CONTROL High Availability Architecture

Overview

To achieve a robust high-availability (HA) architecture for CONTROL Device Management, Service Providers must comprehensively address both application implementation and operational requirements within their network infrastructure. This includes identifying critical points of failure and implementing appropriate HA strategies to mitigate them.

High-Availability Architecture Overview

The primary objective of an HA architecture is to guarantee uninterrupted application service in the event of component failure within the solution.

The standard CONTROL Device Management application architecture is illustrated below:

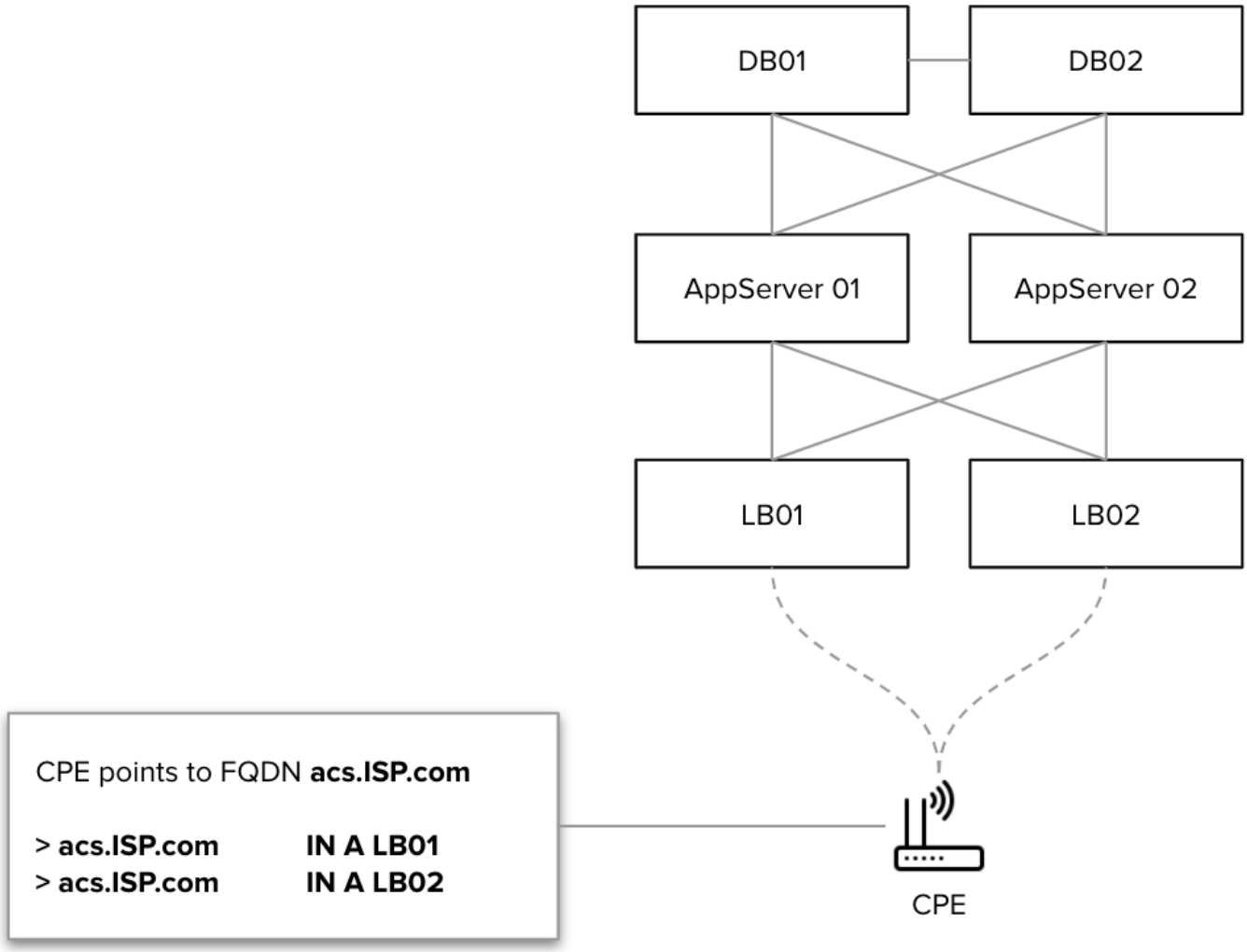

Figure 1. Device Management application architecture

Architecture Prerequisites

The CONTROL HA architecture is designed with the following key requirements:

- **Active-Active Configuration**: Ensures predictable HA operation and optimal resource utilization
- **Cluster Capacity**: Each individual cluster must be capable of handling 100% of the **average** load
- **Deployment Flexibility**: HA clusters can be deployed either co-located or geographically distributed (typically requires site-to-site latency below 250 ms)

Principles of Operation

The HA architecture operates based on the following mechanisms:

- **DNS Resolution**: Customer Premises Equipment (CPE) points to a Fully Qualified Domain Name (FQDN) that resolves to the load balancer array
- **DNS Health Checks**: DNS periodically verifies load balancer array availability using HTTP health checks
- **Load Balancer Health Checks**: Load balancers periodically verify Auto Configuration Server (ACS) service availability through HTTP/API methods

Failure Scenarios

The following failure scenarios describe system behavior in a geographically-distributed HA deployment:

Complete Site or Load Balancer Failure

When DNS detects a failure, it automatically removes the failing load balancer from the DNS group, redirecting traffic to healthy sites.

Application Server Failure

When the load balancer detects an application server failure, it automatically removes the failing application server from the load balancer group, distributing traffic among remaining healthy servers.

Database Failure

In the event of a database failure, both application servers automatically fail over to the surviving database instance, ensuring continuous operation.

Revision #2
Created 2026-02-13 22:24:56 UTC by ipena@zequenze.com
Updated 2026-02-14 01:08:16 UTC by ipena@zequenze.com