



BESTARION

CONTROLLED DOCUMENT

CONVENTION

QM

Security classification: INTERNAL

Document code: Bestarion.QM.CON.001

Last updated by: Khoa Nguyen-Dinh

Effective date: Aug 09, 2023

Version: 2.1

Template ID: ODT\Base\Template

DOCUMENT CONTROL

Version	Change Description	Changed By	Date	Approved By	Date
1.0	Khởi tạo tài liệu và phê duyệt lần đầu.	KhoaND	2025-01-01	CEO/TienDD	2025-01-05
1.1	Cập nhật phần quy trình 3.2.	MinhNT	2025-02-10	N/A	N/A

TABLE OF CONTENT

1	Introduction	5
1.1	Purpose	5
1.2	Scope	5
1.3	Definition And Abbreviation	5
2	Architecture	5
2.1	Flow	5
2.2	Functional	6
3	Requirements	6
3.1	Module & Codes	6
3.2	Version & Provider	6

INDEX OF TABLES

1 Terms	5
-----------------------	---

INDEX OF FIGURES

1. Introduction

1.1. Purpose

This document explains Terraform module frontend in MAX project, documenting the CloudFront distribution deployment for the PERFORMANCE environment. It provides developers with architecture overview, deployment instructions, and operational guidance to understand, setup, and maintain the infrastructure.

1.2. Scope

This Terraform module deploys an AWS CloudFront Distribution specifically for the PERFORMANCE environment. It focuses on CloudFront configuration with optimized cache behaviors, custom HTTPS via ACM certificate, and access logging.

Related dependencies relevant to the configuration are managed externally and out of scope for this document:

- How S3 bucket is provisioned and managed about static asset deployment
- Networking layer (VPC, ALB, subnets).
- Certificate creation and domain validation.
- Backend application services behind Load Balancing.

1.3. Definition And Abbreviation

Term	Definition
CDN	Content Delivery Network
SPA	Single Page Application
ALB	Application Load Balancer
ACM	AWS Certificate Manager
TTL	Time To Live

Table 1: Terms

2. Architecture

2.1. Flow

Request Flow: Users accessing `perf.maxapp.com` would hit CloudFront edge locations worldwide.

- Static assets like js and css files are cached up to 24 hours (with `CachingOptimized`) for lightning-fast delivery.
- When user typing `/about` directly or refreshing the page, Cloudfront returns `index.html` so React router can handle the route.
- API calls with prefix `/api/*` and `/auth/*` will bypass cache completely, forwarding all headers + CORS to ALB backend services.

2.2. Functional

- SPA frontend from S3 (cached static + routing)
- API backend via ALB (no-cache, CORS).
- Global CDN, custom HTTPS (TLS 1.2+), access logging to S3

3. Requirements

3.1. Module & Codes

Depends on : Networking layer remote state

3.2. Version & Provider