Fyntrix Frontend - AWS S3 + CloudFront Deployment Guide

Complete guide to deploy Fyntrix frontend to AWS S3 with CloudFront CDN and HTTPS.

Last Updated: January 20, 2026
Deployment Status: ✅ Successfully deployed with HTTPS/SSL

📋 Table of Contents

🎯 Overview

What Gets Deployed

- **Frontend Application**: React + Vite SPA
- **Storage**: AWS S3 bucket (static website hosting)
- **CDN**: CloudFront distribution (HTTPS, caching, global delivery)
- **SSL Certificate**: AWS Certificate Manager (ACM)
- **Domain**: fyntrix.ai (with www.fyntrix.ai)

Current Deployment

- **S3 Bucket**: fyntrix-frontend
- **Region**: ap-south-1 (Mumbai)
- **CloudFront Distribution**: E26V0HY1P4H54K
- **CloudFront Domain**: doyrrqst8fjq3.cloudfront.net
- **SSL Certificate**: arn:aws:acm:us-east-1:563391529004:certificate/654ffd7c-377d-4235-b16e-4f78705b77c6
- **Primary Domain**: https://fyntrix.ai
- **Alternate Domain**: https://www.fyntrix.ai

Important URLs

- **Production**: https://fyntrix.ai
- **WWW**: https://www.fyntrix.ai

- **CloudFront Direct**: https://doyrrqst8fjq3.cloudfront.net
- **S3 Website**: http://fyntrix-frontend.s3-website.ap-south-1.amazonaws.com

---

📋 Prerequisites

1. AWS CLI Configuration

```
# Verify AWS profile
aws sts get-caller-identity --profile fyntrix

# Expected output:
# Account: 563391529004
# Region: ap-south-1
```

2. Build the Frontend

```
cd /Users/adeeb/Documents/Ronttera/Fyntrix/fyntrix-frontend

# Install dependencies (if needed)
npm install

# Build for production
npm run build

# Verify build output
ls -la dist/
```

Build Output Location: `dist/` folder

3. Required Tools

- AWS CLI v2
- Node.js & npm
- Git (for version control)

🏗️ Deployment Architecture

```
┌─────────────────────────────────────────────────────────┐
│                    User Browser                         │
└─────────────────────────────────────────────────────────┘
                     │ HTTPS Request
                     │ https://fyntrix.ai
                     ▼
┌─────────────────────────────────────────────────────────┐
│              Route 53 DNS (fyntrix.ai)                  │
│              A Record → CloudFront                        │
└───────────────────────────────────────────────────────────┘
                              │
                              │
                              ▼
┌───────────────────────────────────────────────────────────┐
│         CloudFront Distribution (Global CDN)              │
│         - HTTPS/SSL (ACM Certificate)                     │
│         - Caching & Compression                           │
│         - Edge Locations Worldwide                        │
└───────────────────────────────────────────────────────────┘
                              │
                              │
                              ▼
┌───────────────────────────────────────────────────────────┐
│         S3 Bucket (fyntrix-frontend)                      │
│         - Static Website Hosting                          │
│         - index.html, assets/, etc.                       │
└───────────────────────────────────────────────────────────┘
```

🚀 Step-by-Step Deployment

Step 1: Create S3 Bucket

```
# Set AWS profile
export AWS_PROFILE=fyntrix

# Create S3 bucket
aws s3 mb s3://fyntrix-frontend --region ap-south-1 --profile fyntrix
```

Output: `make_bucket: fyntrix-frontend`

Step 2: Configure S3 for Website Hosting

```
# Enable static website hosting
aws s3 website s3://fyntrix-frontend \
  --index-document index.html \
  --error-document index.html \
  --profile fyntrix

# Disable public access block
aws s3api put-public-access-block \
  --bucket fyntrix-frontend \
  --public-access-block-configuration \

"BlockPublicAcls=false,IgnorePublicAcls=false,BlockPublicPolicy=false,RestrictPublicBuckets=false" \
  --profile fyntrix
```

```
# Set bucket policy for public read access
cat > /tmp/s3-bucket-policy.json << 'EOF'
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadGetObject",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::fyntrix-frontend/*"
    }
  ]
}
EOF

aws s3api put-bucket-policy \
  --bucket fyntrix-frontend \
  --policy file:///tmp/s3-bucket-policy.json \
  --profile fyntrix
```

Step 3: Upload Build Files to S3

```
# Navigate to frontend directory
cd /Users/adeeb/Documents/Pronttera/Fyntrix/fyntrix-frontend

# Sync build files to S3 (delete removes old files)
aws s3 sync dist/ s3://fyntrix-frontend --delete --profile fyntrix
```

Expected Output: Files uploaded with progress indicators

Verify Upload:

```
# List files in bucket
aws s3 ls s3://fyntrix-frontend/ --recursive --profile fyntrix

# Test S3 website endpoint
curl -I http://fyntrix-frontend.s3-website.ap-south-1.amazonaws.com
```

Step 4: Request SSL Certificate (ACM)

⚠ **IMPORTANT**: CloudFront requires certificates in `us-east-1` region!

```
# Request certificate for fyntrix.ai and *.fyntrix.ai
aws acm request-certificate \
  --domain-name fyntrix.ai \
  --subject-alternative-names "*.fyntrix.ai" \
  --validation-method DNS \
  --region us-east-1 \
  --profile fyntrix
```

Output: Certificate ARN

```
arn:aws:acm:us-east-1:563391529004:certificate/654ffd7c-377d-4235-b16e-
4f78705b77c6
```

Step 5: Validate SSL Certificate via DNS

```
# Get validation records
CERT_ARN="arn:aws:acm:us-east-1:563391529004:certificate/654ffd7c-377d-
4235-b16e-4f78705b77c6"

aws acm describe-certificate \
  --certificate-arn $CERT_ARN \
  --region us-east-1 \
  --profile fyntrix \
  --query 'Certificate.DomainValidationOptions'
```

Add CNAME Record to Route 53:

```
cat > /tmp/ssl-validation-fyntrix.json << 'EOF'
{
  "Changes": [
    {
      "Action": "CREATE",
      "ResourceRecordSet": {
        "Name": "_128a0402e8dc1ba4062600854f8a4e75.fyntrix.ai.",
        "Type": "CNAME",
        "TTL": 300,
        "ResourceRecords": [
          {
            "Value": "_c636a3e244623d0bfd3491c03ba017c7.jkddzztszm.acm-
validations.aws."
          }
        ]
      }
    }
  ]
}
EOF
```

```
aws route53 change-resource-record-sets \
  --hosted-zone-id Z0855289DANZGUB2HQ52 \
  --change-batch file:///tmp/ssl-validation-fyntrix.json \
  --profile fyntrix
```

Wait for Validation (usually 5-10 minutes):

```
aws acm describe-certificate \
  --certificate-arn $CERT_ARN \
  --region us-east-1 \
  --profile fyntrix \
  --query 'Certificate.Status' \
  --output text
```

Expected Output: `ISSUED`

Step 6: Create CloudFront Distribution

```
cat > /tmp/cloudfront-config.json << 'EOF'
{
  "CallerReference": "fyntrix-frontend-2026-01-20",
  "Comment": "Fyntrix Frontend Distribution",
  "Enabled": true,
  "DefaultRootObject": "index.html",
  "Origins": {
    "Quantity": 1,
    "Items": [
      {
        "Id": "S3-fyntrix-frontend",
        "DomainName": "fyntrix-frontend.s3-website.ap-south-1.amazonaws.com",
        "CustomOriginConfig": {
          "HTTPPort": 80,
          "HTTPSPort": 443,
          "OriginProtocolPolicy": "http-only"
        }
      }
    ]
  },
  "DefaultCacheBehavior": {
    "TargetOriginId": "S3-fyntrix-frontend",
    "ViewerProtocolPolicy": "redirect-to-https",
    "AllowedMethods": {
      "Quantity": 2,
      "Items": ["GET", "HEAD"],
      "CachedMethods": {
        "Quantity": 2,
        "Items": ["GET", "HEAD"]
      }
    },
    "Compress": true,
    "ForwardedValues": {
      "QueryString": false,
      "Cookies": {
        "Forward": "none"
      }
    },
    "MinTTL": 0,
    "DefaultTTL": 86400,
    "MaxTTL": 31536000
  },
  "CustomErrorResponses": {
    "Quantity": 1,
    "Items": [
      {
        "ErrorCode": 404,
        "ResponsePagePath": "/index.html",
        "ResponseCode": "200",
        "ErrorCachingMinTTL": 300
      }
    ]
  },
  "Aliases": {
    "Quantity": 2,
    "Items": ["fyntrix.ai", "www.fyntrix.ai"]
  },
  "ViewerCertificate": {
    "ACMCertificateArn": "arn:aws:acm:us-east-1:563391529004:certificate/654ffd7c-377d-4235-b16e-4f78705b77c6",
    "SSLSupportMethod": "sni-only",
    "MinimumProtocolVersion": "TLSv1.2_2021"
  }
}
EOF

aws cloudfront create-distribution \
  --distribution-config file:///tmp/cloudfront-config.json \
  --profile fyntrix
```

Output: Distribution ID and Domain Name

```
{
  "Distribution": {
    "Id": "E26V0HY1P4H54K",
    "DomainName": "doyrrqst8fjq3.cloudfront.net",
    "Status": "InProgress"
  }
}
```

Wait for Deployment (15-20 minutes):

```
aws cloudfront get-distribution \
  --id E26V0HY1P4H54K \
  --profile fyntrix \
  --query 'Distribution.Status' \
  --output text
```

Expected Output: `Deployed`

Step 7: Update DNS to Point to CloudFront

```
cat > /tmp/route53-cloudfront.json << 'EOF'
{
  "Changes": [
    {
      "Action": "UPSERT",
      "ResourceRecordSet": {
        "Name": "fyntrix.ai",
        "Type": "A",
        "AliasTarget": {
          "HostedZoneId": "Z2FDTNDATAQYW2",
          "DNSName": "doyrrqst8fjq3.cloudfront.net",
          "EvaluateTargetHealth": false
        }
      }
    },
    {
      "Action": "UPSERT",
      "ResourceRecordSet": {
        "Name": "www.fyntrix.ai",
        "Type": "A",
        "AliasTarget": {
          "HostedZoneId": "Z2FDTNDATAQYW2",
          "DNSName": "doyrrqst8fjq3.cloudfront.net",
          "EvaluateTargetHealth": false
        }
      }
    }
  ]
}
EOF

aws route53 change-resource-record-sets \
  --hosted-zone-id Z0855289DANZGUB2HQ52 \
  --change-batch file:///tmp/route53-cloudfront.json \
  --profile fyntrix
```

Note: `Z2FDTNDATAQYW2` is the CloudFront hosted zone ID (constant for all CloudFront distributions)

✅ Verification

Test HTTPS Access

```
# Test main domain
curl -I https://fyntrix.ai

# Test www subdomain
curl -I https://www.fyntrix.ai

# Test HTTP redirect to HTTPS
curl -I http://fyntrix.ai

# Test CloudFront directly
curl -I https://doyrrqst8fjq3.cloudfront.net
```

Check DNS Resolution

```
# Check DNS
dig +short fyntrix.ai
dig +short www.fyntrix.ai

# Should return CloudFront IPs (e.g., 13.225.5.55)
```

Open in Browser

```
# macOS
open https://fyntrix.ai

# Linux
xdg-open https://fyntrix.ai
```

Verify SSL Certificate

```
# Check certificate details
openssl s_client -connect fyntrix.ai:443 -servername fyntrix.ai < /dev/null 2>/dev/null | openssl x509 -noout -dates -subject
```

🔄 Updates & Maintenance

Deploy New Version

```
# 1. Build the frontend
cd /Users/adeeb/Documents/Ponttera/Fyntrix/fyntrix-frontend
npm run build

# 2. Upload to S3
aws s3 sync dist/ s3://fyntrix-frontend --delete --profile fyntrix

# 3. Invalidate CloudFront cache (force refresh)
aws cloudfront create-invalidation \
  --distribution-id E26V0HY1P4H54K \
  --paths "/*" \
  --profile fyntrix
```

Invalidation Time: 1-5 minutes

Quick Update Script

Create `deploy-frontend.sh`:

```
#!/bin/bash
set -e

echo "🏗️  Building frontend..."
npm run build

echo "📤 Uploading to S3..."
aws s3 sync dist/ s3://fyntrix-frontend --delete --profile fyntrix

echo "🔄 Invalidating CloudFront cache..."
aws cloudfront create-invalidation \
  --distribution-id E26V0HY1P4H54K \
  --paths "/*" \
  --profile fyntrix

echo "✅ Deployment complete!"
echo "🌐 Visit: https://fyntrix.ai"
```

Make it executable:

```
chmod +x deploy-frontend.sh
./deploy-frontend.sh
```

🔧 Troubleshooting

Issue: 403 Forbidden Error

Cause: S3 bucket policy not set correctly

Solution:

```
# Verify bucket policy
aws s3api get-bucket-policy --bucket fyntrix-frontend --profile fyntrix

# Re-apply policy if needed
aws s3api put-bucket-policy \
  --bucket fyntrix-frontend \
  --policy file:///tmp/s3-bucket-policy.json \
  --profile fyntrix
```

Issue: Certificate Not Validating

Cause: DNS CNAME record not added or incorrect

Solution:

```
# Check certificate status
aws acm describe-certificate \
  --certificate-arn arn:aws:acm:us-east-
1:563391529004:certificate/654ffd7c-377d-4235-b16e-4f78705b77c6 \
  --region us-east-1 \
  --profile fyntrix

# Verify CNAME record in Route 53
aws route53 list-resource-record-sets \
  --hosted-zone-id Z0855289DANZGUB2HQ52 \
  --profile fyntrix | grep -A 5 "_128a0402e8dc1ba4062600854f8a4e75"
```

Issue: Old Content Still Showing

Cause: CloudFront cache not invalidated

Solution:

```
# Create cache invalidation
aws cloudfront create-invalidation \
  --distribution-id E26V0HY1P4H54K \
  --paths "/*" \
  --profile fyntrix

# Check invalidation status
aws cloudfront list-invalidations \
  --distribution-id E26V0HY1P4H54K \
  --profile fyntrix
```

Issue: 404 on React Routes

Cause: CloudFront not configured to handle SPA routing

Solution: Already configured via `CustomErrorResponses` - 404 errors return `index.html` with 200 status

Issue: DNS Not Resolving

Cause: DNS propagation delay or local DNS cache

Solution:

```
# Clear local DNS cache (macOS)
sudo dscacheutil -flushcache
sudo killall -HUP mDNSResponder

# Test with Google DNS
dig @8.8.8.8 fyntrix.ai

# Wait 5-10 minutes for DNS propagation
```

📞 Quick Reference

Current Deployment

- **S3 Bucket**: fyntrix-frontend
- **Region**: ap-south-1 (Mumbai)
- **CloudFront ID**: E26V0HY1P4H54K
- **CloudFront Domain**: doyrrqst8fjq3.cloudfront.net
- **SSL Certificate**: arn:aws:acm:us-east-1:563391529004:certificate/654ffd7c-377d-4235-b16e-4f78705b77c6
- **Route 53 Hosted Zone**: Z0855289DANZGUB2HQ52

Important URLs

- **Production**: https://fyntrix.ai
- **WWW**: https://www.fyntrix.ai
- **CloudFront**: https://doyrrqst8fjq3.cloudfront.net
- **S3 Website**: http://fyntrix-frontend.s3-website.ap-south-1.amazonaws.com

Quick Commands

```
# Set AWS profile
export AWS_PROFILE=fyntrix

# Build frontend
cd /Users/adeeb/Documents/Ronttera/Fyntrix/fyntrix-frontend
npm run build

# Deploy to S3
aws s3 sync dist/ s3://fyntrix-frontend --delete --profile fyntrix

# Invalidate CloudFront cache
aws cloudfront create-invalidation \
  --distribution-id E26V0HY1P4H54K \
  --paths "/*" \
  --profile fyntrix

# Check CloudFront status
aws cloudfront get-distribution \
  --id E26V0HY1P4H54K \
  --profile fyntrix \
  --query 'Distribution.Status'

# Check certificate status
aws acm describe-certificate \
  --certificate-arn arn:aws:acm:us-east-
1:563391529004:certificate/654ffd7c-377d-4235-b16e-4f78705b77c6 \
  --region us-east-1 \
  --profile fyntrix \
  --query 'Certificate.Status'

# List S3 files
aws s3 ls s3://fyntrix-frontend/ --recursive --profile fyntrix

# Test website
curl -I https://fyntrix.ai
open https://fyntrix.ai
```

Cost Estimation

Monthly Costs (approximate):

Service	Usage	Cost
S3 Storage	~10 MB	$0.02
S3 Requests	10,000 requests	$0.05
CloudFront	10 GB transfer	$1.00
Route 53	Hosted zone + queries	$0.50
ACM Certificate	Free	$0.00

Service	Usage	Cost
Total		**~$1.57/month**

Free Tier Benefits:

- CloudFront: 1 TB transfer/month (first 12 months)
- S3: 5 GB storage, 20,000 GET requests (first 12 months)

🎉 Deployment Complete!

Last Successful Deployment: January 20, 2026, 1:40 AM IST
Deployment Status: ✅ Live on HTTPS with CloudFront CDN
Domain: https://fyntrix.ai
SSL Status: ✅ Valid Certificate
CloudFront Status: ✅ Deployed

Access Your Application:

- **Main Site**: https://fyntrix.ai
- **WWW**: https://www.fyntrix.ai

Next Steps:

1. Test all frontend routes and functionality
2. Verify API integration with backend (https://api.fyntrix.ai)
3. Set up monitoring and analytics
4. Configure custom error pages if needed
5. Add CloudWatch alarms for CloudFront metrics

Questions or Issues?

- Check the Troubleshooting section
- Review CloudFront logs in S3 (if enabled)
- Check CloudWatch metrics for CloudFront distribution