

Cloud Configuration

Cloud configuration est un utilitaire permettant de configurer une application mobile sans avoir à redéployer une nouvelle version.

- [URL des Endpoints AWS et Docker](#)
 - [URL et Configurations sur AWS](#)
- [Déployer Web Assembly Blazor sur AWS](#)

URL des Endpoints AWS et Docker

URL et Configurations sur AWS

Les URL de connexion pour les API AWS sont les suivantes :

DEV

Composant	URL
Web Admin	https://cloudconfiguration-qvr.pe-qvr.fr/
API Admin	https://cloudconfiguration-api-admin-qvr.pe-qvr.fr
API Mobile	https://cloudconfiguration-api-qvr.pe-qvr.fr
Postgres BDD	user : Postgres db name : mobilesquadbddtempublique1012 read/write : db-mobilesquad-publique-cluster.cluster-cliviqvbr8hu.eu-west-3.rds.amazonaws.com readonly : db-mobilesquad-publique-cluster.cluster-ro-cliviqvbr8hu.eu-west-3.rds.amazonaws.com

Fabricant

Composant	URL
Web Admin	https://cloudconfiguration-qvr.pe-qvr.fr/
API Admin	https://cloudconfiguration-api-admin-qvr.pe-qvr.fr
API Mobile	https://cloudconfiguration-api-qvr.pe-qvr.fr
Cognito Login URL	https://cloudconfiguration-qvr.pe-qvr.fr/authentication/login-callback

Cognito Logout URL	https://cloudconfiguration-qvr.pe-qvr.fr/authentication/logout-callback
------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------

Production

Composant	URL
Web Admin	https://cloudconfiguration.pole-emploi.fr
API Admin	https://cloudconfiguration-api-admin.pole-emploi.fr
API Mobile	https://cloudconfiguration-api.pole-emploi.fr
Cognito Login URL	https://cloudconfiguration.pole-emploi.fr/authentication/login-callback
Cognito Logout URL	https://cloudconfiguration.pole-emploi.fr/authentication/logout-callback

Redirection DNS :

cloudconfigurationapi.magic-time.fr => d-agim7csjdi.execute-api.eu-west-3.amazonaws.com

Déployer Web Assembly Blazor sur AWS

La Technologie employée sur AWS s'appelle CloudFront, elle permet la mise en cache sur internet de fichier sur ce que l'on appelle des CDN.

Le Web Assembly Blazor sera déposé sur un bucket S3 avec un accès depuis le CloudFront.

Voici un exemple de YAML utilisé pour effectuer créer la configuration sur CloudFront.

```
AWSTemplateFormatVersion: '2010-09-09'
Description: '(Cloudfront-as-default-03) Simple CloudFront distribution with an S3 origin'

Parameters:
  S3BucketName:
    Type: String
    Default: Your S3 Bucket Name
    Description: Name of existing S3 bucket

  OAIEnabled:
    Type: String
    Default: "no"
    AllowedValues: ["yes", "no"]
    Description: Enable OAI (Access to bucket only through CloudFront)

Conditions:
  WithOAIEnabled: !Equals [!Ref "OAIEnabled", "yes"]
  WithoutOAIEnabled: !Equals [!Ref "OAIEnabled", "no"]

Resources:
  S3BucketPolicyWithAOI:
```

Metadata:

Comment: 'Bucket policy to allow cloudfront to access the data'

Properties:

Bucket: !Ref S3BucketName

PolicyDocument:

Statement:

- Action:

- 's3: GetObject'

Effect: 'Allow'

Principal:

CanonicalUser: !GetAtt CfOriginAccessIdentity.S3CanonicalUserId

Resource:

- !Sub 'arn: aws: s3: :: \${S3BucketName}/*'

Type: 'AWS::S3::BucketPolicy'

Condition: WithOAIEnabled

CfDistribution:

Type: "AWS::CloudFront::Distribution"

Condition: WithoutOAIEnabled

Metadata:

Comment: 'A simple CloudFront distribution with an S3 origin'

Properties:

DistributionConfig:

Comment: 'A simple distribution with an S3 origin'

DefaultCacheBehavior:

AllowedMethods:

- 'HEAD'

- 'GET'

CachedMethods:

- 'HEAD'

- 'GET'

Compress: false

DefaultTTL: 86400

ForwardedValues:

Cookies:

Forward: 'none'

Headers:

- 'Origin'

QueryString: false

MaxTTL: 31536000

MinTTL: 86400
TargetOriginId: !Sub 's3-origin- \${S3BucketName}'
ViewerProtocolPolicy: 'redirect-to-https'
DefaultRootObject: 'index.html'
Enabled: true
HttpVersion: 'http1.1'
IPV6Enabled: false
Origins:
- DomainName: !Sub '\${S3BucketName}.s3.amazonaws.com'
 Id: !Sub 's3-origin- \${S3BucketName}'
 S3OriginConfig: {}
PriceClass: 'PriceClass_All'

CfDistributionWithAOI:

Type: "AWS::CloudFront::Distribution"

Condition: WithOAIEnabled

Metadata:

Comment: 'A simple CloudFront distribution with an S3 origin'

Properties:

DistributionConfig:

Comment: 'A simple distribution with an S3 origin'

DefaultCacheBehavior:

AllowedMethods:

- 'HEAD'
- 'GET'

CachedMethods:

- 'HEAD'
- 'GET'

Compress: false

DefaultTTL: 86400

ForwardedValues:

Cookies:

Forward: 'none'

Headers:

- 'Origin'

QueryString: false

MaxTTL: 31536000

MinTTL: 86400

TargetOriginId: !Sub 's3-origin- \${S3BucketName}'

```
ViewerProtocolPolicy: 'redirect-to-https'
DefaultRootObject: 'index.html'
Enabled: true
HttpVersion: 'http1.1'
IPV6Enabled: false
Origins:
  - DomainName: !Sub '${S3BucketName}.s3.amazonaws.com'
    Id: !Sub 's3-origin-${S3BucketName}'
    OriginPath: ''
    S3OriginConfig:
      OriginAccessIdentity: !Sub 'origin-access-identity/cloudfront/${CfOriginAccessIdentity}'
      PriceClass: 'PriceClass_All'

CfOriginAccessIdentity:
  Type: "AWS::CloudFront::CloudFrontOriginAccessIdentity"
  Condition: WithOAIEnabled
  Metadata:
    Comment: 'Access S3 bucket content only through CloudFront'
  Properties:
    CloudFrontOriginAccessIdentityConfig:
      Comment: 'Access S3 bucket content only through CloudFront'

Outputs:
  S3BucketName:
    Description: 'Bucket name'
    Value: !Ref S3BucketName
  CfDistributionWithOAI:
    Description: 'Id for our cloudfront distribution'
    Value: !Ref CfDistributionWithAOI
    Condition: WithOAIEnabled
  CfDistributionId:
    Description: 'Id for our cloudfront distribution'
    Value: !Ref CfDistribution
    Condition: WithoutOAIEnabled
  CfDistributionDomainNameWithOAI:
    Description: 'Domain name for our cloudfront distribution'
    Value: !GetAtt CfDistributionWithAOI.DomainName
    Condition: WithOAIEnabled
  CfDistributionDomainName:
```

Description: 'Domain name for our cloudfront distribution'

Value: !GetAtt CfDistribution.DomainName

Condition: WithOutOAIEnabled