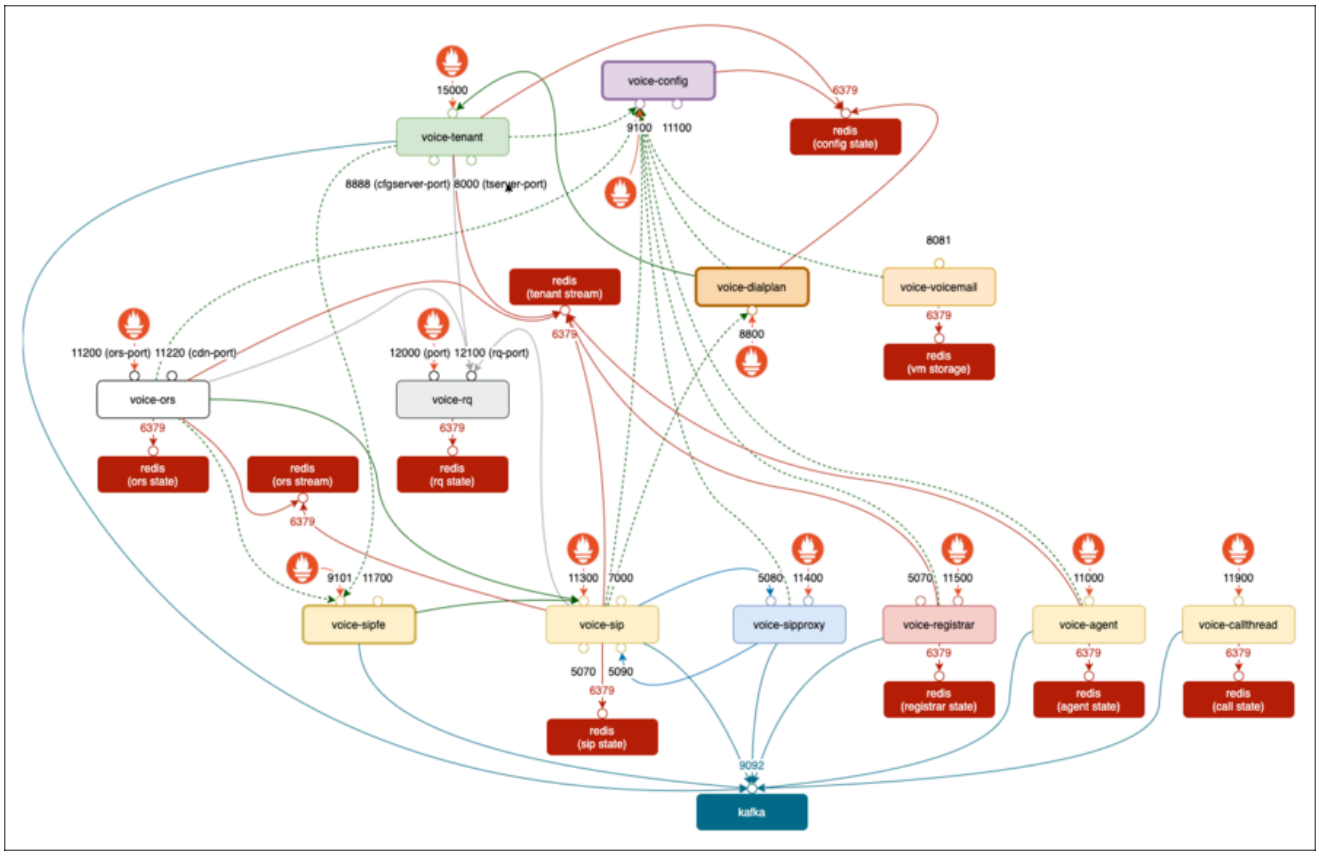
Cross-region architecture

The following diagram shows an example of cross-region architecture for Voice Microservices.



Voice connections

The following diagram illustrates the Voice Services connection architecture.


15000
voice-tenant
8888 (cfgserver-port) 8000 (tserver-port)
voice-config
9100 11100
6379
redis
(config state)
8081
voice-voicemail
6379
redis
(vm storage)
voice-dialplan
8800
redis
(tenant stream)
6379
11200 (ors-port) 11220 (cdn-port)
voice-ors
6379
redis
(ors state)
redis
(ors stream)
6379
12000 (port) 12100 (rq-port)
voice-rq
6379
redis
(rq state)
9101 11700
voice-sipfe
11300 7000
voice-sip
5070 5090
6379
redis
(sip state)
5080 11400
voice-sipproxy
5070 11500
voice-registrar
6379
redis
(registrar state)
11000
voice-agent
6379
redis
(agent state)
11900
voice-callthread
6379
redis
(call state)
9092
kafka

High availability and disaster recovery

Find out how this service provides disaster recovery in the event the service goes down.

Related documentation:

-

Service	High Availability	Disaster Recovery	Where can you host this service?
Voice Microservices	N = N (N+1)	Active-spare	Primary or secondary unit

This information is under development: Flagged items aren't yet confirmed or have info coming soon; Checked items are valid.

See High Availability information for all services: High availability and disaster recovery

Before you begin

Contents

Find out what to do before deploying Voice Microservices.

Related documentation:

-

Limitations and assumptions

Not applicable

Download the Helm charts

For information about how to download the Helm charts, see Downloading your Genesys Multicloud CX containers.

The following services are included with Voice Microservices:

- Voice Agent State Service
- Voice Config Service
- Voice Dial Plan Service
- Voice Front End Service
- Voice Orchestration Service
- Voice Registrar Service
- Voice Call State Service
- Voice RQ Service
- Voice SIP Cluster Service
- Voice SIP Proxy Service
- Voice Voicemail Service
- Voice Tenant Service

See Helm charts and containers for Voice Microservices for the Helm chart version you must download for your release.

For information about the Voicemail Service, see Before you begin in the *Configure and deploy Voicemail* section of this guide.

For information about the Tenant service, also included with Voice Microservices, see the *Tenant Service Private Edition Guide*.

Third-party prerequisites

For information about setting up your Genesys Multicloud CX private edition platform, see Software Requirements.

The following table lists the third-party prerequisites for Voice Microservices.

Third-party services

Name	Version	Purpose	Notes
Redis	6.x	Used for caching. Only distributions of Redis that support Redis cluster mode are supported, however, some services may not support cluster mode.	
Consul	1.9.5 - 1.9.x	Service discovery, service mesh, and key/value store.	For additional information, see Deploy Consul.
Kafka	2.x	Message bus.	

Storage requirements

Content coming soon

Network requirements

Content coming soon

Browser requirements

Not applicable

Genesys dependencies

For detailed information about the correct order of services deployment, see Order of services deployment.

GDPR support

Multi-Tenant Inbound Voice: Voicemail Service

Customer data that is likely to identify an individual, or a combination of other held data to identify an individual is considered as Personally Identifiable Information (PII). Customer name, phone number, email address, bank details, and IP address are some examples of PII.

According to EU GDPR:

- When a customer requests to access personal data that is available with the contact center, the PII associated with the client is exported from the database in client-understandable format. You use the **Export Me** request to do this.
- When a customer requests to delete personal data, the PII associated with that client is deleted from the database within 30 days. However, the Voicemail service is designed in a way that the Customer PII data is deleted in one day using the **Forget Me** request.

Both **Export Me** and **Forget Me** requests depend only on Caller ID/ANI input from the customer. The following PII data is deleted or exported during the **Forget Me** or **Export Me** request process, respectively:

- Voicemail Message
- Caller ID/ANI

GDPR feature is supported only when **StorageInterface' *is configured as*** *BlobStorage****, and*** *Voicemail service* is configured with Azure storage account data store.

Adding caller_id tag during voicemail deposit

Index tag **caller_id** is included in voicemail messages and metadata blob files during voicemail deposit. Using the index tags, you can easily filter the **Forget Me** or **Export Me** instead of searching every mailbox.

GDPR multi-region support

In voicemail service, all voicemail metadata files are stored in master region and voicemail messages are deposited/stored in the respective region. Therefore, It is required to connect all the regions of a tenant to perform Forget Me, Undo Forget Me, or Export Me processes for GDPR inputs.

To provide multi-region support for GDPR, follow these steps while performing GDPR operation:

1. Get the list of regions of a tenant.
2. Ensure all regions storage accounts are up. If any one of storage accounts is down, you cannot perform the GDPR operation.
3. GDPR operates in the master region files, first.
4. Then, GDPR operates in all the non-master region files.

APIs

Voicemail service provides APIs to **Export Me** and **Forget Me** requests of GDPR authenticated with GWS. APIs support any valid client ID. The API can process more than one user's data in a single API request. The API follows the same standard input format used in the current PEC (legacy component).

Forget Me and Export Me API Input.json	Forget Me Undo Input.json

<pre>{
"caseid":"123456789",
"consumers":[
{"consumer":
[
{"name":"John Doe"},
{"name": "John Q. Doe"},
{"phone":"555551212"}
]
},
{"consumer":
[
{"name":"Dan Archer"},
{"phone":"555556161"},
{"phone":"555556162"},
{"email":"danny@hollywood.com"},
{"email":"funnyguy@comedy.org"},
{"fbid":"Dan Archer"}
]
}],
"gim-attached-
data":{"kvlist":["AcctNum", "SSN"]}
}</pre>
	<pre>{
	"caseid":"123456789"
	}</pre>

Voicemail service stores only the caller ANI. Therefore, the voicemail processes the records only with the "phone" parameter from the given input and does not include any other parameters.

Forget Me: API for **Forget me** deletes the PII data related to the consumer after one day based on the API request. The files are deleted through the operations:

- Message and metadata files are reuploaded with **forgetme**=`true` and **case_id**=*[case_id_value]* index tag during the Forget Me API call.
- Deleting files using Azure lifecycle management rules. A rule named **forgetme** is created in Azure lifecycle management. The **Forgetme** rule deletes the file if it meets the following conditions:
  - The file is not modified in a day
  - The file has **forgetme**=`true` index tag

The **Forgetme** rule is executed automatically by Azure lifecycle management once a day. Therefore, there are limitations in deleting files and capturing them in the limitations section.

- **Undo Forget Me**: The API to undo the **Forget Me** request with the same case id.
  If the admin/user has wrongly requested/entered the caller ANI, then the voicemail service provides an option to undo the **Forget Me** request using another API call with the same case ID, to avoid data loss.
- **Export Me**: The API for **Export Me** returns the list of message IDs with message media URL to

download the media.

  - The media URL is also authenticated and authorized with the GWS token.

- The Voicemail Service is exposed via the Kubernetes service, and can be accessed by URL in any region: http://voice-voicemail-service.voice.svc.cluster.local:8081/fs (The FQDN remains same in all the regions wherever voicemail service is deployed).
- Append the API URL with the above-mentioned base URL for accessing the APIs.
- The Voicemail service authenticates and authorizes each request with GWS. The Voicemail service requires the OAuth token in the header for the following the API calls:
  - Authorization: Basic (or) Bearer
  - Contact center ID is taken from the authorization token
- Here is the API definition:
  - **messageId**: Unique message ID of the message.
- The API sample response is given based on the sample input mentioned above.

Configure Voice Microservices

Contents

Learn how to configure Voice Microservices.

Related documentation:

-

Override Helm chart values

For general information about overriding Helm chart values, see Overriding Helm Chart values in the *Genesys Multicloud CX Private Edition Guide*.

If you want to use arbitrary UIDs in your OpenShift deployment, you must override the **securityContext** settings in the **values.yaml** file, so that no user or group IDs are specified. For details, see Security context configuration, below.

When deploying Voice services, certain parameters need to be enabled or modified based on customer requirements and environment. For each of the Voice services, an override **values.yaml** file must be created that overrides certain sections of the default configuration for the service. In this document, we use the following format for creating an override **values.yaml** file: **_override_values.yaml**.

The **override_values.yaml** file contains the following sections:

- Deployment
- Image
- Config
- Secrets
- HPA
- Resources
- Log volume

Deployment section

This section can be used to specify minimum and max instances that will be started for each service. By default, the minimum replica count is 1, and the maximum replica count is 10. You can modify it per your load requirements. For RQ service alone it is recommended to set replica count to 2 or more based on load for high availability.

```
deployment:
namespace: voice      # Namespace of voice service
replicaCount: 1       # Min replica count when service is deployed
maxReplicas: 10       # Max replica count to which the service will scale.
```

Image section

This section has information about the registry from which the voice services will be deployed.

```
image:
  registry: pureengage-docker-staging.jfrog.io  # registry from where image needs to be
deployed
  pullPolicy: Always                            # whether to pull image always
  imagePullSecrets: "mycred"                    # Secrets needed for pulling image from
registry
```

Config section

The config section contains configuration parameters that need to be overridden for all voice services.

Additional information needs to be passed for SIP Service: dnsServer. Get the DNS Server value from the above section (Configure DNS server for voice-sip).

```
# Set the redis port to be used.
context:
  envs:
    redis:
      port: 6379             # Redis port
    dnsServer: "10.202.0.10" # DNS server address. Needed only for SIP Service.
```

Secrets section

This section captures all the secrets needed by voice services for connecting to infraservices (Consul, Kafka, Redis). The default values for Redis and Kafka secrets are the same as what is created above.

```
# set the secrets
secrets:
  redisCache:
    general:
      enabled: true
  consulACL:
    volumes:
    - name: consul-shared-secret
      secret:
        secretName: consul-voice-token
```

HPA section

The HPA section captures whether HPA is enabled for a service or not and what is the CPU and memory percentage used for scale up and scale down. Common HPA for the following voice services: Agent Service, Config Service, Call State Service, Registrar Service, SIP Front End service, Dial Plan Service.

```
hpa:
  targetCPUPercent: 60         # Average CPU percentage which determine scale up and down
  targetMemoryPercent: 60      # Average Memory percentage which determine scale up and down
  enabled: true                # Horizontal Pod scalar enabled
```

For SIPProxy and RQ, HPA is set to false:

```
hpa:
  enabled: false                # Horizontal Pod scalar enabled
```

For SIP and ORS, HPA is set as follows:

```
hpa:
  targetCPUPercent: 50         # Average CPU percentage which determine scale up and down
  targetMemoryPercent: 50      # Average Memory percentage which determine scale up and down
  enabled: true                # Horizontal Pod scalar enabled
```

Resources section

This section captures the resource request and limits for each voice service. The default resource given below is set for each service. You can modify this request and limit based on your load requirement.

```
resources:
  requests:
    cpu: "250m"
    memory: "256Mi"
  limits:
    cpu: "500m"
    memory: "512Mi"
```

For ORS and SIPS service the CPU and memory requirement is high so Genesys recommends the following setting:

```
resources:
  requests:
    cpu: "500m"
    memory: "1Gi"
  limits:
    cpu: "1500m"
    memory: "4Gi"
```

Log volume

This section captures parameters pertaining to log volumes needed by SIP Service. These parameters are needed for storing logging of SIP Server binary that is run inside the SIP Cluster service. The values for **storageClass** and **volumeName** should be configured based on the recommendation given in the Persistent Volume section.

```
# pvc will be created for logs
volumes:
  pvcLog:
    create: true
    claim: sip-log-pvc
    storageClass:
    volumeName:

  pvcJsonLog:
    create: true
    claim: sip-json-log-pvc
    storageClass:
    volumeName:

  log:
    mountPath:
```

```
  jsonLog:
    mountPath:
```

Configure Kubernetes

For information, see the following resources:

- Override Helm chart values
- Configure security
- Secrets for Voice services
- Deploy Voice Microservices

Configure security

Before you deploy the Voice Microservices, be sure to read Security Settings in the *Setting up Genesys Multicloud CX Private Edition* guide.

Security context configuration

The security context settings define the privilege and access control settings for pods and containers. For more information, see the Kubernetes documentation.

By default, the user and group IDs are set in the **values.yaml** file as `500:500:500`, meaning the **genesys** user.

```
containerSecurityContext:
  readOnlyRootFilesystem: false
  runAsNonRoot: true
  runAsUser: 500
  runAsGroup: 500

podSecurityContext:
  fsGroup: 500
  runAsUser: 500
  runAsGroup: 500
  runAsNonRoot: true
```

Arbitrary UIDs in OpenShift

If you want to use arbitrary UIDs in your OpenShift deployment, you must override the **securityContext** settings in the **values.yaml** file, so that you do not define any specific IDs.

```
containerSecurityContext:
  readOnlyRootFilesystem: false
  runAsNonRoot: true
  runAsUser: null
  runAsGroup: 0
```

```
podSecurityContext:
  fsGroup: null
  runAsUser: null
  runAsGroup: 0
  runAsNonRoot: true
```

Secrets for Voice services

Create the following Kubernetes secrets for other infrastructure services:

1. Kafka
2. docker-registry
3. Redis

Kafka secrets

Kafka secrets must be created when Kafka is deployed. The secret is referenced in the Voice Microservices **values.yaml** file.

When Kafka is deployed without authentication, create the secret for Kafka as follows:

```
kubectl create secret generic -n voice kafka-secrets-token --from-literal=kafka-
secrets={\"bootstrap\":}
for ex, kubectl create secret generic -n voice kafka-secrets-token --from-literal=kafka-
secrets={\"bootstrap\":\"infra-kafka-cp-kafka.infra.svc.cluster.local:9092\"}
```

When Kafka is deployed with authentication, create the secret for Kafka using this method:

```
kubectl create secret generic -n voice kafka-secrets-token --from-literal=kafka-
secrets={\"bootstrap\":, \"username\": , \"password\": }
for ex, kubectl create secret generic -n voice kafka-secrets-token --from-literal=kafka-
secrets={\"bootstrap\":\"infra-kafka-cp-
kafka.infra.svc.cluster.local:9092\",\"username\":\"kafka-user\",\"password\":\"kafka-
password\"}
```

Redis secrets

Ensure Redis is installed before you deploy the Voice Services.

Use the following commands to create Redis secrets:

```
export REDIS_PASSWORD=$(kubectl get secret infra-redis-redis-cluster -n infra -o
jsonpath="{.data.redis-password}" | base64 --decode)
kubectl create secret generic -n voice redis-agent-token --from-literal=redis-agent-
state={\"password\":\"$REDIS_PASSWORD\"}
kubectl create secret generic -n voice redis-callthread-token --from-literal=redis-call-
state={\"password\":\"$REDIS_PASSWORD\"}
kubectl create secret generic -n voice redis-config-token --from-literal=redis-config-
state={\"password\":\"$REDIS_PASSWORD\"}
kubectl create secret generic -n voice redis-tenant-token --from-literal=redis-tenant-
stream={\"password\":\"$REDIS_PASSWORD\"}
kubectl create secret generic -n voice redis-registrar-token --from-literal=redis-registrar-
state={\"password\":\"$REDIS_PASSWORD\"}
kubectl create secret generic -n voice redis-sip-token --from-literal=redis-sip-
state={\"password\":\"$REDIS_PASSWORD\"}
kubectl create secret generic -n voice redis-ors-stream-token --from-literal=redis-ors-
```

```
stream={\"password\":\"$REDIS_PASSWORD\"}
kubectl create secret generic -n voice redis-ors-token --from-literal=redis-ors-
state={\"password\":\"$REDIS_PASSWORD\"}
kubectl create secret generic -n voice redis-rq-token --from-literal=redis-rq-
state={\"password\":\"$REDIS_PASSWORD\"}
```

JFrog secrets

Use the following commands to create JFrog secrets:

```
kubectl create secret docker-registry   --docker-server= --docker-username="$JFROG_USER" --
docker-password="$JFROG_PASSWORD" -n voice
```

Provision Voice Microservices

Contents

- Administrator

Learn how to provision Voice Microservices.

Related documentation:

-

Tenant provisioning

Content coming soon

Deploy Voice Microservices

Contents

Learn how to deploy Voice Microservices.

Related documentation:

-

> **Important**
> Make sure to review Before you begin for the full list of prerequisites required to deploy Voice Microservices.
>
> To deploy the Tenant service, see the *Tenant Service Private Edition Guide*.

General deployment prerequisites

Before you deploy the Voice Services, you must deploy the infrastructure services. See Third-party prerequisites for the list of required infrastructure services.

To override values for both the infrastructure services and voice services, see Override Helm chart values.

Genesys recommends the following order of deployment for the Voice Microservices:

- Voice Services
- Tenant Service
- Voicemail Service

Deploy Consul

Consul is required for multiple services in the Genesys package.

In addition to any other Consul configuration, the following Consul features are required for Voice Services:

- connectinject – To deploy sidecar containers in Voice pods.
- controller – To provide service intention functionality.
- openshift – To set OpenShift-specific permissions.
- syncCatalog – To sync Kubernetes services to Consul. Set **toK8S: false** and **addK8SNamespaceSuffix: false** for syncing services from Kubernetes to Consul.

- AccessControlList – To enable ACL, set **manageSystemACLs: true**.
- storageclass – To set the storage class to a predefined storage class.
- TLS – To enable TLS, set **enabled: true** and follow the steps/commands described below to set up TLS.

The file content for the Consul configuration is the following:

```
# config.yaml
global:
  name: consul
  tls:
    enabled: true
    caCert:
      secretName: consul-ca-cert
      # The key of the Kubernetes secret.
      secretKey: tls.crt
    caKey:
      # The name of the Kubernetes secret.
      secretName: consul-ca-key
      # The key of the Kubernetes secret.
      secretKey: tls.key
  acls:
    manageSystemACLs: true
  openshift:
    enabled: true
connectInject:
  enabled: true
controller:
  enabled: true
syncCatalog:
  enabled: true
  toConsul: true
  toK8S: false
  addK8SNamespaceSuffix: false
```

Creation of the Consul bootstrap token

When you enable an Access Control List in Consul, you must ensure that Voice services have access to read and write to Consul. To provide access, you create a token for Voice services in the Consul UI. You can create the necessary Consul bootstrap token when you deploy Consul, although it is possible to do this configuration later, as part of the Voice Services deployment.

When Access Control List (ACL) is enabled in Consul, the Voice services must have the required access for reading and writing into Consul. For that, you must create a token in the Consul UI with the following permissions for the Voice services.

```
service_prefix "" {
  policy = "read"
  intentions = "read"
}
service_prefix "" {
  policy = "write"
  intentions = "write"
}
node_prefix "" {
  policy = "read"
}
node_prefix "" {
  policy = "write"
}
```

```
agent_prefix "" {
  policy = "read"
}
agent_prefix "" {
  policy = "write"
}
session_prefix "" {
  policy = "write"
}
session_prefix "" {
  policy = "read"
}
namespace_prefix "" {
 key_prefix "" {
  policy = "write"
 }
 session_prefix "" {
  policy = "write"
 }
}
key_prefix "" {
  policy = "read"
 }
key_prefix "" {
  policy = "write"
 }
```

To log into the Consul UI and to create a new ACL, you use a bootstrap token. Use the following command to get the bootstrap token:

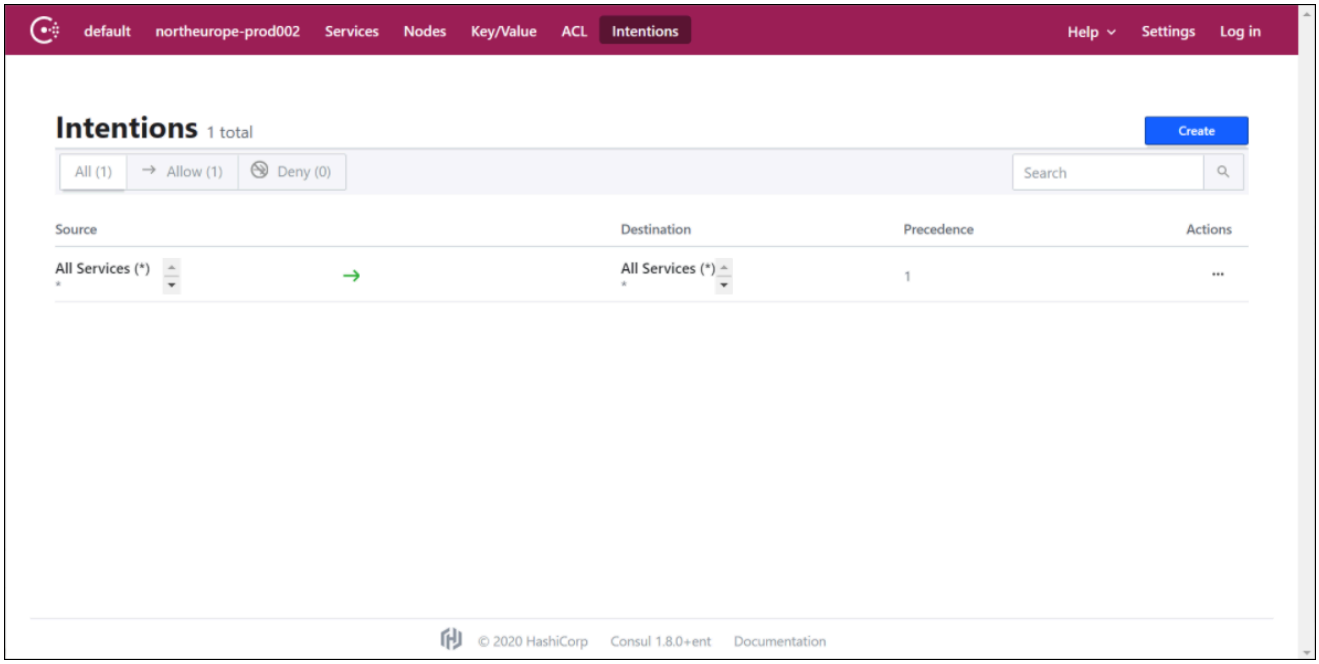
```
kubectl get secret consul-bootstrap-acl-token -n  -o go-template='{{.data.token |
base64decode}}
```

Create a new token and create a policy (voice-policy) with the preceding list of permissions and assign it to this token. For example a token is created with a value of `a7529f8a-1146-e398-8bd7-367894c4b37b`. We can create a Kubernetes secret with this token as shown below:

```
kubectl create secret generic consul-voice-token -n voice --from-literal='consul-consul-voice-
token=a7529f8a-1146-e398-8bd7-367894c4b37b'
```

Creating Intentions in the Consul UI

Voice services use the Consul service mesh to connect between services. Consul has provision to either allow or deny the connection between services. This is done using *intentions*. Log into the **Intentions** tab using the bootstrap token and create a new intention to allow all source services to all destination services as shown in the following screenshot.



Create the Voice namespace

Before deploying Voice Services and their dependencies, create a namespace using the following command:

```
kubectl create ns voice
```

In all Voice Services and the configuration files of their dependencies, the namespace is **voice**. If you want a specific, custom namespace, create the namespace (using the preceding command) and remember to change the namespace in files, as required.

Deploy Voice Services

Register the Redis service in Consul

After the creation of the Redis cluster, the Redis IP address should be registered with Consul. Cluster information needs to be created for the Kubernetes services and endpoints with Redis. Once they are created, Consul will automatically sync those Kubernetes services and register them in Consul.

Kubernetes Service and endpoint creation

The Redis registration should be done for all of the following Redis service names. The Voice services use these service names for connecting to the Redis cluster.

```
redis-agent-state
redis-call-state
redis-config-state
```

```
redis-ors-state
redis-ors-stream
redis-registrar-state
redis-rq-state
redis-sip-state
redis-tenant-stream
```

Manifest file

For all the preceding Redis service names, create a separate service and endpoint using the following example:

```
apiVersion: v1
kind: Service
metadata:
  name:  (ex, redis-agent-state)
  namespace:  (ex, voice)
  annotations:
    "consul.hashicorp.com/service-sync": "true"
spec:
  clusterIP: None
---
apiVersion: v1
kind: Endpoints
metadata:
  name:  (ex, redis-agent-state)
  namespace:  (ex, voice)
subsets:
  - addresses:
      - ip:  (ex, 51.143.122.147)
    ports:
      - port:  (ex, 6379)
        name: redisport
        protocol:  (ex, TCP)
```

In addition, get the Redis primary IP using the following command:

```
kubectl get service infra-redis-redis-cluster -n infra -o jsonpath='{.spec.clusterIP}' (get
Cluster IP of the Redis Service)
```

Deploy the Voice Services

Voice Services require a Persistent Volume Claim (PVC); the Voice SIP Cluster Service uses a persistent volume to store traditional SIP Server logs. Before deploying Voice Services, create the PVC.

Storage class and Claim name

The created persistent volume must be configured in the **sip_node_override_values.yaml** file as shown below:

```
# pvc will be created for logs
volumes:
  pvcLog:
    create: true
    claim: sip-log-pvc
    storageClass: voice
    volumeName:  (ex sip-log-pv)
```

```
  pvcJsonLog:
    create: true
    claim: sip-json-log-pvc
    storageClass: voice
    volumeName:  (ex sip-log-pv)
```

Configure the DNS Server for voice-sip

The Voice SIP Cluster Service requires the DNS server to be configured in its **sip_node_override_values.yaml** file. Follow the steps in the Kubernetes documentation to install a **dnsutils** pod. Using the **dnsutils** pod, get the **dnsserver** that's used in the environment.

The default value in the SIP Helm chart is `10.0.0.10`. If the **dnsserver** address is different, update it in the **sip_node_override_values.yaml** file as shown below:

```
# update dns server ipaddress
  context:
    envs:
      dnsServer: "10.202.0.10"
```

Voice Service Helm chart deployment

Deploy the Voice Services using the provided Helm charts.

```
helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
agent_override_values.yaml voice-agent /voice-agent-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
callthread_override_values.yaml voice-callthread /voice-callthread-.tgz --set version= --
username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
config_override_values.yaml voice-config /voice-config-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
dialplan_override_values.yaml voice-dialplan /voice-dialplan-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
ors_node_override_values.yaml voice-ors /voice-ors-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
registrar_override_values.yaml voice-registrar /voice-registrar-.tgz --set version= --
username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
rq_node_override_values.yaml voice-rq /voice-rq-.tgz --set version= --username "$JFROG_USER"
--password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
sip_node_override_values.yaml voice-sip /voice-sip-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
```

```
sipfe_override_values.yaml voice-sipfe /voice-sipfe-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
sipproxy_override_values.yaml voice-sipproxy /voice-sipproxy-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"
```

The following table contains a list of the minimum recommended Helm chart versions that should be used:

Service name	Helm chart version
voice-config	voice-config-9.0.11.tgz
voice-dialplan	voice-dialplan-9.0.08.tgz
voice-registrar	voice-registrar-9.0.14.tgz
voice-agent	voice-agent-9.0.10.tgz
voice-callthread	voice-callthread-9.0.12.tgz
voice-sip	voice-sip-9.0.22.tgz
voice-sipfe	voice-sipfe-9.0.06.tgz
voice-sipproxy	voice-sipproxy-9.0.09.tgz
voice-rq	voice-rq-9.0.08.tgz
voice-ors	voice-ors-9.0.08.tgz

Deploy in OpenShift

Add a rule for Consul DNS forwarding

In the OpenShift Container Platform (OCP), as part of the general deployment prerequisites, you must add a rule for Consul DNS forwarding. OpenShift sends DNS requests to the DNS server in the **openshift-dns** namespace. To forward Consul FQDN resolution to a Consul DNS server, add the forwarding rule to the **configmap** of the default DNS operator. Save the Consul DNS IP address using the following command:

```
kubectl get svc consul-dns -n  -o jsonpath={.spec.clusterIP} (Internal IP of consul-dns
service)
> oc edit dns.operator/default
Add the below specs:
spec:
servers:
- name: consul-dns
zones:
- consul
forwardPlugin:
upstreams:
-
```

Persistent volumes

The general Voice Services deployment is described in Deploy Voice Services. There are some

differences when creating PVCs in the OpenShift Container Platform (OCP). This section describes the configuration for OCP.

Persistent Volume in OCS Storage Type

A Storage Class might have been created already in the OCP. This Storage Class is used for creating PVCs and must be set in the override values of the **sip_node_override_values.yaml** file.

For an OpenShift cluster with OpenShift Container Storage (OCS), configure the Storage Class to be used for creating the persistent volume. In the case of OCS, the PV is created automatically when the PVC is claimed. For such clusters, the **volumeName** parameter in the **sip_node_override_values.yaml** file must be empty.

```
#  PVC's section
## This section defines about creating PVCs
volumes:
  pvcLog:
    create: true                                    # create defines whether a PVC needs to
be created with the chart.
    claim: sip-log-pvc                              # Name of PVC
    volumeName:                                     # To bind this PVC to specified
Persistent Volume.In case of Openshift, this is required only for NFS mounting and not needed
for OCS.
    claimSize: 10Gi                                 # This field sets the storage size
requested by PVC
    storageClass: voice                             # This field sets the storage class
requested by PVC
    mountPath: /opt/genesys/logs/volume             # Volume mount path for PV

  pvcJsonLog:
    create: true                                    # create defines whether a PVC needs to
be created with the chart.
    claim: sip-json-log-pvc                         # Name of PVC
    volumeName:                                     # To bind this PVC to specified
Persistent Volume.In case of Openshift, this is required only for NFS mounting and not needed
for OCS.
    claimSize: 10Gi                                 # This field sets the storage size
requested by PVC
    storageClass: voice                             # This field sets the storage class
requested by PVC
    mountPath: /opt/genesys/logs/sip_node/JSON      # Volume mount path for PV
```

Configure the DNS Server for voice-sip

The Voice SIP Cluster Service requires the DNS server to be configured in its **sip_node_override_values.yaml** file.

In the OCP environment, you can find the Kubernetes DNS server name using the following command:

```
oc get dns.operator/default -o jsonpath={.status.clusterIP}
```

The default value in the SIP Helm chart is "10.0.0.10"; if the DNS server address is different, update it in the **sip_node_override_values.yaml** file as shown below:

```
# update dns server ipaddress
  context:
    envs:
```

```
    dnsServer: "10.202.0.10"
```

Deploy the Tenant service

The Tenant Service is included with the Voice Microservices, but has its own deployment procedure. To deploy the Tenant Service, see Deploy the Tenant Service.

Validate the deployment

Content coming soon

Upgrade, rollback, or uninstall Voice Microservices

Contents

Learn how to upgrade, rollback or uninstall Voice Microservices.

Related documentation:

-

Upgrade Voice Microservices

Because Voice Services are real-time services, you use canary-based deployment to upgrade. The canary deployment is a technique of deploying one or more canary instances with the new version and verification of the new version to ensure it works as expected and also works with the previous version. Deploying only one or two canary instances should be sufficient to discover a faulty version and to minimize the risk of adding a new version into production.

The upgrade procedure consists of these major steps:

1. Canary deployment
2. Upgrade
3. Delete canary

Canary deployment

For any new Voice Service version, the canary instance of it is deployed, and after the new version of the canary is approved, this version is rolled out to all instances of a Voice Service using the procedure covered in upgrade section.

For the canary deployment, some parameters in the **canary_override_values.yaml** file must be overridden. This file is passed to the Helm chart during the deployment of the canary instance.

```
# serviceaccount is created during initial deployment
serviceAccount:
create: false

deployment:
postfix: canary

# configmap is already created during initial deployment
context:
create: false

# this is needed for SIP canary only
loggingSidecar:
context:
create: false

# this is also needed for SIP canary only
volumes:
pvcLog:
```

```
create: false
pvcJsonLog:
create: false

# podmonitor is not needed for canary, but metric server enabling is needed
prometheus:
podMonitor:
enabled: false
metricServer:
enabled: true

# canary does not need HPA
hpa:
enabled: false
```

The following commands deploy a canary instance:

```
helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
agent_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-agent-
canary /voice-agent-.tgz --set version= --username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
callthread_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-
callthread-canary /voice-callthread-.tgz --set version= --username "$JFROG_USER" --password
"$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
config_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-config-
canary /voice-config-.tgz --set version= --username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
dialplan_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-
dialplan-canary /voice-dialplan-.tgz --set version= --username "$JFROG_USER" --password
"$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
ors_node_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-ors-
canary /voice-ors-.tgz --set version= --username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
registrar_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-
registrar-canary /voice-registrar-.tgz --set version= --username "$JFROG_USER" --password
"$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
rq_node_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-rq-
canary /voice-rq-.tgz --set version= --username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
sip_node_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-sip-
canary /voice-sip-.tgz --set version= --username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
sipfe_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-sipfe-
canary /voice-sipfe-.tgz --set version= --username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
sipproxy_override_values.yaml -f ./voice_helm_values/canary_override_values.yaml voice-
sipproxy-canary /voice-sipproxy-.tgz --set version= --username "$JFROG_USER" --password
"$JFROG_PASSWORD"
```

Service upgrade

When the canary deployment of a Voice Service is ready for an upgrade, use the following commands to upgrade the current version of a Voice Service to the desired version:

```
helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
agent_override_values.yaml voice-agent /voice-agent-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
callthread_override_values.yaml voice-callthread /voice-callthread-.tgz --set version= --
username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
config_override_values.yaml voice-config /voice-config-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
dialplan_override_values.yaml voice-dialplan /voice-dialplan-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
ors_node_override_values.yaml voice-ors /voice-ors-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
registrar_override_values.yaml voice-registrar /voice-registrar-.tgz --set version= --
username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
rq_node_override_values.yaml voice-rq /voice-rq-.tgz --set version= --username "$JFROG_USER"
--password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
sip_node_override_values.yaml voice-sip /voice-sip-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
sipfe_override_values.yaml voice-sipfe /voice-sipfe-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"

helm upgrade --install --force --wait --timeout 300s -n voice -f ./voice_helm_values/
sipproxy_override_values.yaml voice-sipproxy /voice-sipproxy-.tgz --set version= --username
"$JFROG_USER" --password "$JFROG_PASSWORD"
```

Delete the canary instance

If the upgrade of a Voice Service is successful, delete the canary instance of the service by using the following commands:

```
helm delete voice-agent-canary -n voice
helm delete voice-callthread-canary -n voice
helm delete voice-config-canary -n voice
helm delete voice-dialplan-canary -n voice
helm delete voice-ors-canary -n voice
helm delete voice-registrar-canary -n voice
helm delete voice-sip-canary -n voice
helm delete voice-sipfe-canary -n voice
helm delete voice-sipproxy-canary -n voice
```

Upgrade of the RQ node service

The upgrade procedure of the RQ node service differs from other Voice Services and consists of the following steps:

1. Set the strategy to **OnDelete** in **rq_node_override_values.yam**. Note that when a fresh RQ node service is deployed, the strategy is set to **RollingUpdate** in **rq_node_override_values.yaml** by default.
   Example:

   ```
   deployment:
     deploymentType: statefulset
     strategy: OnDelete
   ```

2. Upgrade the voice-rq Helm to the newer version using the following command:

   ```
   helm upgrade --install --force --wait --timeout 200s -n voice -f ./voice_helm_values/
    rq_node_override_values.yaml voice-rq https:///voice-rq/voice-rq-9.0.07.tgz --set
    version=9.0.6 --username "$JFROG_USER" --password "$JFROG_PASSWORD"
   ```

3. Delete the voice-rq-0 pod, and then the voice-rq-0 pod will be upgraded to a new version. Note that only when a pod is deleted, the upgraded Helm version will be considered to new pods. And this canary pod can be verified to ensure it works with other RQ nodes.
4. If other RQ node pods are deleted, they would also get upgraded to a newer version. To avoid such random upgrade of RQ nodes, downgrade Helm version to a previous version. And voice-rq-0 will have a new version available for testing.
5. If a canary pod (voice-rq-0) works correctly with other pods and in the environment, upgrade the voice-rq Helm to the newer version (same as step 2). When the upgrade is successful, delete all RQ pods, so the newer RQ node pods will have the upgraded new version.

Rollback Voice Microservices

Content coming soon

Uninstall Voice Microservices

Content coming soon

Before you begin

Contents

Find out what to do before deploying the Voicemail Service.

Related documentation:

-

Limitations and assumptions

Voice Voicemail Service integration with Workspace Web Edition and Web Services and Applications is in progress. The integration brings an appearance of actionable voice mailbox information in the Workspace Web Edition UI, presenting users with Message Waiting Indicator(s) for each voice mailbox assigned to them either directly as a personal mailbox or as a group mailbox via membership in a group(s) having a mailbox provisioned. Users still have access to voicemail from Workspace Web Edition by dialing directly to a voicemail access number, which is 5555.

Download the Helm charts

For information about how to download the Helm charts, see Downloading your Genesys Multicloud CX containers.

The following table identifies the Helm chart version associated with the Voicemail service.

Service name	Helm chart version
voice-voicemail	voice-voicemail-9.0.xx.tgz

Third-party prerequisites

See the Third-party prerequisites for the Voice Services.

Storage requirements

Choosing Voicemail storage

To store mailbox metadata and messages, consider the following supported options for storage in the Private Edition deployment:

1. Persistent Volumes & Persistent Volume Claims

2. Azure Blob Storage

See the following sections to learn how to use these storage options and to find information about their limitations.

Persistent Volume & Claim

- Persistent Volume (PV) is a piece of storage that can be mounted to a Voicemail Service deployment inside the Kubernetes cluster.
- PVs in OpenShift can be created with different plugins
  - Plugin Reference: Kubernetes Persistent Volumes, Claims, Storage Classes, and More
- Voicemail Service requires a separate storage class and PV to be created for a Voicemail storage.
- If the customer wants to extend the deployment to more than one Kubernetes cluster, Voicemail Service requires to mount the same PV for all the Kubernetes cluster for that customer.
- Create the Persistent Volume Claim (PVC) from the Voicemail PV.
- The access mode for the PVC must be **ReadWriteMany**, since the Voicemail Service will edit the existing data while updating the mailbox settings or the message state.
- Use the sizing doc, which you can find on the (Genesys SIP Feature Server landing page, to calculate the required storage space.

Here is the sample Kubernetes YAML file for creating PVCs for a Voicemail Service. The PVC creation is controlled by the Voicemail Service Helm chart by overriding the **values.yaml**.

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: voice-voicemail-pvc
  namespace: voice
  labels:
    servicename: voice-voicemail
spec:
  storageClassName: voice-voicemail
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 20Gi
```

Limitations

1. Replication strategies are not available for the data.
2. Retention limit: Admins can't configure the auto-expiration for a Voicemail message.
3. When a customer has more than one Kubernetes cluster deployed, the PV for all the Kubernetes clusters must be created from a single storage drive, so that the data from one Kubernetes cluster is shared among other Kubernetes clusters.

Azure blob

- Unlike PV, the Azure Blob Storage provides options to replicate and configure Time to live for the files and can be accessed from any Kubernetes cluster by using the storage access keys.
- Create the Azure Storage with the blob storage.
- The access keys for the blob storage must be securely mounted to the Voicemail pod. You can do one of the following:
  - Store access keys in Azure Key Vault and mount it via a Container Storage Interface (CSI) driver.
  - Create access keys as a Kubernetes secret and volume mount the Kubernetes secret. (This option is considered less secured than the CSI driver approach.)
- The **values.yaml** file can be overridden for configuring either a Kubernetes secret or CSI driver, which is explained in Override Helm chart values.

Network requirements

Content coming soon

Browser requirements

Not applicable

Genesys dependencies

For detailed information about the correct order of services deployment, see Order of services deployment.

GDPR support

Customer data that is likely to identify an individual, or a combination of other held data to identify an individual is considered as Personally Identifiable Information (PII). Customer name, phone number, email address, bank details, and IP address are some examples of PII.

Multi-Tenant Inbound Voice: Voicemail Service

According to EU GDPR:

- When a customer requests to access personal data that is available with the contact center, the PII associated with the client is exported from the database in client-understandable format. You use the

Export Me request to do this.

- When a customer requests to delete personal data, the PII associated with that client is deleted from the database within 30 days. However, the Voicemail service is designed in a way that the Customer PII data is deleted in one day using the **Forget Me** request.

Both **Export Me** and **Forget Me** requests depend only on Caller ID/ANI input from the customer. The following PII data is deleted or exported during the **Forget Me** or **Export Me** request process, respectively:

- Voicemail Message
- Caller ID/ANI

GDPR feature is supported only when **StorageInterface' *is configured as*** *BlobStorage****, and*** *Voicemail service* is configured with Azure storage account data store.

Adding caller_id tag during voicemail deposit

Index tag **caller_id** is included in voicemail messages and metadata blob files during voicemail deposit. Using the index tags, you can easily filter the **Forget Me** messages instead of searching every mailbox.

GDPR multi-region support

In voicemail service, all voicemail metadata files are stored in master region and voicemail messages are deposited/stored in the respective region. Therefore, it is required to connect all the regions of a tenant to perform Forget Me, Undo Forget Me, or Export Me processes for GDPR inputs.

To provide multi-region support for GDPR, follow these steps while performing GDPR operation:

1. Get the list of regions of a tenant.
2. Ensure all regions storage accounts are up. If any one of storage accounts is down, you cannot perform the GDPR operation.
3. GDPR operates in the master region files, first.
4. Then, GDPR operates in all the non-master region files.

APIs

Voicemail service provides APIs to **Export Me** and **Forget Me** requests of GDPR authenticated with GWS. APIs support any valid client ID. The API can process more than one user's data in a single API request. The API follows the same standard input format used in the current PEC (legacy component).

Forget Me and Export Me API Input.json	Forget Me Undo Input.json

<pre>{
"caseid":"123456789",
"consumers":[
{"consumer":
[
{"name":"John Doe"},</pre>
	<pre>{
	"caseid":"123456789"
	}</pre>

```
            {"name": "John Q. Doe"},
            {"phone":"555551212"}
         ]
      },
      {"consumer":
         [
            {"name":"Dan Archer"},
            {"phone":"555556161"},
            {"phone":"555556162"},
            {"email":"danny@hollywood.com"},
            {"email":"funnyguy@comedy.org"},
            {"fbid":"Dan Archer"}
         ]
      }],
   "gim-attached-
data":{"kvlist":["AcctNum", "SSN"]}
 }
```

Voicemail service stores only the caller ANI. Therefore, voicemail processes the records only with the "phone" parameter from the given input and does not include any other parameters.

Forget Me: API for **Forget Me** deletes the PII data related to the consumer after one day based on the API request. The files are deleted through the operations:

- Message and metadata files are reuploaded with **forgetme**=`true` and **case_id**=*[case_id_value]* index tag during the **Forget Me** API call.
- Deleting files using Azure lifecycle management rules. A rule named **forgetme** is created in Azure lifecycle management. The **forgetme** rule deletes the file if it meets the following conditions:
  - The file is not modified in a day
  - The file has **forgetme**=`true` index tag

The **forgetme** rule is executed automatically by Azure lifecycle management once a day. Therefore, there are limitations in deleting files and capturing them in the limitations section.

- **Undo Forget Me**: The API to undo the **Forget Me** request with the same case id.
  If the admin/user has wrongly requested/entered the caller ANI, then the voicemail service provides an option to undo the **Forget Me** request using another API call with the same case ID, to avoid data loss.
- **Export Me**: The API for **Export Me** returns the list of message IDs with message media URL to download the media.
  - The media URL is also authenticated and authorized with the GWS token.
- The Voicemail Service is exposed via the Kubernetes service, and can be accessed by URL in any region: http://voice-voicemail-service.voice.svc.cluster.local:8081/fs (The FQDN remains same in all the regions wherever voicemail service is deployed).
- Append the API URL with the above-mentioned base URL for accessing the APIs.
- The Voicemail service authenticates and authorizes each request with GWS. The Voicemail service requires the OAuth token in the header for the following the API calls:
  - Authorization: Basic (or) Bearer
  - Contact center ID is taken from the authorization token

- Here is the API definition:
  - **messageId**: Unique message ID for the message.
- The API sample response is given based on the sample input mentioned above.

API	API HTTP Method	Sample Success Response

/api/messages/forget	DELETE	Response Status: 200 OK
		Response Body:
		{
		"caseid":"123456789",
		"consumers":
		{
		"555551212": "Deleted x messages deposited from the caller",
		"555556161": "No messages available for deletion from the caller",
		"555556162": "Deleted y messages deposited from the caller"
		}
		}

/api/messages/forget/undo	POST	Response Status: 200 OK
		{
		"caseid":"123456789"
		}
		}

/api/messages/export	POST	Response Status: 200 OK
		Response Body:
		{
		"caseid":"123456789",
		"consumers":
		{
		"555551212": ["/api/messages/export/:mailboxId/:messageId", "/api/messages/export/:mailboxId/:messageId"], //list of message URLs
		"555556161": [],
		"555556162": ["/api/messages/export/:mailboxId/:messageId"]
		}

/api/messages/export/:messageId	GET	Response Status: 200 OK Response Body: Filestream of the message

Here is the API response if there is a failure:

Response Status: 500 Response Body:

```
{
reason: ""
}
```

Standalone Scripts

You can invoke the **Forget Me** and **Export Me** APIs from a standalone Node.js script. This script can be executed by a user or an automated scheduler. When a user executes the script:

- The script authenticates with the user auth token.
- The user must have the bearer or the basic token.

In the case of an automated scheduler, the script uses the client credential (also known as system account) and processes the request. In this scenario, the user has to configure the GWS URL as an environment variable. The script would generate the auth token for the client and access the GDPR APIs. The script can be integrated into the GitHub Actions pipeline and invoked from the GitHub pipeline.

Script Inputs

Parameter	Value	Is it mandatory	Description
-i or --input	file-path	Yes	Input.json is the same as the JSON input passed to the REST API. The client credentials ccid (contact center id) must be included as key-value pair in the input.json file because ccid cannot be fetched from auth token of the client credentials. Sample value "ccid" : "2c5ea4c0-4067-11e9-8bad-9b1deb4d3b7d"
-o or --output	output-location	Yes	output.json Forget Me Operation: The output.json is the same as the response from the Forget Me API. Export Me Operation: { "caseid":"123456789", "consumers": {

			//The message media is exported in the output location and the filename is the same as the message IDs.
			"555551212": ["filename of message1", "filename of message2"],
			"555556161": [],
			"555556162": ["filename of message1"]
			}
			}
			
			Undo Operation:
			
			The output.json is the same as the response from Undo API.
			
			execution.log
			
			Execution logs are available in the execution.log file

-u or --user	User token	Either user token or client credentials	User token is fetched from GWS
-c or --client	Client credentials	Either user token or client credentials	Client credential is required when scheduling the script. Client credentials can be obtained by requesting the GWS team.
-p or --operation	forgetme exportme undoforgot	Yes	Type of operation to be done when the script is executed.

Limitations

MWI count is not updated automatically on deleting the files during **Forgetme** operation. It is updated during the next voicemail message deposit or voicemail message delete of a mailbox.

- If the **Forgetme** rule is first executed at 10:00 UTC in the day, then the file **X** marked for **Forgetme** at 10.01 UTC same day, the **Forgetme** rule does not delete the file 'X' on the second day at 10:00 UTC since it does not meet the **file has not been modified in one day** condition. However, it gets deleted in next day.
- If the message is deposited and not read by any agent, the **Forget Me** API is executed and marked for deletion. Before deleting the file, if the agent reads the message/forward, the message metadata and the last modified time are updated. In such cases, the file may not be deleted in one day because the last modified date condition is not met.

Configure the Voicemail Service

Contents

Learn how to configure the Voicemail Service.

Related documentation:

-

Override Helm chart values

Content coming soon

Provision the Voicemail Service

Learn how to provision the Voicemail Service.

Related documentation:

-

Content coming soon

Deploy the Voicemail Service

Learn how to deploy the Voicemail Service.

Related documentation:

-

Content coming soon

Upgrade, rollback, or uninstall the Voicemail Service

Learn how to upgrade, rollback or uninstall the Voicemail Service.

Related documentation:

-

Content coming soon

Observability in Voice Microservices

Contents

Learn about the logs, metrics, and alerts you should monitor for Voice Microservices.

Related documentation:

-

Monitoring

Private edition services expose metrics that can be scraped by Prometheus, to support monitoring operations and alerting.

- As described on Monitoring overview and approach, you can use a tool like Grafana to create dashboards that query the Prometheus metrics to visualize operational status.
- As described on Customizing Alertmanager configuration, you can configure Alertmanager to send notifications to notification providers such as PagerDuty, to notify you when an alert is triggered because a metric has exceeded a defined threshold.

The services expose a number of Genesys-defined and third-party metrics. The metrics that are defined in third-party software used by private edition services are available for you to use as long as the third-party provider still supports them. For descriptions of available Voice Microservices metrics, see:

- Agent State Service metrics
- Call State Service metrics
- Config Service metrics
- Dial Plan Service metrics
- FrontEnd Service metrics
- ORS metrics
- Voice Registrar Service metrics
- Voice RQ Service metrics
- Voice SIP Cluster Service metrics
- Voice SIP Proxy Service metrics
- Voicemail metrics

See also System metrics.

Deploy dashboard and alert dashboards

Deploy dashboards and alert rules using these Helm charts:

- **voice-dashboards** – This installs the dashboards that are created to monitor various Voice Services.
- **voice-alertrules** – This installs the alert rules that specify what type of alarm must be triggered based on the metrics.

```
helm repo add helm-staging https:// --username "$JFROG_USER" --password "$JFROG_PASSWORD"
helm repo update

helm install voice-alertrules -n voice https:///voice-monitoring/voice-alertrules-1.0.5.tgz --
username "$JFROG_USER" --password "$JFROG_PASSWORD"

helm install voice-dashboards -n voice https:///voice-monitoring/voice-dashboards-1.0.8.tgz --
username "$JFROG_USER" --password "$JFROG_PASSWORD"
```

Enable monitoring

You can expose metrics on a service-by-service basis. To do so, edit the **Values.yaml** file associated with each service, and enable metrics using either the **Prometheus** operator, or **Prometheus Annotation**.

```
prometheus:
  # Enable for Prometheus Annotation
  metricServer:
    enabled: false
    path: /metrics
```

OR

```
  # Enable for Prometheus operator
  podMonitor:
    enabled: false
    path: /metrics
    interval: 30s
```

Service	CRD or annotations?	Port	Endpoint/ Selector	Metrics update interval
Agent State Service	PodMonitor	11000	http://:11000/ metrics	30 seconds
Call State Service	Supports both CRD and annotations	11900	http://:11900/ metrics	30 seconds
Config Service	Supports both CRD and annotations	9100	http://:9100/ metrics	30 seconds
Dial Plan Service	Supports both CRD and annotations	8800	http://:8800/ metrics	30 seconds
FrontEnd Service	Supports both CRD and annotations	9101	http://:9101/ metrics	30 seconds
ORS	Supports both CRD and annotations	11200	http://:11200/ metrics	30 seconds
Voice Registrar Service	Supports both CRD and annotations	11500	http://:11500/ metrics	30 seconds
Voice RQ Service	Supports both CRD and annotations	12000	http://:12000/ metrics	30 seconds
Voice SIP Cluster	Supports both CRD	11300	http://:11300/	30 seconds

Service	CRD or annotations?	Port	Endpoint/ Selector	Metrics update interval
Service	and annotations		metrics	
Voice SIP Proxy Service	Supports both CRD and annotations	11400	http://:11400/ metrics	30 seconds
Voicemail	Supports both CRD and annotations	8081	http://:8081/ metrics	30 seconds

Configure metrics

The metrics that are exposed by the Voice Microservices are available by default. No further configuration is required in order to define or expose these metrics. You cannot define your own custom metrics.

The Metrics pages linked to above show some of the metrics the Voice Microservices expose. You can also query Prometheus directly or via a dashboard to see all the metrics available from the Voice Microservices.

Alerting

Private edition services define a number of alerts based on Prometheus metrics thresholds.

> **Important**
> You can use general third-party functionality to create rules to trigger alerts based on metrics values you specify. Genesys does not provide support for custom alerts that you create in your environment.

For descriptions of available Voice Microservices alerts, see:

- Agent State Service alerts
- Call State Service alerts
- Config Service alerts
- Dial Plan Service alerts
- FrontEnd Service alerts
- ORS alerts
- Voice Registrar Service alerts
- Voice RQ Service alerts
- Voice SIP Cluster Service alerts
- Voice SIP Proxy Service alerts

- Voicemail alerts

Configure alerts

Private edition services define a number of alerts by default (for Voice Microservices, see the pages linked to above). No further configuration is required.

The alerts are defined as **PrometheusRule** objects in a **prometheus-rule.yaml** file in the Helm charts. As described above, Voice Microservices does not support customizing the alerts or defining additional **PrometheusRule** objects to create alerts based on the service-provided metrics.

Logging

Voice Microservices can write logs generated by internal components to the following locations:

- Persistent Volume/Persistent Volume Claim with RWX storage. For more information, see Log volume, Deploy the Voice Services, and Persistent volumes.
- Ephemeral volume (emptyDir) with a Fluent Bit logging sidecar that tails log files and sends them to standard output (stdout). For more information, see Forwarding logs to stdout.

Forwarding logs to stdout

You can optionally forward logs from internal components to stdout using a logging sidecar (Genesys currently supports Fluent Bit) and an ephemeral volume (emptyDir). The Fluent Bit sidecar tails the logs and sends them to stdout. For more information, see Sidecar processed logging in the *Genesys Multicloud CX Private Edition Operations guide*.

The SIPNode logs within the SIP Cluster service can be forwarded to stdout. By default, forwarding logs to stdout is disabled. To enable the log forwarding option, set the following parameters in the voice-sip Helm Chart **values.yaml** file:

```
volumes:
   # Mount an Ephemeral Volume for storing the legacy SIPS logs.
  sipsLog:
   mountPath: "/opt/genesys/logs/sip_node/SIPS"

sipsLoggingSidecar:
  enabled: true                              # sips-logging-sidecar container will be created
  image:
    registry: genesysengagedev.azurecr.io    # Registry from where the images will be fetched
    repository: sre/fluent-bit               # repository and folder where the particular
service image is located
    pullPolicy: Always                       # Policy to pull always or only when the image is
not there
    tag: 1.8.x                               # fluent-bit version that will be used by the
logging sidecar
  context:
    create: true                             #Create configmap for sips-logging-sidecar.Set to
```

```
true in Values.yaml and set to false in canary_values.yaml
```

Agent State Service metrics and alerts

Find the metrics Agent State Service exposes and the alerts defined for Agent State Service.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Agent State Service	PodMonitor	11000	http://:11000/metrics	30 seconds

See details about:

- Agent State Service metrics
- Agent State Service alerts

Metrics

Voice Agent State Service exposes Genesys-defined, Agent State Service–specific metrics as well as some standard Kafka metrics. You can query Prometheus directly to see all the metrics that the Agent State Service exposes. The following metrics are likely to be particularly useful. Genesys does not commit to maintain other currently available Agent State Service metrics not documented on this page.

Metric and description	Metric details	Indicator of
agent_redis_state Current Redis connection state: -1 - error 0 - disconnected 1 - connected 2 - ready	**Unit:** N/A **Type:** gauge **Label:** location, redis_cluster_name **Sample value:** 2	
agent_stream_redis_state Current Tenant Redis connection state: 0 - disconnected 1 - connected	**Unit:** N/A **Type:** gauge **Label:** location, redis_cluster_name **Sample value:** 1	
agent_total_sessions Total number of agent sessions.	**Unit:** N/A **Type:** gauge **Label:** tenant **Sample value:**	Saturation
agent_callevents Total number of received call events.	**Unit:** N/A **Type:** counter **Label:** tenant **Sample value:**	Traffic
agent_logged_in_agents Number of logged-in agents.	**Unit:** N/A **Type:** gauge **Label:** tenant **Sample value:**	Saturation

Metric and description	Metric details	Indicator of
agent_health_level Health level of the agent node: -1 - error 0 - fail 1 - degraded 2 - pass	**Unit:** N/A **Type:** gauge **Label:** tenant **Sample value:** 2	Traffic
agent_envoy_proxy_status Status of the Envoy proxy: -1 - error 0 - disconnected 1 - connected	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 1	
agent_config_node_status Status of the config node connection: 0 - disconnected 1 - connected	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 1	
http_client_request_duration_seconds HTTP client time from request to response, in seconds.	**Unit:** seconds **Type:** histogram **Label:** target_service_name **Sample value:**	
http_client_response_count HTTP client responses received.	**Unit:** N/A **Type:** counter **Label:** target_service_name, tenant, status **Sample value:**	Traffic
kafka_consumer_recv_messages_total Number of messages received from Kafka.	**Unit:** N/A **Type:** counter **Label:** topic, tenant, kafka_location **Sample value:**	Traffic
kafka_consumer_error_total Number of Kafka consumer errors.	**Unit:** N/A **Type:** counter **Label:** topic, kafka_location **Sample value:**	Errors
kafka_consumer_latency Consumer latency is the time difference between when the message is produced and when the message is consumed. That is, the time when the consumer received the message minus the time when the producer produced the message.	**Unit:** **Type:** histogram **Label:** topic, tenant, kafka_location **Sample value:**	Latency
kafka_consumer_rebalance_total Number of Kafka consumer re-balance events.	**Unit:** N/A **Type:** counter **Label:** topic, kafka_location	

Metric and description	Metric details	Indicator of
	Sample value:	
kafka_consumer_state Current state of the Kafka consumer.	**Unit:** N/A **Type:** gauge **Label:** topic, kafka_location **Sample value:**	
kafka_producer__messages_total Number of messages received from Kafka.	**Unit:** N/A **Type:** counter **Label:** topic, tenant, kafka_location **Sample value:**	
kafka_producer_queue_depth Number of Kafka producer pending events.	**Unit:** N/A **Type:** gauge **Label:** kafka_location **Sample value:**	Saturation
kafka_producer_queue_age_seconds Age of the oldest producer pending event in seconds.	**Unit:** seconds **Type:** gauge **Label:** kafka_location **Sample value:**	
kafka_producer_error_total Number of Kafka producer errors.	**Unit:** N/A **Type:** counter **Label:** kafka_location **Sample value:**	
kafka_producer_state Current state of the Kafka producer.	**Unit:** N/A **Type:** gauge **Label:** kafka_location **Sample value:**	
log_output_bytes_total Total amount of log output, in bytes.	**Unit:** bytes **Type:** counter **Label:** level, format, module **Sample value:**	

Alerts

The following alerts are defined for Agent State Service.

Alert	Severity	Description	Based on	Threshold
Kafka events latency is too high	Warning	Actions:	kafka_consumer_latency_bucket	Latency for more than 5% of messages is more

Alert	Severity	Description	Based on	Threshold
		• If the alarm is triggered for multiple topics, ensure there are no issues with Kafka (CPU, memory, or network overload). • If the alarm is triggered only for topic {{ $labels.topic }}, check if there is an issue with the service related to the topic (CPU, memory, or network overload).		than 0.5 seconds for topic {{ $labels.topic }}.
Possible messages lost	Critical	Actions: • Check Kafka and {{ $labels.job }} service overload, network degradation.	kafka_consumer_recv_messages_total, kafka_producer_sent_messages_total	Number of sent requests is two times higher than received for topic {{ $labels.topic }}.
Too many Kafka consumer failed health checks	Warning	Actions: • If the alarm is triggered for multiple services, ensure there are no issues with Kafka. Restart Kafka. • If the alarm is triggered only for container {{ $labels.container }}, check if there is an issue with the service.	kafka_consumer_error_total	Health check failed more than 10 times in 5 minutes for Kafka consumer for topic {{ $labels.topic}}.
Too many Kafka	Warning	Actions:	kafka_consumer_error_total	More than 10

Alert	Severity	Description	Based on	Threshold
consumer request timeouts		• If the alarm is triggered for multiple services, ensure there are no issues with Kafka. Restart Kafka. • If the alarm is triggered only for container {{ $labels.container }}, check if there is an issue with the service.		request timeouts appeared in 5 minutes for Kafka consumer for topic {{ $labels.topic}}.
Too many Kafka consumer crashes	Critical	Actions: • If the alarm is triggered for multiple services, ensure there are no issues with Kafka. Restart Kafka. • If the alarm is triggered only for container {{ $labels.container }}, check if there is an issue with the service.	kafka_consumer_error_total	More than 3 Kafka consumer crashes in 5 minutes for service {{ $labels.container }}.
Pod status Failed	Warning	Actions: • Restart the pod. Check if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Failed state.
Pod status Unknown	Warning	Actions: • Restart the pod. Check if there are any	kube_pod_status_phase	Pod {{ $labels.pod }} is in Unknown state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		issues with pod after restart.		
Pod status Pending	Warning	Actions: • Restart the pod. Check if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Pending state for 5 minutes.
Pod status NotReady	Critical	Actions: • Restart the pod. Check if there are any issues with the pod after restart.	kube_pod_status_ready	Pod {{ $labels.pod }} is in NotReady status for 5 minutes.
Container restarted repeatedly	Critical	Actions: • Check if the new version of the image was deployed. • Check for issues with the Kubernetes cluster.	kube_pod_container_status_restarts_total	Container {{ $labels.container }} was restarted 5 or more times within 15 minutes.
Max replicas is not sufficient for 5 mins	Critical	The desired number of replicas is higher than the current available replicas for the past 5 minutes.	kube_statefulset_replicas, kube_statefulset_status_replicas	The desired number of replicas is higher than the current available replicas for the past 5 minutes.
Kafka not available	Critical	Actions: • If the alarm is triggered for multiple services, ensure there are no issues with Kafka. Restart Kafka. • If the alarm is	kafka_producer_state, kafka_consumer_state	Kafka is not available for pod {{ $labels.pod }} for 5 consecutive minutes.

Alert	Severity	Description	Based on	Threshold
		triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.		
Redis not available	Critical	Actions: • If the alarm is triggered for multiple services, ensure there are no issues with Redis. Restart Redis. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.	agent_redis_state, agent_stream_redis_state	Redis is not available for pod {{ $labels.pod }} for 5 consecutive minutes.
Agent service fail	Critical	Actions: • Check if there is any problem with pod {{ $labels.pod }}, then restart the pod.	agent_health_level	Agent health level is Fail for pod {{ $labels.pod }} for 5 consecutive minutes.
Config node fail	Warning	Actions: • Check if there is any problem with pod {{ $labels.pod }} and the config node.	http_client_response_count	Requests to the config node fail for 5 consecutive minutes.
Pod CPU greater than 65%	Warning	High CPU load for pod {{ $labels.pod }}.	container_cpu_usage_seconds_total, container_spec_cpu_period	Container {{ $labels.container }} CPU usage exceeded 65% for 5 minutes.
Pod CPU greater than 80%	Critical	Critical CPU load for pod {{	container_cpu_usage_seconds_total, container_spec_cpu_period	Container {{ $labels.container

Alert	Severity	Description	Based on	Threshold
		$labels.pod }}.		}} CPU usage exceeded 80% for 5 minutes.
Pod memory greater than 65%	Warning	High memory usage for pod {{ $labels.pod }}.	container_memory_working_set_bytes, kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 65% for 5 minutes.
Pod memory greater than 80%	Critical	Critical memory usage for pod {{ $labels.pod }}.	container_memory_working_set_bytes, kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 80% for 5 minutes.
Too many Kafka pending events	Critical	Actions: • Ensure there are no issues with Kafka or {{ $labels.pod }} pod's CPU and network.	kafka_producer_queue_depth	Too many Kafka producer pending events for pod {{ $labels.pod }} (more than 100 in 5 minutes).

Call State Service metrics and alerts

Find the metrics Call State Service exposes and the alerts defined for Call State Service.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Call State Service	Supports both CRD and annotations	11900	http://:11900/metrics	30 seconds

See details about:

- Call State Service metrics
- Call State Service alerts

Metrics

Voice Call State Service exposes Genesys-defined, Call State Service–specific metrics as well as some standard Kafka metrics. You can query Prometheus directly to see all the metrics that the Call State Service exposes. The following metrics are likely to be particularly useful. Genesys does not commit to maintain other currently available Call State Service metrics not documented on this page.

Metric and description	Metric details	Indicator of
callthread_call_threads Number of monitored call threads.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Saturation
callthread_envoy_proxy_status Status of the envoy proxy: -1 - error 0 - disconnected 1 – connected	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
callthread_health_level Health level of the agent node: -1 - error 0 - fail 1 - degraded 2 - pass	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
callthread_healthcheck_generic_exception Generic error during health check.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
callthread_redis_state Current Redis connection state: -1 – error	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors

Metric and description	Metric details	Indicator of
0 – disconnected 1 – connected 2 – ready		
http_client_request_duration_seconds HTTP client time from request to response, in seconds.	**Unit:** seconds **Type:** histogram **Label:** target_service_name **Sample value:**	
http_client_response_count The number of HTTP client responses received.	**Unit:** N/A **Type:** counter **Label:** target_service_name, tenant, status **Sample value:**	
kafka_consumer_recv_messages_total Number of messages received from Kafka.	**Unit:** N/A **Type:** counter **Label:** topic, tenant, kafka_location **Sample value:**	Traffic
kafka_consumer_error_total Number of Kafka consumer errors.	**Unit:** N/A **Type:** counter **Label:** topic, kafka_location **Sample value:**	Errors
kafka_consumer_latency Consumer latency is the time difference between when the message is produced and when the message is consumed. That is, the time when the consumer received the message minus the time when the producer produced the message.	**Unit:** **Type:** histogram **Label:** topic, tenant, kafka_location **Sample value:**	Latency
kafka_consumer_rebalance_total Number of Kafka consumer re-balance events.	**Unit:** N/A **Type:** counter **Label:** topic, kafka_location **Sample value:**	
kafka_consumer_state Current state of Kafka consumer.	**Unit:** N/A **Type:** gauge **Label:** topic, kafka_location **Sample value:**	
kafka_producer__messages_total Number of messages received from Kafka.	**Unit:** N/A **Type:** counter **Label:** topic, tenant, kafka_location **Sample value:**	Traffic
kafka_producer_queue_depth Number of Kafka producer pending events.	**Unit:** N/A **Type:** gauge **Label:** kafka_location **Sample value:**	Saturation

Metric and description	Metric details	Indicator of
kafka_producer_queue_age_seconds Age of the oldest producer pending event, in seconds.	**Unit:** seconds **Type:** gauge **Label:** kafka_location **Sample value:**	
kafka_producer_error_total Number of Kafka producer errors.	**Unit:** N/A **Type:** counter **Label:** kafka_location **Sample value:**	Errors
kafka_producer_state Current state of the Kafka producer.	**Unit:** N/A **Type:** gauge **Label:** kafka_location **Sample value:**	
log_output_bytes_total Total amount of log output, in bytes.	**Unit:** bytes **Type:** counter **Label:** level, format, module **Sample value:**	

Alerts

The following alerts are defined for Call State Service.

Alert	Severity	Description	Based on	Threshold
Kafka events latency is too high	Critical	Actions: • If the alarm is triggered for multiple topics, ensure there are no issues with Kafka (CPU, memory, or network overload). • If the alarm is triggered only for topic {{ $labels.topic }}, check if there is an issue with the service related to the topic (CPU, memory,	kafka_consumer_latency_bucket	Latency for more than 5% of messages is more than 0.5 seconds for topic {{ $labels.topic }}.

Alert	Severity	Description	Based on	Threshold
		or network overload).		
Too many Kafka consumer failed health checks	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Kafka, and then restart Kafka. • If the alarm is triggered only for {{ $labels.container }}, check if there is an issue with the service.	kafka_consumer_error_total	Health check failed more than 10 times in 5 minutes for Kafka consumer for topic {{ $labels.topic }}.
Too many Kafka consumer request timeouts	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Kafka, and then restart Kafka. • If the alarm is triggered only for {{ $labels.container }}, check if there is an issue with the service.	kafka_consumer_error_total	More than 10 request timeouts appeared in 5 minutes for Kafka consumer for topic {{ $labels.topic }}.
Too many Kafka consumer crashes	Critical	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Kafka, and	kafka_consumer_error_total	More than 3 Kafka consumer crashes in 5 minutes for topic {{ $labels.topic }}.

Alert	Severity	Description	Based on	Threshold
		then restart Kafka. • If the alarm is triggered only for {{ $labels.container }}, check if there is an issue with the service.		
Pod status Failed	Warning	Actions: • Restart the pod. Check if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Failed state.
Pod status Unknown	Warning	Actions: • Restart the pod. Check if there are any issues with pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Unknown state for 5 minutes.
Pod status Pending	Warning	Actions: • Restart the pod. Check if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Pending state for 5 minutes.
Pod status NotReady	Critical	Actions: • Restart the pod. Check if there are any issues with the pod after restart.	kube_pod_status_ready	Pod {{ $labels.pod }} is in NotReady status for 5 minutes.
Container restarted repeatedly	Critical	Actions: • Check if the new version of the image was deployed.	kube_pod_container_status_restarts_total	Container {{ $labels.container }} was restarted 5 or more times within 15 minutes.

Alert	Severity	Description	Based on	Threshold
		• Check for issues with the Kubernetes cluster.		
Max replicas is not sufficient for 5 mins	Critical	The desired number of replicas is higher than the current available replicas for the past 5 minutes.	kube_statefulset_replicas, kube_statefulset_status_replicas	The desired number of replicas is higher than the current available replicas for the past 5 minutes.
Kafka not available	Critical	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Kafka, and then restart Kafka. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.	kafka_producer_state, kafka_consumer_state	Kafka is not available for pod {{ $labels.pod }} for 5 consecutive minutes.
Redis not available	Critical	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, and then restart Redis. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.	callthread_redis_state	Redis is not available for pod {{ $labels.pod }} for 5 consecutive minutes.
Pod CPU greater than 65%	Warning	High CPU load for pod {{ $labels.pod	container_cpu_usage_seconds_total, container_spec_cpu_period	Container {{ $labels.container

Alert	Severity	Description	Based on	Threshold
		}}.		}} CPU usage exceeded 65% for 5 minutes.
Pod CPU greater than 80%	Critical	Critical CPU load for pod {{ $labels.pod }}.	container_cpu_usage_seconds_total container_spec_cpu_period	Container {{ $labels.container }} CPU usage exceeded 80% for 5 minutes.
Pod memory greater than 65%	Warning	High memory usage for pod {{ $labels.pod }}.	container_memory_working_set_bytes kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 65% for 5 minutes.
Pod memory greater than 80%	Critical	Critical memory usage for pod {{ $labels.pod }}.	container_memory_working_set_bytes kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 80% for 5 minutes.
Too many Kafka pending events	Critical	Actions: • Ensure there are no issues with Kafka or {{ $labels.container }} service's CPU and network.	kafka_producer_queue_depth	Too many Kafka producer pending events for service {{ $labels.container }} (more than 100 in 5 minutes).

Config Service metrics and alerts

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

Contents

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 Current Redis connection state: -1 – error 0 – disconnected 1 – connected 2 – ready	**Unit:** N/A **Type:** gauge **Label:** location, redis_cluster_name **Sample value:** 2	Errors
service_version_info	**Unit:** N/A	

Metric and description	Metric details	Indicator of
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.	**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 pod.	redis_state	Redis is not available for pod {{ $labels.pod }} for 5 minutes.
Redis disconnected for 10 minutes	Critical	Actions: • If the alarm is	redis_state	Redis is not available for the pod {{ $labels.pod }} for 10 minutes.

Alert	Severity	Description	Based on	Threshold
		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.		
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.
Pod Pending state	Warning	Actions: • If the alarm is triggered for	kube_pod_status_phase	Pod {{ $labels.pod }} is in Pending state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		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.		
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 the horizontal pod autoscaler has triggered	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
		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 pod autoscaler has triggered and if the	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
		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.

Dial Plan Service metrics and alerts

Find the metrics Dial Plan Service exposes and the alerts defined for Dial Plan Service.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Dial Plan Service	Supports both CRD and annotations	8800	http://:8800/metrics	30 seconds

See details about:

- Dial Plan Service metrics
- Dial Plan Service alerts

Metrics

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

Metric and description	Metric details	Indicator of
dialplan_health_level Aggregated health level of the dialplan node for dependent services such as Redis and the Envoy sidecar connection: -1 – fail 0 – starting 1 – degraded 2 – pass	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 2	Health
dialplan_redis_state Current Redis connection state: 0 – disconnected 1 – connecting 2 – connected	**Unit:** N/A **Type:** gauge **Label:** redis_cluster_name **Sample value:** 2	Health
dialplan_total_request Number of dialplan requests received.	**Unit:** N/A **Type:** counter **Label:** tenant, pod, operation_type **Sample value:**	Traffic
dialplan_failure_response The number of Dial Plan failure responses.	**Unit:** N/A **Type:** counter **Label:** tenant, pod, operation_type, status, reason **Sample value:**	Traffic
dialplan_response_time Dialplan request processing duration histogram, in ms.	**Unit:** milliseconds **Type:** histogram **Label:**	Latency

Metric and description	Metric details	Indicator of
	Sample value:	
dialplan_redis_cache_latency_msec Redis fetch latency, measured in milliseconds.	**Unit:** milliseconds **Type:** histogram **Label:** tenant **Sample value:**	Latency

Alerts

The following alerts are defined for Dial Plan Service.

Alert	Severity	Description	Based on	Threshold
DialPlan processing time > 0.5 seconds	Warning	Actions: • If the alarm is generated for all dialplan pods, then Redis or network delay might be the most probable cause. • If the alarm is generated in a single dialplan pod, then it might be due to Envoy or a network issue.	dialplan_response_time	When the latency for 95% of the dial plan messages is more than 0.5 seconds for a duration of 5 minutes, then this warning alarm is raised for the {{ $labels.container }}.
DialPlan processing time > 2 seconds	Critical	Actions: • If the alarm is generated for all dialplan pods, then Redis or network delay might be the most probable cause. • If the alarm is generated in a single dialplan	dialplan_response_time	If the latency for 95% of the dial plan messages is more than 2 seconds for a duration of 5 minutes, then this warning alarm is raised for the {{ $labels.container }}.

Alert	Severity	Description	Based on	Threshold
		pod, then it might be due to Envoy or a network issue.		
Aggregated service health failing for 5 minutes	Critical	Actions: • Check the dialplan dashboard for Aggregated Service Health errors and, in case of a Redis error, first check for any issues/crashes in the pod and then restart Redis. • In the case of an Envoy error, the dialplan container will be restarted by the liveness probe. If the issue still exists after that, restart the pod.	dialplan_health_level	Dependent services or the Envoy sidecar is not available for 5 minutes in the pod {{ $labels.pod }}.
Redis disconnected for 5 minutes	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis and 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 5 minutes.
Redis disconnected for 10 minutes	Critical	Actions:	redis_state	Redis is not available for the pod {{ $labels.pod

Alert	Severity	Description	Based on	Threshold
		• If the alarm is triggered for multiple services, make sure there are no issues with Redis and 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.		}} 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 {{ $labels.pod }} failed.
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 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.
Pod Pending state	Warning	Actions: • If the alarm is triggered for	kube_pod_status_phase	Pod {{ $labels.pod }} is in the Pending state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		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.		
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 the NotReady state for 10 minutes.
Pod memory greater than 65%	Warning	High 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. • Collect the service logs;	container_memory_working_set_bytes, kube_pod_container_resource_limits	Container {{ $labels.container }} memory usage exceeded 65% for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		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_limits	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 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	container_cpu_usage_seconds_total, kube_pod_container_resource_limits	Container {{ $labels.container }} CPU usage exceeded 65% for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		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, kube_pod_container_resource_limits	Container {{ $labels.container }} CPU usage exceeded 80% for 5 minutes.

FrontEnd Service metrics and alerts

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

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
FrontEnd Service	Supports both CRD and annotations	9101	http://:9101/metrics	30 seconds

See details about:

- FrontEnd Service metrics
- FrontEnd Service alerts

Metrics

Voice FrontEnd Service exposes Genesys-defined, FrontEnd Service–specific metrics as well as some standard Kafka metrics. You can query Prometheus directly to see all the metrics that the FrontEnd Service exposes. The following metrics are likely to be particularly useful. Genesys does not commit to maintain other currently available FrontEnd Service metrics not documented on this page.

Metric and description	Metric details	Indicator of
kafka_producer_queue_depth Number of Kafka producer pending events.	**Unit:** N/A **Type:** gauge **Label:** kafka_location **Sample value:** 0	
kafka_producer_queue_age_seconds Age of the oldest producer pending event, in seconds.	**Unit:** seconds **Type:** gauge **Label:** kafka_location **Sample value:**	
kafka_producer_error_total Number of Kafka producer errors.	**Unit:** N/A **Type:** counter **Label:** kafka_location **Sample value:**	
kafka_producer_state Current state of the Kafka producer.	**Unit:** N/A **Type:** gauge **Label:** kafka_location **Sample value:**	
kafka_producer_biggest_event_size Biggest event size so far.	**Unit:** **Type:** gauge **Label:** kafka_location, topic **Sample value:** 515	
kafka_max_request_size Exposed config to compare with biggest	**Unit:** **Type:** gauge	

Metric and description	Metric details	Indicator of
event size.	**Label:** kafka_location **Sample value:**	
log_output_bytes_total Total amount of log output, in bytes.	**Unit:** bytes **Type:** counter **Label:** level, format, module **Sample value:**	
sipfe_requests_total Number of requests.	**Unit:** N/A **Type:** counter **Label:** tenant **Sample value:**	Traffic
sipfe_responses_total Number of responses for the requests.	**Unit:** N/A **Type:** counter **Label:** tenant **Sample value:**	Traffic
sipfe_sip_nodes_total Number of SIP nodes that are alive.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
sipfe_sip_node_requests_total Number of requests to the SIP nodes.	**Unit:** N/A **Type:** counter **Label:** sip_node_id, tenant **Sample value:**	
sipfe_sip_node_responses_total Number of responses from the SIP nodes for the requests.	**Unit:** N/A **Type:** counter **Label:** sip_node_id, tenant, status **Sample value:**	
sipfe_sip_node_request_duration_seconds The duration of time between the SIP node request and the response, measured in seconds.	**Unit:** seconds **Type:** histogram **Label:** le, sip_node_id, tenant, status **Sample value:**	Latency
service_version_info Displays the version of Voice FrontEnd 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:** **Type:** gauge **Label:** version **Sample value:** service_version_info{version="100.0.1000006"} 1	
sipfe_health_level Health level of the sipfe node: -1 – fail 0 – starting 1 – degraded	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 2	Errors

Metric and description	Metric details	Indicator of
2 – pass		
sipfe_health_check_error Health check errors for the sipfe node: 1 – has error 0 – no error	**Unit:** N/A **Type:** gauge **Label:** reason **Sample value:** 0	Errors

Alerts

The following alerts are defined for FrontEnd Service.

Alert	Severity	Description	Based on	Threshold
Too many Kafka pending producer events	Critical	Actions: • Make sure there are no issues with Kafka or {{ $labels.pod }} pod's CPU and network.	kafka_producer_queue_depth	Too many Kafka producer pending events for pod {{ $labels.pod }} (more than 100 in 5 minutes).
Too many received requests without a response	Critical	Actions: • Collect the service logs for pod {{ $labels.pod }}; raise an investigation ticket. • Restart the service.	sipfe_requests_total	For too many requests, the Front End service at pod {{ $labels.pod }} did not send any response (more than 100 requests without a response, measured over 5 minutes).
SIP Cluster Service response latency is too high	Critical	Actions: • If the alarm is triggered for multiple pods, make sure there are no issues with the SIP Cluster Service (CPU,	sipfe_sip_node_request_duration_seconds_bucket	Latency for 95% of messages is more than 0.5 seconds for service {{ $labels.container }}.

Alert	Severity	Description	Based on	Threshold
		memory, or network overload). • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod (CPU, memory, or network overload).		
No requests received	Critical	Absence of received requests for pod {{ $labels.pod }}. Actions: • For pod {{ $labels.pod }}, make sure there are no issues with Orchestration Service and Tenant Service or the network to them.	sipfe_requests_total	increase(sipfe_requests_total{pod .+"}[5m]) 100
Too many failure responses sent	Critical	Too many failure responses are sent by the Front End service at pod {{ $labels.pod }}. Actions: • For pod {{ $labels.pod }}, make sure received requests are valid.	sipfe_responses_total	More than 100 failure responses in 5 consecutive minutes.
Too many Kafka producer errors	Critical	Kafka responds with errors at pod {{ $labels.pod }}. Actions: • For pod {{	kafka_producer_error_total	More than 100 errors in 5 consecutive minutes.

Alert	Severity	Description	Based on	Threshold
		$labels.pod }}, make sure there are no issues with Kafka.		
Too many SIP Cluster Service error responses	Critical	SIP Cluster Service responds with errors at pod {{ $labels.pod }}. Actions: • If the alarm is triggered for multiple pods, make sure there are no issues with the SIP Cluster Service (CPU, memory, or network overload). • If the alarm is triggered only for pod {{ $labels.pod }}, check if there are issues with requests sent by the pod.	sipfe_sip_node_responses_total	More than 100 errors in 5 consecutive minutes.
Kafka not available	Critical	Kafka is not available for pod {{ $labels.pod }}. Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Kafka, and then restart Kafka. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there	kafka_producer_state	Kafka is not available for pod {{ $labels.pod }} for 5 consecutive minutes.

Alert	Severity	Description	Based on	Threshold
		is an issue with the pod.		
SIP Node(s) is not available	Critical	No available SIP Nodes for pod {{ $labels.pod }}. Actions: • If the alarm is triggered for multiple services, make sure there are no issues with SIP Nodes, and then restart SIP Nodes. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod or the network to SIP Nodes.	sipfe_sip_nodes_total	No available SIP Nodes for pod {{ $labels.pod }} for 5 consecutive minutes.
Pod status Failed	Warning	Pod {{ $labels.pod }} is in Failed state. Actions: • Restart the pod. Check to see if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Failed state.
Pod status Unknown	Warning	Pod {{ $labels.pod }} is in Unknown state for 5 minutes. Actions: • Restart the pod. Check to see if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Unknown state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
Pod status Pending	Warning	Pod {{ $labels.pod }} is in Pending state for 5 minutes. Actions: • Restart the pod. Check to see if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Pending state for 5 minutes.
Pod status NotReady	Critical	Pod {{ $labels.pod }} is in the NotReady state for 5 minutes. Actions: • Restart the pod. Check to see if there are any issues with the pod after restart.	kube_pod_status_ready	Pod {{ $labels.pod }} is in the NotReady state for 5 minutes.
Container restarted repeatedly	Critical	Container {{ $labels.container }} was restarted 5 or more times within 15 minutes. Actions: • Check if a new version of the image was deployed. • Check for issues with the Kubernetes cluster.	kube_pod_container_status_restarts_total	Container {{ $labels.container }} was restarted 5 or more times within 15 minutes.
Max replicas is not sufficient for 5 mins	Critical	For the past 5 minutes, the desired number of replicas is higher than the number of replicas currently available. Actions:	kube_statefulset_replicas, kube_statefulset_status_replicas	Desired number of replicas is higher than current available replicas for the past 5 minutes.

Alert	Severity	Description	Based on	Threshold
		• Check resources available for Kubernetes. Increase resources, if necessary.		
Pods scaled up greater than 80%	Critical	For the past 5 minutes, the desired number of replicas is greater than the number of replicas currently available. Actions: • Check resources available for Kubernetes. Increase resources, if necessary.	kube_hpa_status_current_replicas, kube_hpa_spec_max_replicas	(kube_hpa_status_current_replicas{node-hpa"} * 100) / kube_hpa_spec_max_replicas{node-hpa"} > 80 for: 5m
Pods less than Min Replicas	Critical	The current number of replicas is lower than the minimum number of replicas that should be available. Actions: • Check if Kubernetes cannot deploy new pods or if pods are failing in their status to be active/ read.	kube_hpa_status_current_replicas, kube_hpa_spec_min_replicas	For the past 5 minutes, the current number of replicas is lower than the minimum number of replicas that should be available.
Pod CPU greater than 65%	Warning	High CPU load for pod {{ $labels.pod }}. Actions: • Check whether the horizontal pod autoscaler	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
		has triggered and if the maximum number of pods has been reached. • Check Grafana for abnormal load. • Collect the service logs for pod {{ $labels.pod }}; 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.	container_cpu_usage_seconds_total, container_spec_cpu_period	Container {{ $labels.container }} CPU usage exceeded 80% for 5 minutes.
Pod memory greater than 65%	Warning	High memory usage for pod {{ $labels.pod }}. Actions: • Check whether the horizontal pod autoscaler has triggered and if the maximum number of	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
		pods has been reached. • Check Grafana for abnormal load. • Collect the service logs for pod {{ $labels.pod }}; 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 for pod {{ $labels.pod }}.	container_memory_working_set_bytes kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 80% for 5 minutes.

ORS metrics and alerts

Find the metrics ORS exposes and the alerts defined for ORS.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
ORS	Supports both CRD and annotations	11200	http://:11200/metrics	30 seconds

See details about:

- ORS metrics
- ORS alerts

Metrics

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

Metric and description	Metric details	Indicator of
orsnode_callevents Total number of received call events.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
orsnode_ha_writes The number of HA writes to Redis.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
orsnode_ha_reads The number of HA reads from Redis.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
orsnode_interactions The number of active interactions.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
orsnode_total_interactions The total number of interactions that have been created.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
orsnode_cleared_interactions The total number of call interactions that have been cleared.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	

Metric and description	Metric details	Indicator of
orsnode_strategies The number of strategies that are running.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
orsnode_total_strategies The total number of strategies that have been created.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
orsnode_load_errors The total number of strategy load errors.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_fetch_errors The total number of errors encountered when a strategy tried to fetch data from a Designer Application Server (DAS).	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_config_errors The total number of strategy configuration errors.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_invoke_errors The total number of strategy invoke errors.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_treatments The total number of strategy treatments.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
orsnode_failed_treatments The total number of failed strategy treatments.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_userdata_updates The total number of times that a strategy updated user data.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
orsnode_scxml_transitions The total number of SCXML transitions.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
orsnode_scxml_events	**Unit:** N/A	Traffic

Metric and description	Metric details	Indicator of
The total number of SCXML events.	**Type:** counter **Label:** **Sample value:**	
orsnode_scxml_error_events The total number of SCXML `error.*` events.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_http_fetch_requests The total number of HTTP `fetch` requests.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_http_fetch_duration The HTTP fetch time, measured in milliseconds (ms).	**Unit:** milliseconds **Type:** histogram **Label:** **Sample value:**	Latency
orsnode_http_fetch_errors The total number of HTTP `fetch` errors.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_http_fetch_error_status Status of the HTTP `fetch` error.	**Unit:** **Type:** histogram **Label:** **Sample value:**	Errors
orsnode_urs_rlib_latency_msec The Universal Routing Server (URS) `rlib` latency, measured in milliseconds (ms).	**Unit:** milliseconds **Type:** histogram **Label:** **Sample value:**	Latency
orsnode_urs_rlib_errors The total number of URS `rlib` errors.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_urs_rlib_requests The total number of URS `rlib` requests.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_urs_rlib_events The total number of URS `rlib` events.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_urs_rlib_timeouts	**Unit:** N/A	

Metric and description	Metric details	Indicator of
The total number of URS `rlib` timeouts.	**Type:** counter **Label:** **Sample value:**	
orsnode_redis_state Current Redis connection state.	**Unit:** N/A **Type:** gauge **Label:** redis_cluster_name **Sample value:**	
orsnode_redis_disconnect The number of times that the ORS node disconnected from Redis.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_sdr_messages_sent The number of SDR messages that have been sent.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_rq_latency_msec Redis queue latency, measured in milliseconds (ms).	**Unit:** milliseconds **Type:** histogram **Label:** le, service **Sample value:**	Latency
orsnode_routing_latency_msec Routing latency, measured in milliseconds (ms).	**Unit:** milliseconds **Type:** histogram **Label:** **Sample value:**	Latency
orsnode_rstream_latency_msec Redis stream latency, measured in (ms).	**Unit:** milliseconds **Type:** histogram **Label:** le, node **Sample value:**	Latency
orsnode_digital_latency_msec Digital stream latency, measured in milliseconds (ms).	**Unit:** milliseconds **Type:** histogram **Label:** **Sample value:**	Latency
orsnode_sip_health_check ORS health check.	**Unit:** N/A **Type:** gauge **Label:** node **Sample value:**	
orsnode_ixn_health_check Interaction health check.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_rq_state	**Unit:** N/A	

Metric and description	Metric details	Indicator of
Current Redis queue connection state.	**Type:** gauge **Label:** **Sample value:**	
orsnode_ixn_events Total number of interaction stream events received.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_rq_disconnect Number of times the ORS node disconnected from the RQ Service.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
service_version_info Displays the version of Voice Orchestration 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	
orsnode_route_redirected Total number of EventRouteUsed events without a ReferenceID.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_balancer_stream_state The state of the voice balancer stream.	**Unit:** N/A **Type:** gauge **Label:** balancer_stream_type **Sample value:**	
orsnode_high_memory Indicates when the ORS node is using a lot of memory.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_urs_rlib_state Indicates a Tenant `rlib` request timeout.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_stuck_interactions The number of stuck interactions.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_urs_scxml_submit_requests The total number of URS `SCXMLSubmit` requests.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	

Metric and description	Metric details	Indicator of
orsnode_urs_scxml_cancel_requests The total number of URS `SCXMLQueueCancel` requests.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_urs_queue_submit_done_events Total number of URS `queue.submit.done` events.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_health_level Summarized health level of the ORS node: -1 – fail 0 – starting 1 – degraded 2 – pass	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_health_check_error Health check errors for the ORS node: 1 – has error 0 – no error	**Unit:** N/A **Type:** gauge **Label:** reason **Sample value:**	Errors
orsnode_running_applications The number of active sessions for each Designer application.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_failed_applications The number of failed sessions for each Designer application.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_total_applications The total number of sessions created for each Designer application.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_failed_scripts The number of scripts that failed to load in the Tenant Service configuration management environment.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_session_load_time_msec The time it takes for the strategy to be compiled and go through its initial states.	**Unit:** milliseconds **Type:** histogram **Label:** **Sample value:**	
orsnode_service_started Timestamp when the ORS node started.	**Unit:** N/A **Type:** gauge	

Metric and description	Metric details	Indicator of
	Label: started **Sample value:**	
orsnode_total_terminal_requests Total number of terminal requests (like Deliver, PlaceInQueue, StopProcessing for Digital and RequestClearCall, RequestRouteCall for Voice).	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_total_non_terminal_requests Total number of non-terminal requests to the Interaction Server (for Digital) or SIP Server (for Voice).	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_sip_post_errors Total number of errors encountered in POST requests to the SIP node.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
orsnode_pending_tlib_requests Total number of pending TLib requests.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_sips_rest_connections The number of active REST connections with SIP Cluster Service.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_number_compiled_applications The number of compiled applications in the cache.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
orsnode_cached_applications_size The sum of the sizes of the cached applications.	**Unit:** **Type:** gauge **Label:** **Sample value:**	
orsnode_tlib_latency_msec The TLib Rest API request latency, measured in (ms).	**Unit:** milliseconds **Type:** histogram **Label:** le **Sample value:**	Latency
orsnode_application_size The compiled size of the Designer application.	**Unit:** **Type:** gauge **Label:** **Sample value:**	
orsnode_application_microstep_count	**Unit:** N/A	

Metric and description	Metric details	Indicator of
The number of microsteps while executing the Designer application.	**Type:** gauge **Label:** **Sample value:**	
orsnode_application_run_time_msec The length of time the Designer application was running, measured in milliseconds (ms).	**Unit:** milliseconds **Type:** gauge **Label:** **Sample value:**	
orsnode_application_compiled_date The date on which the Designer application was compiled.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
orsnode_application_last_invoked_date The date when the Designer application was last invoked.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	

Alerts

The following alerts are defined for ORS.

Alert	Severity	Description	Based on	Threshold
Number of running strategies is too high	Warning	Too many active sessions. Actions: • Check whether the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached. • Check the number of voice, digital, and callback calls in the system.	orsnode_strategies	More than 400 strategies running in 5 consecutive seconds.
Number of running strategies is	Critical	Too many active sessions.	orsnode_strategies	More than 600 strategies running

Alert	Severity	Description	Based on	Threshold
critical		Actions: • Check whether the horizontal pod autoscaler has triggered and if the maximum number of pods has been reached. • Check the number of voice, digital, and callback calls in the system.		in 5 consecutive seconds.
Redis disconnected for 5 minutes	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, and then restart Redis. • If alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.	redis_state	Redis is not available for the pod {{ $labels.pod }} for 5 minutes.
Redis disconnected for 10 minutes	Critical	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, and then restart Redis. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with	redis_state	Redis is not available for the pod {{ $labels.pod }} for 10 minutes.

Alert	Severity	Description	Based on	Threshold
		the pod.		
Pod status Failed	Warning	Pod {{ $labels.pod }} failed. 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 {{ $labels.pod }} is in Failed state.
Pod in Unknown state	Warning	Pod {{ $labels.pod }} is in Unknown state. 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 pod {{ $labels.pod }}, check 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.
Pod in Pending state	Warning	Pod {{ $labels.pod }} is in Pending state. Actions: • If the alarm is triggered for multiple services, make sure the Kubernetes nodes where the pod is	kube_pod_status_phase	Pod {{ $labels.pod }} is in Pending state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		running are alive in the cluster. • If the alarm is triggered only for the pod {{ $labels.pod }}, check the health of the pod.		
Pod Not ready for 10 minutes	Critical	Pod {{ $labels.pod }} in NotReady state. 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 }} in NotReady state for 10 minutes.
Container restored 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 the horizontal pod autoscaler has triggered and if the maximum number of	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
		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 pod autoscaler has triggered and if the maximum number of pods has been	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
		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.

Voice Registrar Service metrics and alerts

Find the metrics Voice Registrar Service exposes and the alerts defined for Voice Registrar Service.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Voice Registrar Service	Supports both CRD and annotations	11500	http://:11500/metrics	30 seconds

See details about:

- Voice Registrar Service metrics
- Voice Registrar Service alerts

Metrics

Voice Registrar Service exposes Genesys-defined, Registrar Service–specific metrics as well as some standard Kafka metrics. You can query Prometheus directly to see all the metrics that the Registrar Service exposes. The following metrics are likely to be particularly useful. Genesys does not commit to maintain other currently available Voice Registrar Service metrics not documented on this page.

Metric and description	Metric details	Indicator of
registrar_register_count Number of registrations.	**Unit:** N/A **Type:** counter **Label:** location, tenant **Sample value:**	Traffic
registrar_health_level Health level of the registrar node: -1 – fail 0 – starting 1 – degraded 2 – pass	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
registrar_request_latency Time taken to process the request (ms).	**Unit:** milliseconds **Type:** histogram **Label:** le, location, tenant **Sample value:**	Latency
registrar_active_sip_registrations Number of active SIP registrations.	**Unit:** N/A **Type:** gauge **Label:** tenant **Sample value:**	Traffic
kafka_consumer_latency Consumer latency is the time difference between when the message is produced and when the message is consumed. That is, the time when the consumer received the message minus the time when the	**Unit:** **Type:** histogram **Label:** tenant, topic **Sample value:**	Latency

Metric and description	Metric details	Indicator of
producer produced the message.		
kafka_consumer_state Current Kafka consumer connection state: 0 – disconnected 1 – connected	**Unit:** **Type:** gauge **Label:** **Sample value:**	

Alerts

The following alerts are defined for Voice Registrar Service.

Alert	Severity	Description	Based on	Threshold
Kafka events latency is too high	Warning	Actions: • If the alarm is triggered for multiple topics, make sure there are no issues with Kafka (CPU, memory, or network overload). • If the alarm is triggered only for topic {{ $labels.topic }}, check if there is an issue with the service related to the topic (CPU, memory, or network overload).	kafka_consumer_latency_bucket	Latency for more than 5% of messages is more than 0.5 seconds for topic {{ $labels.topic }}.
Too many Kafka consumer failed health checks	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with	kafka_consumer_error_total	Health check failed more than 10 times in 5 minutes for Kafka consumer for topic {{$labels.topic}}.

Alert	Severity	Description	Based on	Threshold
		Kafka, and then restart Kafka. • If the alarm is triggered only for {{ $labels.container }}, check if there is an issue with the service.		
Too many Kafka consumer request timeouts	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Kafka, and then restart Kafka. • If the alarm is triggered only for {{ $labels.container }}, check if there is an issue with the service.	kafka_consumer_error_total	There were more than 10 request timeouts within 5 minutes for the Kafka consumer for topic {{$labels.topic}}.
Too many Kafka consumer crashes	Critical	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Kafka, and then restart Kafka. • If the alarm is triggered only for {{ $labels.container }}, check if there is an issue with the service.	kafka_consumer_error_total	There were more than 3 Kafka consumer crashes within 5 minutes for service {{ $labels.container }}.

Alert	Severity	Description	Based on	Threshold
Kafka not available	Critical	Kafka is not available for pod {{ $labels.pod }}. Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Kafka, and then restart Kafka. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.	kafka_producer_state, kafka_consumer_state	Kafka is not available for pod {{ $labels.pod }} for 5 consecutive minutes.
Redis disconnected for 5 minutes	Warning	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, and then restart Redis. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.	redis_state	Redis is not available for pod {{ $labels.pod }} for 5 minutes.
Redis disconnected for 10 minutes	Critical	Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, and then restart Redis. • If the alarm is triggered only	redis_state	Redis is not available for pod {{ $labels.pod }} for 10 minutes.

Alert	Severity	Description	Based on	Threshold
		for pod {{ $labels.pod }}, check if there is an issue with the pod.		
Pod Failed	Warning	Pod {{ $labels.pod }} failed. 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 {{ $labels.pod }} is in Failed state.
Pod Unknown state	Warning	Pod {{ $labels.pod }} is in Unknown state. 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 pod {{ $labels.pod }}, check 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.
Pod Pending state	Warning	Pod {{ $labels.pod }} is in Pending state. Actions: • If the alarm is triggered for multiple services, make	kube_pod_status_phase	Pod {{ $labels.pod }} is in Pending state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		sure the Kubernetes nodes where the pod is running are alive in the cluster. • If the alarm is triggered only for pod {{ $labels.pod }}, check the health of the pod.		
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 the NotReady state for 10 minutes.
Container restarted repeatedly	Critical	Actions: • One of the container 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 CPU greater than 65%	Warning	High CPU load for pod {{ $labels.pod }}. Actions: • Check whether the horizontal pod autoscaler has triggered and if the maximum	container_cpu_usage_seconds_total, kube_pod_container_resource_limits	Container {{ $labels.container }} CPU usage exceeded 65% for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		number of pods has been reached. • Check Grafana for abnormal load. • Collect the service logs; raise an investigation ticket.		
Pod memory greater than 65%	Warning	High 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. • Collect the service logs; raise an investigation ticket.	container_memory_working_set_bytes kube_pod_container_resource_limits	Container {{ $labels.container }} memory usage exceeded 65% for 5 minutes.
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.	container_memory_working_set_bytes kube_pod_container_resource_limits	Container {{ $labels.container }} memory usage exceeded 80% for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		• Check Grafana for abnormal load. • Restart the service. • 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.	container_cpu_usage_seconds_total, kube_pod_container_resource_limits	Container {{ $labels.container }} CPU usage exceeded 80% for 5 minutes.

Voice RQ Service metrics and alerts

Find the metrics Voice RQ Service exposes and the alerts defined for Voice RQ Service.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Voice RQ Service	Supports both CRD and annotations	12000	http://:12000/metrics	30 seconds

See details about:

- Voice RQ Service metrics
- Voice RQ Service alerts

Metrics

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

Metric and description	Metric details	Indicator of
rqnode_clients Number of clients connected.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
rqnode_streams Number of active streams present.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
rqnode_xreads Number of XREAD requests received.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
rqnode_xadds Number of XADD requests received.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
rqnode_redis_state Current Redis connection state.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
rqnode_redis_disconnects The number of Redis disconnects that occurred for the RQ node.	**Unit:** **Type:** counter **Label:** **Sample value:**	Errors

Metric and description	Metric details	Indicator of
rqnode_consul_leader_error Number of errors received from Consul during the leadership process.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
rqnode_active_master Service master role is active.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
rqnode_active_backup Service backup role is active.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
rqnode_read_latency RQ latency; that is, the time duration between when an event is added to Redis and when it's read via XREAD.	**Unit:** **Type:** histogram **Label:** le, healthcheck **Sample value:**	Latency
rqnode_add_latency RQ latency; that is, the time duration between when a message is received and when it's added to the list.	**Unit:** **Type:** histogram **Label:** le, healthcheck **Sample value:**	Latency
rqnode_redis_latency Latency caused by Redis read/write.	**Unit:** **Type:** histogram **Label:** le **Sample value:**	Latency

Alerts

The following alerts are defined for Voice RQ Service.

Alert	Severity	Description	Based on	Threshold
Number of Redis streams is too high	Warning	Too many active sessions. Actions: • Check whether the horizontal pod autoscaler has triggered and if the maximum	rqnode_streams	More than 10000 active streams running.

Alert	Severity	Description	Based on	Threshold
		number of pods has reached. • Check the number of voice, digital, and callback calls in the system.		
Redis disconnected for 5 minutes	Warning	Redis is not available for the pod {{ $labels.pod }}. Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, restart Redis. • If the alarm is triggered only for the pod {{ $labels.pod }}, check to see if there is any issue with the pod.	redis_state	Redis is not available for the pod {{ $labels.pod }} for 5 minutes.
Redis disconnected for 10 minutes	Critical	Redis is not available for the pod {{ $labels.pod }}. Actions: • If the alarm is triggered for multiple services, make sure there are no issues with Redis, and then restart Redis. • If the alarm is triggered only for the pod {{	redis_state	Redis is not available for the pod {{ $labels.pod }} for 10 minutes.

Alert	Severity	Description	Based on	Threshold
		$labels.pod }}, check to see if there is any issue with the pod.		
Pod failed	Warning	Pod {{ $labels.pod }} failed. 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 {{ $labels.pod }} is in Failed state.
Pod Unknown state	Warning	Pod {{ $labels.pod }} in Unknown state. 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 whether the image is correct and if the container is starting up.	kube_pod_status_phase	Pod {{ $labels.pod }} in Unknown state for 5 minutes.
Pod Pending state	Warning	Pod {{ $labels.pod }} is in the Pending state. Actions: • If the alarm is triggered for multiple services, make	kube_pod_status_phase	Pod {{ $labels.pod }} is in the Pending state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		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.		
Pod not ready for 10 minutes	Critical	Pod {{ $labels.pod }} in NotReady state. 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 }} in NotReady state for 10 minutes.
Container restored repeatedly	Critical	Container {{ $labels.container }} was repeatedly restarted. 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 }}.	container_memory_working_set_bytes, kube_pod_container_resource_memory_bytes	Container {{ $labels.container }} memory usage exceeded 65% for

Alert	Severity	Description	Based on	Threshold
		Actions: • Check whether 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.		5 minutes.
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:	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
		• Check whether 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 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.

Voice SIP Cluster Service metrics and alerts

Find the metrics Voice SIP Cluster Service exposes and the alerts defined for Voice SIP Cluster Service.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Voice SIP Cluster Service	Supports both CRD and annotations	11300	http://:11300/metrics	30 seconds

See details about:

- Voice SIP Cluster Service metrics
- Voice SIP Cluster Service alerts

Metrics

Voice SIP Cluster Service exposes Genesys-defined, SIP Cluster Service–specific metrics as well as some standard Kafka metrics. You can query Prometheus directly to see all the metrics that the SIP Cluster Service exposes. The following metrics are likely to be particularly useful. Genesys does not commit to maintain other currently available SIP Cluster Service metrics not documented on this page.

Metric and description	Metric details	Indicator of
http_client_request_duration_seconds HTTP client time from request to response, measured in seconds.	**Unit:** seconds **Type:** histogram **Label:** le, target_service_name **Sample value:**	Latency
http_client_response_count Number of received HTTP client responses.	**Unit:** N/A **Type:** counter **Label:** target_service_name **Sample value:**	Traffic
kafka_producer_queue_depth Number of Kafka producer pending events.	**Unit:** N/A **Type:** gauge **Label:** kafka_location **Sample value:**	Traffic
kafka_producer_queue_age_seconds Age of the oldest producer pending event, measured in seconds.	**Unit:** seconds **Type:** gauge **Label:** kafka_location **Sample value:**	Traffic
kafka_producer_error_total Number of Kafka producer errors.	**Unit:** N/A **Type:** counter **Label:** kafka_location **Sample value:**	Errors
log_output_bytes_total	**Unit:** bytes	Traffic

Metric and description	Metric details	Indicator of
Total amount of log output in bytes.	**Type:** counter **Label:** level, format, module **Sample value:**	
sipnode_requests_total Number of processed requests.	**Unit:** N/A **Type:** counter **Label:** tenant, request **Sample value:**	Traffic
sipnode_pending_requests_current Number of pending requests.	**Unit:** N/A **Type:** gauge **Label:** tenant, request **Sample value:**	Traffic
sipnode_requests_queue_size Number of postponed requests.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
sipnode_sips_request_duration_seconds Duration of the request processed by SIP Cluster Service, measured in seconds.	**Unit:** seconds **Type:** histogram **Label:** le, tenant, request **Sample value:**	Traffic
sipnode_events_total Call events streamed to Redis.	**Unit:** N/A **Type:** counter **Label:** tenant, event **Sample value:**	Traffic
sipnode_ha_writes_total Number of HA writes to Redis.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
sipnode_ha_reads_total Number of HA reads from Redis.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
sipnode_monitoring_events_total Number of monitoring events submitted to Kafka.	**Unit:** N/A **Type:** counter **Label:** tenant **Sample value:**	Traffic
sipnode_redis_restored_calls_total Total number of restored calls from Redis cache.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
sipnode_sips_restarts_total	**Unit:** N/A	Errors

Metric and description	Metric details	Indicator of
Total number of SIP Server restarts.	**Type:** counter **Label:** **Sample value:**	
sipnode_sips_disconnects_total Total number of SIP Cluster Service disconnections from SIP Server.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
sipnode_redis_state Current Redis connection state.	**Unit:** N/A **Type:** gauge **Label:** redis_cluster_name **Sample value:**	Errors
sipnode_ors_tlib_latency_msec T-Library latency from Orchestration Service to SIP Cluster, measured in milliseconds.	**Unit:** milliseconds **Type:** histogram **Label:** le, ors **Sample value:**	Latency
sipnode_ors_health_check SIP Cluster Service to Orchestration Service health check.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
service_version_info Displays the version of Voice SIP Cluster 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	
sipnode_treatment_not_applied Number of unsuccessful treatments.	**Unit:** N/A **Type:** counter **Label:** tenant **Sample value:**	Errors
sipnode_default_routing_total Total number of default routed calls.	**Unit:** N/A **Type:** counter **Label:** tenant **Sample value:**	Traffic
sipnode_envoy_proxy_status Status of the Envoy proxy: -1 – error 0 – disconnected 1 – connected	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 1	Health
sipnode_config_node_status Status of the config node connection:	**Unit:** N/A **Type:** gauge **Label:**	Health

Metric and description	Metric details	Indicator of
0 – disconnected 1 – connected	**Sample value:** 1	
sipnode_health_level Health level of the SIP node (SIP Cluster Service): -1 – fail 0 – starting 1 – degraded 2 – pass	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 2	Traffic
sipnode_call_state_health_check SIP Cluster Service to Call State Service health check.	**Unit:** N/A **Type:** gauge **Label:** memberId **Sample value:**	Health
sips_hastate Current HA state of SIP Server: 0 – Unknown 1 – backup 2 – primary	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 2	
sips_calls Current number of calls.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_call_rate Call rate.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_cpu_usage_sips SIP Server CPU usage.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
sips_cpu_usage_main SIP Server main thread CPU usage.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
sips_cpu_usage_cm CPU usage of the call manager thread.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
sips_calls_created Total number of created calls.	**Unit:** N/A **Type:** gauge **Label:**	Traffic

Metric and description	Metric details	Indicator of
	Sample value:	
sips_abandoned_calls Total number of abandoned calls.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_rejected_calls Total number of rejected calls.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_dialogs_created Total number of created SIP dialogs.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_call_recording_failed Number of failed call recording sessions.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_urs_response_1_to_5_sec Number of URS responses from 1 to 5 seconds.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Latency
sips_urs_response_more_5_sec Number of URS responses more than 5 seconds.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Latency
sips_user_data_updates Number of UserData updates.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_routing_timeouts Number of routing timeouts.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_trequest_rate T-Requests rate.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_treatment_rate TApplyTreatment requests rate.	**Unit:** N/A **Type:** gauage **Label:**	Traffic

Metric and description	Metric details	Indicator of
	Sample value:	
sips_userdata_rate UserData change rate.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_sips_memory_usage Memory usage of the SIP Server process.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
sips_stat_fetch_total Number of successful SIP Server statistic fetches.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Other
sips_sip_response_time_ms SIP Server metric of response time, measured in milliseconds.	**Unit:** milliseconds **Type:** histogram **Label:** le **Sample value:**	Latency
sips_trunk_in_service Trunk devices that are in service.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Traffic
sips_trunk_ncallscreated Number of created calls per trunk.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Traffic
sips_trunk_noos_detected Number of trunks that are out of service.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Errors
sips_trunk_n4xx_received Number of received 4xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Errors
sips_trunk_n5xx_received Number of received 5xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Errors
sips_trunk_n6xx_received Number of received 6xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant	Errors

Metric and description	Metric details	Indicator of
	Sample value:	
sips_softswitch_in_service Softswitch devices that are in service.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Traffic
sips_softswitch_ncallscreated Number of created calls per softswitch device.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Traffic
sips_softswitch_noos_detected Number of softswitch devices that are out of service.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Errors
sips_softswitch_n4xx_received Number of received 4xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Errors
sips_softswitch_n5xx_received Number of received 5xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Errors
sips_softswitch_n6xx_received Number of received 6xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name, tenant **Sample value:**	Errors
sips_msml_in_service MSML devices that are in service.	**Unit:** N/A **Type:** gauge **Label:** device_name **Sample value:**	Traffic
sips_msml_ncallscreated Number of created calls per MSML device.	**Unit:** N/A **Type:** gauge **Label:** device_name **Sample value:**	Traffic
sips_msml_noos_detected Number of MSML devices that are out of service.	**Unit:** N/A **Type:** gauge **Label:** device_name **Sample value:**	Errors
sips_msml_n4xx_received Number of received 4xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name	Errors

Metric and description	Metric details	Indicator of
	Sample value:	
sips_msml_n5xx_received Number of received 5xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name **Sample value:**	Errors
sips_msml_n6xx_received Number of received 6xx messages.	**Unit:** N/A **Type:** gauge **Label:** device_name **Sample value:**	Errors
sips_dp_state Dial Plan Service state: 0 – Out-Of-Service 1 – In-Service	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 1	Traffic
sips_dp_queue_size Size of the request queue to Dial Plan Service.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_dp_avg_queue_time Average queue time (msec) of requests to Dial Plan Service.	**Unit:** milliseconds **Type:** gauge **Label:** **Sample value:**	Latency
sips_dp_connections Number of connections to Dial Plan Service per URL.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_dp_active_connections Number of active connections to Dial Plan Service.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_dp_req_rate Request rate to Dial plan Service.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Traffic
sips_dp_400_errors Dial Plan Service 400 type of errors.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_dp_404_errors Dial Plan Service 404 type of errors.	**Unit:** N/A **Type:** gauge	Errors

Metric and description	Metric details	Indicator of
	Label: **Sample value:**	
sips_dp_4xx_errors Dial Plan Service 4xx type of errors.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_dp_500_errors Dial Plan Service 500 type of errors.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_dp_501_errors Dial Plan Service 501 type of errors.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_dp_5xx_errors Dial Plan Service 5xx type of errors.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_dp_timeouts Dial Plan Service timeouts.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Errors
sips_dp_average_response_latency Dial Plan Service average response latency.	**Unit:** **Type:** gauge **Label:** **Sample value:**	Latency
sips_sipproxy_in_service SIP Proxy Service state: 0 – Out-Of-Service 1 – In-Service	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 1	Traffic
trunk_config_synced_count Number of trunks synchronized with SIP Server.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
trunk_config_cached_count Number of trunks obtained from the config node.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
trunk_config_cfg_node_error_count	**Unit:** N/A	

Metric and description	Metric details	Indicator of
Number of failed attempts to read from the config node.	**Type:** counter **Label:** **Sample value:**	
trunk_config_tlib_connection Number of trunks with the T-Library connection.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	

Alerts

The following alerts are defined for Voice SIP Cluster Service.

Alert	Severity	Description	Based on	Threshold
Too many Kafka pending events	Critical	Too many Kafka producer pending events for pod {{ $labels.pod }}. Actions: • Ensure there are no issues with Kafka, {{ $labels.pod }} pod's CPU, and network.	kafka_producer_queue_depth	Too many Kafka producer pending events for service {{ $labels.container }} (more than 100 in 5 minutes).
Dial Plan node is overloaded	Critical	Dial Plan node is overloaded as the response latency increases. Actions: • Check that the inbound call rate to SIP Server is not too high. • Check the Dial Plan node CPU and memory usage. • Check the network	sips_dp_average_response_latency	Dial Plan node is overloaded as the response latency increases (more than 1000).

Alert	Severity	Description	Based on	Threshold
		connection between SIP Server and Dial Plan nodes.		
Dial Plan Queue Increase	Critical	Because Dial Plan requests are huge in size or there is a connection issue with the Dial Plan node, the processing queue size increases in size. Actions: • Check SIP Server inbound call rate. • Check the connection between SIP Server and the Dial Plan node.	sips_dp_queue_size	The processing queue size is greater than 10 requests for 1 minute.
SIP Proxy overloaded	Critical	SIP Proxy is overloaded. Actions: • Check SIP Proxy nodes for CPU and memory usage. • If SIP Proxy nodes have acceptable CPU and memory usage, then check for errors or a "hang-up" state which could delay SIP Proxy in forwarding. • Check the SBC side for network delays.	sips_sip_response_time_ms_sum, sips_sip_response_time_ms_count	Response time is greater than 20 milliseconds for 1 minute
SIP Node	Critical	SIP Node health	sipnode_health_level	SIP Node health

Alert	Severity	Description	Based on	Threshold
HealthCheck Fail		level fails for pod {{ $labels.pod }}. Actions: • Check for failure of dependent services (Redis/ Kafka/SIP Proxy/GVP/Dial Plan). • Check for Envoy proxy failure, then restart the pod.		level fails for pod {{ $labels.pod }} for 5 minutes.
Kafka not available	Critical	Kafka is not available for pod {{ $labels.pod }}. Actions: • If the alarm is triggered for multiple services, ensure there are no issues with Kafka. Restart Kafka. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.	kafka_producer_state	Kafka is not available for pod {{ $labels.pod }} for 5 minutes.
Pod Status Error	Warning	Actions: • Restart the pod. Check if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Failed, Unknown, or Pending state.
Pod Status NotReady	Warning	Pod {{ $labels.pod }} is in NotReady state. Actions:	kube_pod_status_ready	Pod {{ $labels.pod }} is in NotReady state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		• Restart the pod. Check if there are any issues with the pod after restart.		
Container Restarted Repeatedly	Critical	Container {{ $labels.container }} was repeatedly restarted. Actions: • Check if the new version of the image was deployed. • Check for issues with the Kubernetes cluster.	kube_pod_container_status_restarts_total	Container {{ $labels.container }} was restarted 5 or more times within 15 minutes.
Ready Pods below 60%	Critical	The number of statefulset {{ $labels.statefulset}} pods in the Ready state has dropped below 60%. Actions: • Check if the new version of the image was deployed. • Check for issues with the Kubernetes cluster.	kube_statefulset_status_replicas_ready, kube_statefulset_status_replicas_current	For the last 5 minutes, fewer than 60% of the currently available statefulset {{ $labels.statefulset}} pods have been in the Ready state.
Pods scaled up greater than 80%	Critical	The current number of replicas is more than 80% of the maximum number of replicas. Actions: • Check if max replicas must be modified	kube_hpa_status_current_replicas, kube_hpa_spec_max_replicas	For 5 consecutive minutes, the number of replicas is more than 80% of the maximum number of replicas.

Alert	Severity	Description	Based on	Threshold
		based on load.		
Pods less than Min Replicas	Critical	The current number of replicas is less than the minimum replicas that should be available. This might be because Kubernetes cannot deploy a new pod or pods are failing to be active/ready. Actions: • If all services have the same issue, then check Kubernetes nodes and Consul health. • If the issue is only with the SIP Cluster service, then check pod logs or the deployment manifest/helm errors.	kube_hpa_status_current_replicas kube_hpa_spec_min_replicas	For 5 consecutive minutes, the number of replicas is less than the minimum replicas that should be available.
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. • Collect the	container_cpu_usage_seconds_total container_spec_cpu_period	Container {{ $labels.container }} CPU usage exceeded 80% for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		service logs for pod {{ $labels.pod }}; raise an investigation ticket.		
Pod CPU greater than 65%	Warning	High 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. • Collect the service logs for pod {{ $labels.pod }}; raise an investigation ticket.	container_cpu_usage_seconds_total, container_spec_cpu_period	Container {{ $labels.container }} CPU usage exceeded 65% for 5 minutes.
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.	container_memory_working_set_bytes, kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 80% for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		• Restart the service for pod {{ $labels.pod }}.		
Pod memory greater than 65%	Warning	High 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. • Collect the service logs for pod {{ $labels.pod }}; raise an investigation ticket.	container_memory_working_set_bytes kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 65% for 5 minutes.
Redis not available	Critical	Redis is not available for pod {{ $labels.pod }}. Actions: • If the alarm is triggered for multiple services, ensure there are no issues with Redis. Restart Redis. • If the alarm is triggered only for pod {{ $labels.pod }}, check if there is an issue with the pod.	redis_state	Redis is not available for pod {{ $labels.pod }} for 5 consecutive minutes.

Alert	Severity	Description	Based on	Threshold
Too many Kafka producer errors	Critical	Kafka responds with errors at pod {{ $labels.pod }}. Actions: • For pod {{ $labels.pod }}, ensure there are no issues with Kafka.	kafka_producer_error_total	More than 100 errors for 5 consecutive minutes.
SIP Server main thread consuming more than 65% CPU for 5 mins	Warning	Main thread consumes too much CPU. Actions: • Collect SIP Server Main thread logs; that is, log files without index in the file name (appname_date.log files). Raise an investigation ticket.	sips_cpu_usage_main	Main thread consumes too much CPU (more than 65% for 5 consecutive minutes).
Calls activity drop	Warning	A noticeable reduction in the number of active calls on a specific SIP Server and no new calls are arriving for processing. Actions: • If a problematic SIP Server is primary, do a switchover, and then restart the former primary server. • If a problematic SIP Server is backup, restart the backup server. Collect SIP Server Main thread logs;	sips_calls, sips_calls_created	The absolute value of active calls on a specific SIP Server dropped by more than 30 calls in 2 minutes and no new calls are arriving at the SIP Server for processing.

Alert	Severity	Description	Based on	Threshold
		that is, log files without index in the file name (appname_date.log files). Raise an investigation ticket.		
Dial Plan Node Down	Critical	No Dial Plan nodes are reachable from SIP Server and all connections to Dial Plan nodes are down. Actions: • Check the network connection between SIP Server and the Dial Plan node host. • Check the Dial Plan node CPU and memory usage.	sips_dp_active_connections	All connections to Dial Plan nodes are down.
Dialplan Node problem	Warning	Dial Plan node rejects requests with an error or it doesn't respond to requests and requests are timed out. Actions: • Check the network connection between SIP Server and the Dial Plan host. • Check that Dial Plan nodes are running.	sips_dp_timeouts	During 1 minute, the Dial Plan node rejects more than 5 requests with an error or more than 5 requests time out because the Dial Plan node fails to respond.
Routing timeout counter growth	Warning	The trigger detects that routing timeouts are increasing.	sips_routing_timeouts	The absolute value of NROUTINGTIMEOUTS on a specific SIP

Alert	Severity	Description	Based on	Threshold
		Actions: • Check the URS_RESPONSE_MORE5SEC stat value. If it's increasing, then investigate why URS doesn't respond to SIP Server in time. • Check SIPS-to-URS network connectivity.		Server increased by more than 20 in 2 minutes.

SIP trunk is out of service	Critical	SIP trunk is out of service.
		Actions:
		• For Primary and Secondary trunks:
		  • Troubleshoot SIP Server-to-SBC network connection. Collect network stats and escalate to the Network team to resolve network issues, if necessary.
		  • Troubleshoot the SBC.
		For Inter-SIP Server trunks: troubleshoot the SIP Server-to-SIP Server network connection. Collect
			sips_trunk_in_service	SIP trunk is out of service for more than 1 minute.

Alert	Severity	Description	Based on	Threshold
		network stats and escalate to the Network team to resolve network issues, if necessary.		
Media service is out of service	Critical	Media service is out of service. Actions: • Troubleshoot the SIP Server-to-Resource Manager (RM) network connection. Collect network stats and escalate to the Network team to resolve network issues, if necessary. • Troubleshoot RM, consider RM restart. • After 5 minutes, redirect traffic to another site.	sips_msml_in_service	Media service is out of service for more than 1 minute.
SIP softswitch is out of service	Critical	Actions: • Troubleshoot the SIP Server-to-SBC network connection. Collect network stats and escalate to the Network team to resolve network issues, if necessary. • Troubleshoot the SBC.	sips_softswitch_in_service	SIP softswitch is out of service.

Alert	Severity	Description	Based on	Threshold
SIP Proxy is out of service	Critical	Actions: • Troubleshoot the SIP Server-to-SIP Proxy nodes network connections. Collect network stats and escalate to the Network team to resolve network issues, if necessary. • Troubleshoot SIP Proxy nodes.	sips_sipproxy_in_service	SIP Proxy is out of service.

Voice SIP Proxy Service metrics and alerts

Find the metrics Voice SIP Proxy Service exposes and the alerts defined for Voice SIP Proxy Service.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Voice SIP Proxy Service	Supports both CRD and annotations	11400	http://:11400/metrics	30 seconds

See details about:

- Voice SIP Proxy Service metrics
- Voice SIP Proxy Service alerts

Metrics

Voice SIP Proxy Service exposes Genesys-defined, SIP Proxy Service–specific metrics as well as some standard Kafka metrics. You can query Prometheus directly to see all the metrics that the SIP Proxy Service exposes. The following metrics are likely to be particularly useful. Genesys does not commit to maintain other currently available SIP Proxy Service metrics not documented on this page.

Metric and description	Metric details	Indicator of
sipproxy_requests_total Total number of received requests.	**Unit:** N/A **Type:** counter **Label:** method **Sample value:**	Traffic
sipproxy_rejected_requests_total The total number of rejected requests.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
sipproxy_requests_processed_self_total The total number of received requests that were processed by SIP Proxy itself.	**Unit:** N/A **Type:** counter **Label:** method **Sample value:**	Traffic
sipproxy_requests_forwarded_total The total number of forwarded requests.	**Unit:** N/A **Type:** counter **Label:** method, request_direction, sip_node_id **Sample value:**	Traffic
sipproxy_requests_sip_node_reselected_total Total count of sip-node reselection.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
sipproxy_responses_forwarded_total	**Unit:** N/A	Traffic

Metric and description	Metric details	Indicator of
Total count of forwarded responses.	**Type:** counter **Label:** method, sip_node_id, request_direction **Sample value:**	
sipproxy_response_latency SIP response latency.	**Unit:** **Type:** histogram **Label:** le, sip_node_id, request_direction, target, node_in_cache **Sample value:**	Latency
sipproxy_register_processed_total Total number of REGISTER requests that SIP Proxy received for processing.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
sipproxy_register_rejected_total Total number of REGISTER requests for processing that were rejected.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
sipproxy_calls_per_second_count Current calculated calls per second.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
sipproxy_active_sip_nodes_count Current number of active SIP nodes.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
sipproxy_sip_nodes_count Current number of discovered SIP nodes.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
sipproxy_tenants_count Current count of discovered tenants.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
sipproxy_consul_record_processing_errors_count Current number of errors while processing records got from Consul.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
sipproxy_consul_errors_count Current number of Consul errors.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
sipproxy_sip_node_is_capacity_available	**Unit:**	

Metric and description	Metric details	Indicator of
Indicates whether SIP node has available capacity or not.	**Type:** gauge **Label:** sip_node_id **Sample value:**	
service_version_info Displays the version of Voice SIP Proxy 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	
sipproxy_health_level Health level of the SIP Proxy node: -1 – fail 0 – starting 1 – degraded 2 – pass	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	
sipproxy_envoy_proxy_status Status of the Envoy proxy: -1 – error 0 – disconnected 1 – connected	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 1	
sipproxy_config_node_status Status of the Config node connection: 0 – disconnected 1 – connected	**Unit:** N/A **Type:** gauge **Label:** **Sample value:** 1	
sip_server_transactions_created_total Total number of created server transactions.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
sip_client_transactions_created_total Total number of created client transactions.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
sip_server_transactions_deleted_total Total number of deleted server transactions.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
sip_client_transactions_deleted_total Total number of deleted client transactions.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
sip_client_transactions_count	**Unit:** N/A	Saturation

Metric and description	Metric details	Indicator of
Current number of client transactions.	**Type:** gauge **Label:** **Sample value:**	
sip_server_transactions_count Current number of server transactions.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
sip_server_transactions_rejected_total Total number of server transactions rejected for internal reasons.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
sip_proxy_contexts_count Current number of active SIP Proxy forwarding contexts.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Saturation
sip_received_bytes_total Total traffic received, measured in bytes.	**Unit:** bytes **Type:** counter **Label:** transport **Sample value:**	Traffic
sip_sent_bytes_total Total traffic sent, measured in bytes.	**Unit:** bytes **Type:** counter **Label:** transport **Sample value:**	Traffic
sip_transport_errors_total Total number of transport errors.	**Unit:** N/A **Type:** counter **Label:** transport, address **Sample value:**	Errors
sip_stream_transport_wait_drain_total Total number of requests to wait for drain events on stream transports.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
sip_stream_transport_flood_total Total number of flood events on the stream transports.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	
http_client_request_duration_seconds The time duration between the HTTP client request and the response, measured in seconds.	**Unit:** seconds **Type:** histogram **Label:** le, target_service_name **Sample value:**	Latency
http_client_response_count	**Unit:** N/A	Traffic

Metric and description	Metric details	Indicator of
The number of HTTP client responses received.	**Type:** counter **Label:** target_service_name, status **Sample value:**	

log_output_bytes_total
The total amount of log output, measured in bytes.
	Unit: bytes
	Type: counter
	Label: level, format, module
	Sample value:
	`log_output_bytes_total{level="info",format="txt",module="sipproxy_node@config-manager"} 3175`
	`log_output_bytes_total{level="info",format="txt",module="sipproxy_node@sipproxy-node"} 96`
	`log_output_bytes_total{level="info",format="txt",module="sipproxy_node@sipproxy@sip"} 181`
	`log_output_bytes_total{level="info",format="json",module="sipproxy_node@config-manager"} 4184`
	`log_output_bytes_total{level="info",format="json",module="sipproxy_node@sipproxy-node"} 135`
	`log_output_bytes_total{level="info",format="json",module="sipproxy_node@sipproxy@sip"} 259`	

kafka_consumer_recv_messages_total Number of messages received from Kafka.	**Unit:** **Type:** counter **Label:** **Sample value:**	Traffic
kafka_consumer_error_total Number of Kafka consumer errors.	**Unit:** **Type:** counter **Label:** **Sample value:**	Errors
kafka_consumer_latency Consumer latency is the time difference between when the message is produced and when the message is consumed. That is, the time when the consumer received the message minus the time when the producer produced the message.	**Unit:** **Type:** histogram **Label:** **Sample value:**	Latency
kafka_consumer_rebalance_total Number of Kafka consumer rebalance events.	**Unit:** **Type:** counter **Label:** **Sample value:**	
kafka_consumer_state Current state of the Kafka consumer.	**Unit:** **Type:** gauge **Label:** **Sample value:**	
kafka_producer__messages_total Number of messages received from Kafka.	**Unit:** **Type:** counter **Label:** **Sample value:**	Traffic
kafka_producer_queue_depth Number of Kafka producer pending	**Unit:** **Type:** gauge	Saturation

Metric and description	Metric details	Indicator of
events.	**Label:** kafka_location **Sample value:**	
kafka_producer_queue_age_seconds Age of the oldest producer pending event in seconds.	**Unit:** seconds **Type:** gauge **Label:** kafka_location **Sample value:**	
kafka_producer_error_total Number of Kafka producer errors.	**Unit:** **Type:** counter **Label:** kafka_location **Sample value:**	Errors
kafka_producer_state Current state of the Kafka producer.	**Unit:** **Type:** gauge **Label:** kafka_location **Sample value:**	
kafka_producer_biggest_event_size Biggest event size so far.	**Unit:** **Type:** gauge **Label:** kafka_location, topic **Sample value:** 231	
kafka_max_request_size Exposed config to compare with biggest event size.	**Unit:** **Type:** gauge **Label:** kafka_location **Sample value:** 1000000	
kafka_producer_dropped_event_number Number of dropped events.	**Unit:** **Type:** gauge **Label:** **Sample value:**	

Alerts

The following alerts are defined for Voice SIP Proxy Service.

Alert	Severity	Description	Based on	Threshold
Too many Kafka pending events	Critical	Too many Kafka producer pending events for pod {{ $labels.pod }}. This alert means there are issues with SIP REGISTER processing on this	kafka_producer_queue_depth	Too many Kafka producer pending events for service {{ $labels.container }} (more than 100 in 5 minutes).

Alert	Severity	Description	Based on	Threshold
		voice-sipproxy. Actions: • Make sure there are no issues with Kafka or with the {{ $labels.pod }} pod's CPU and network.		
SIP server response time too high	Warning	Actions: • If the alarm is triggered for multiple sipproxy-nodes, make sure there are no issues on {{ $labels.sip_node_id }}. • If the alarm is triggered only for sipproxy-node {{ $labels.pod }}, check to see if there is an issue with the service related to the topic (CPU, memory, or network overload).	sipproxy_response_latency_bucket	SIP response latency for more than 95% of messages forwarded to {{ $labels.sip_node_id }} is more than 1 second for sipproxy-node {{ $labels.pod }}.
Pod status failed	Warning	Actions: • Restart the pod and check to see if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Failed state.
Pod status Unknown	Warning	Pod {{ $labels.pod }} is in Unknown state. Actions: • Restart the pod	kube_pod_status_phase	Pod {{ $labels.pod }} is in Unknown state for 5 minutes.

Alert	Severity	Description	Based on	Threshold
		and check to see if there are any issues with the pod after restart.		
Pod status Pending	Warning	Pod {{ $labels.pod }} is in Pending state. Actions: • Restart the pod and check to see if there are any issues with the pod after restart.	kube_pod_status_phase	Pod {{ $labels.pod }} is in Pending state for 5 minutes.
Pod status NotReady	Critical	Pod {{ $labels.pod }} is in NotReady state. Actions: • Restart the pod and check to see if there are any issues with the pod after restart.	kube_pod_status_ready	Pod {{ $labels.pod }} is in NotReady state for 5 minutes.
Container restarted repeatedly	Critical	Container {{ $labels.container }} was repeatedly restarted. Actions: • Check to see if a new version of the image was deployed. Also check for issues with the Kubernetes cluster.	kube_pod_container_status_restarts_total	Container {{ $labels.container }} was restarted 5 or more times within 15 minutes.
No sip-nodes available for 2 minutes	Critical	No sip-nodes are available for the pod {{ $labels.pod }}. Actions:	sipproxy_active_sip_nodes_count	No sip-nodes are available for the pod {{ $labels.pod }} for 2 minutes.

Alert	Severity	Description	Based on	Threshold
		• If the alarm is triggered for multiple services, make sure there are no issues with sip-nodes. • If the alarm is triggered only for pod {{ $labels.pod }}, check to see if there is any issues with the pod.		
sip-node capacity limit reached	Warning	The sip-node {{ $labels.sip_node_id }} hit capacity limit on {{ $labels.pod }}. Actions: • If alarm is triggered for multiple services make sure there is no issues with sip-node {{ $labels.sip_node_id }}. • If alarm is triggered only for pod {{ $labels.pod }} check if there is any issue with the pod	sipproxy_sip_node_is_capacity_available	The sip-node {{ $labels.sip_node_id }} hit capacity limit on {{ $labels.pod }} for 3 consecutive minutes.
Pod CPU greater than 80%	Critical	Critical CPU load for pod {{ $labels.pod }}. Actions: • Check whether the horizontal pod autoscaler has triggered and the maximum	container_cpu_usage_seconds_total, container_spec_cpu_period	Container {{ $labels.container }} CPU usage exceeded 80% for 5 minutes.

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

Alert	Severity	Description	Based on	Threshold
		maximum number of pods has been reached. • Check Grafana for abnormal load. • Restart the service for pod {{ $labels.pod }}.		
Pod memory greater than 65%	Warning	Pod {{ $labels.pod }} has high memory usage. Actions: • Check whether the horizontal pod autoscaler has triggered and the maximum number of pods has been reached. • Check Grafana for abnormal load. • Collect the service logs for pod {{ $labels.pod }} and raise an investigation ticket	container_memory_working_set_bytes kube_pod_container_resource_requests_memory_bytes	Container {{ $labels.container }} memory usage exceeded 65% for 5 minutes.
Config node fail	Warning	The request to the config node failed. Action: • Check if there is any problem with pod {{ $labels.pod }} and config node.	http_client_response_count	Requests to the config node fail for 5 consecutive minutes.

Voicemail metrics and alerts

Find the metrics Voicemail exposes and the alerts defined for Voicemail.

Contents

Service	CRD or annotations?	Port	Endpoint/Selector	Metrics update interval
Voicemail	Supports both CRD and annotations	8081	http://:8081/metrics	30 seconds

See details about:

- Voicemail metrics
- Voicemail alerts

Metrics

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

Metric and description	Metric details	Indicator of
voicemail_access_call_rate The voicemail access call rate.	**Unit:** **Type:** gauge **Label:** **Sample value:**	Traffic
voicemail_deposit_call_rate The voicemail deposit call rate.	**Unit:** **Type:** gauge **Label:** **Sample value:**	Traffic
voicemail_gws_request_total The total number of requests sent to GWS.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
voicemail_redis_request_total The total number of requests sent to Redis.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
voicemail_config_request_total The total number of requests sent to the Config node.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Traffic
voicemail_config_request_failed_total The total number of requests sent to the config node that failed.	**Unit:** N/A **Type:** counter **Label:** response code; for example, `Internal Server Error` or `Service`	Errors

Metric and description	Metric details	Indicator of
	`Unavailable` **Sample value:**	
voicemail_redis_request_failed_total The total number of Message Waiting Indicator (MWI) notification requests sent to the Redis stream that failed.	**Unit:** N/A **Type:** counter **Label:** **Sample value:**	Errors
voicemail_gws_request_failed_total The total number of authentication errors when the Voicemail API is accessing via GWS SSO.	**Unit:** N/A **Type:** counter **Label:** response code; for example, `Internal Server Error` or `Service Unavailable` **Sample value:**	Errors
voicemail_service_health_check Status of the service health check: 2 – The service health check is alive 1 – The service health check is degraded 0 – Initializing -1 – The service health check has failed The service health check takes the status of all the dependencies into consideration. The overall Voicemail Service health is updated every two minutes.	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Aggregated health check
voicemail_envoy_proxy_status The status of the Envoy proxy: 1 – The Envoy proxy is alive 0 – The Envoy proxy is down	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Aggregated health check
voicemail_gws_status The status of GWS: 1 – GWS is alive 0 – GWS is down	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Aggregated health check
voicemail_config_node_status Config node status: 1 – the Config node is alive 0 – The Config node is down	**Unit:** N/A **Type:** gauge **Label:** **Sample value:**	Aggregated health check
voicemail_redis_state Indicator of redis_state: 2 – redis_state is ready 1 – redis_state is degraded 0 – redis_state is failed	**Unit:** **Type:** gauge **Label:** **Sample value:**	Aggregated health check

Alerts

The following alerts are defined for Voicemail.

Alert	Severity	Description	Based on	Threshold
voicemail_storage_failed_account	Critical	The Storage account is down and, as a result, the service will not be able to fetch the data.	voicemail_storage_failed_account	The Storage account is down.
VoicemailConfigRequestFailure	Critical	Voicemail Service {{$labels.pod}} unable to connect to Config Node.	voicemail_config_request_failed_total	At least 6 requests failed per minute for the past 10 minutes.
VoicemailRedisConnectionFail	Critical	Voicemail Service {{$labels.pod}} unable to connect to the Redis cluster.	voicemail_redis_connection_failure	At least 6 requests failed per minute for the past 10 minutes.
voicemail_node_memory_usage_80	Critical	Critical memory usage for pod {{ $labels.pod }}.	container_memory_working_set_bytes, kube_pod_container_resource_requests_memory_bytes	The Voicemail pod exceeded 80% memory usage for 5 minutes.
voicemail_node_cpu_usage_80	Critical	Critical CPU load for pod {{ $labels.pod }}.	container_cpu_usage_seconds_total, kube_pod_container_resource_requests_cpu_cores	The Voicemail pod exceeded 80% CPU usage for 5 minutes.
PodStatusNotReadyforCritical	Critical	The Voicemail pod is down.	kube_pod_status_ready	The Voicemail pod is down for more than 10 minutes.
ContainerRestartedRepeatedly	Critical	The Voicemail pod restarts repeatedly.	kube_pod_container_status_restarts_total	Container {{ $labels.container }} was restarted 5 or more times within 15 minutes.
VoicemailEnvoyHealthFailed	Critical	Voicemail Service {{$labels.pod}} Envoy service is	voicemail_envoy_proxy_state	Voicemail Service {{$labels.pod}} Envoy service is

Alert	Severity	Description	Based on	Threshold
		not available.		not available for 10 minutes.
VoicemailConfigHealthFailed	Critical	Voicemail Service {{$labels.pod}} GWS service is not available.	voicemail_config_node_status	Voicemail Service {{$labels.pod}} GWS service is not available for 10 minutes.
VoicemailGWSHealthFailed	Critical	Voicemail Service {{$labels.pod}} GWS service is not available.	voicemail_gws_status	Voicemail Service {{$labels.pod}} GWS service is not available for 15 minutes.