Agent Pacing Service Deployment Guide

Architecture

2/24/2026

Contents

Read about performance and high availability considerations for Agent Pacing Service.

Architecture diagram
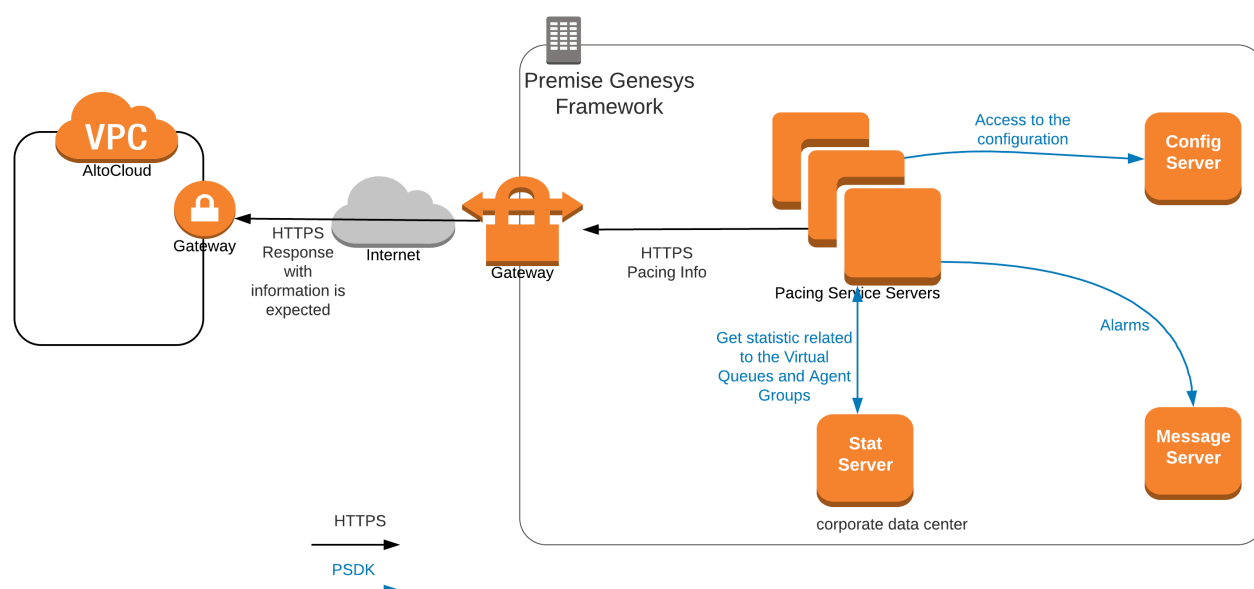
The Agent Pacing Service connects to only one Stat Server (high-availability pair).

Important

The Agent Pacing Service requires that the Stat Server has visibility of all potential pacing targets.

Multiple actively running pacing nodes can belong to the same pacing cluster. Under normal conditions, there is only one master pacing node that provides pacing target information to Genesys Predictive Engagement. All pacing nodes are interconnected, and they self-determine which is the master pacing node based on an internal election algorithm.

Performance

Each pacing node is designed to support up to 10,000 pacing targets. The execution of the pacing algorithm does not put significant pressure on either the memory or the CPU. However, as the number of pacing targets increases, so does the size of the HTTP request and the corresponding response between Pacing Service and Genesys Predictive Engagement. For this reason, network bandwidth becomes a concern in terms of system stability.

Problems may occur if the roundtrip of the HTTP package reaches the timeout limit or the pacing refresh period. If you experience performance problems, try tuning your network bandwidth or reducing the number of pacing targets.

Important

The performance of a single pacing server node has been verified for the following configuration:

- 2 CPUs (Core i7)
- 6 Gb RAM
- 2 Gb of allocated Java Heap memory

Scalability

For nodes belonging to same pacing cluster, the scalability model is N+1. Based on performance testing, two pacing nodes should be sufficient to sustain most deployments.

High availability

This release of Agent Pacing Service works with a single Stat Server HA pair. To ensure high availability in a premises-based environment, install two (2) Pacing Server nodes.