

Config AWS Access Key

You can use the AWS Management Console to create/manage the access keys. Follow the below steps to create AWS Access Key 1. sign in to the AWS console.



Sign in

Root user
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

IAM user
User within an account that performs daily tasks.
[Learn more](#)

Root user email address

Next

By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

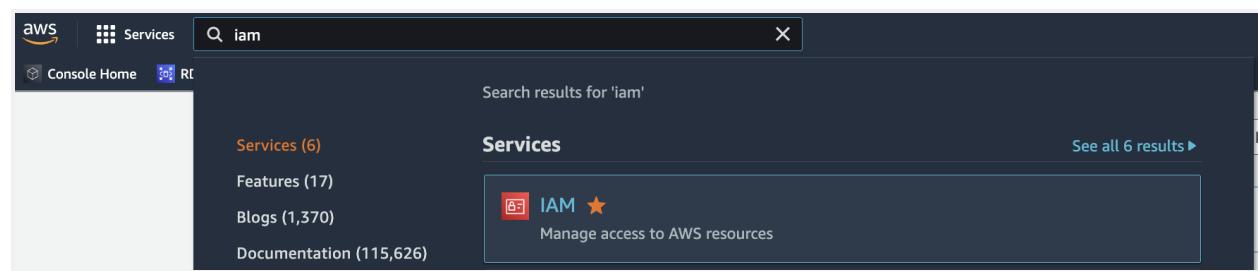
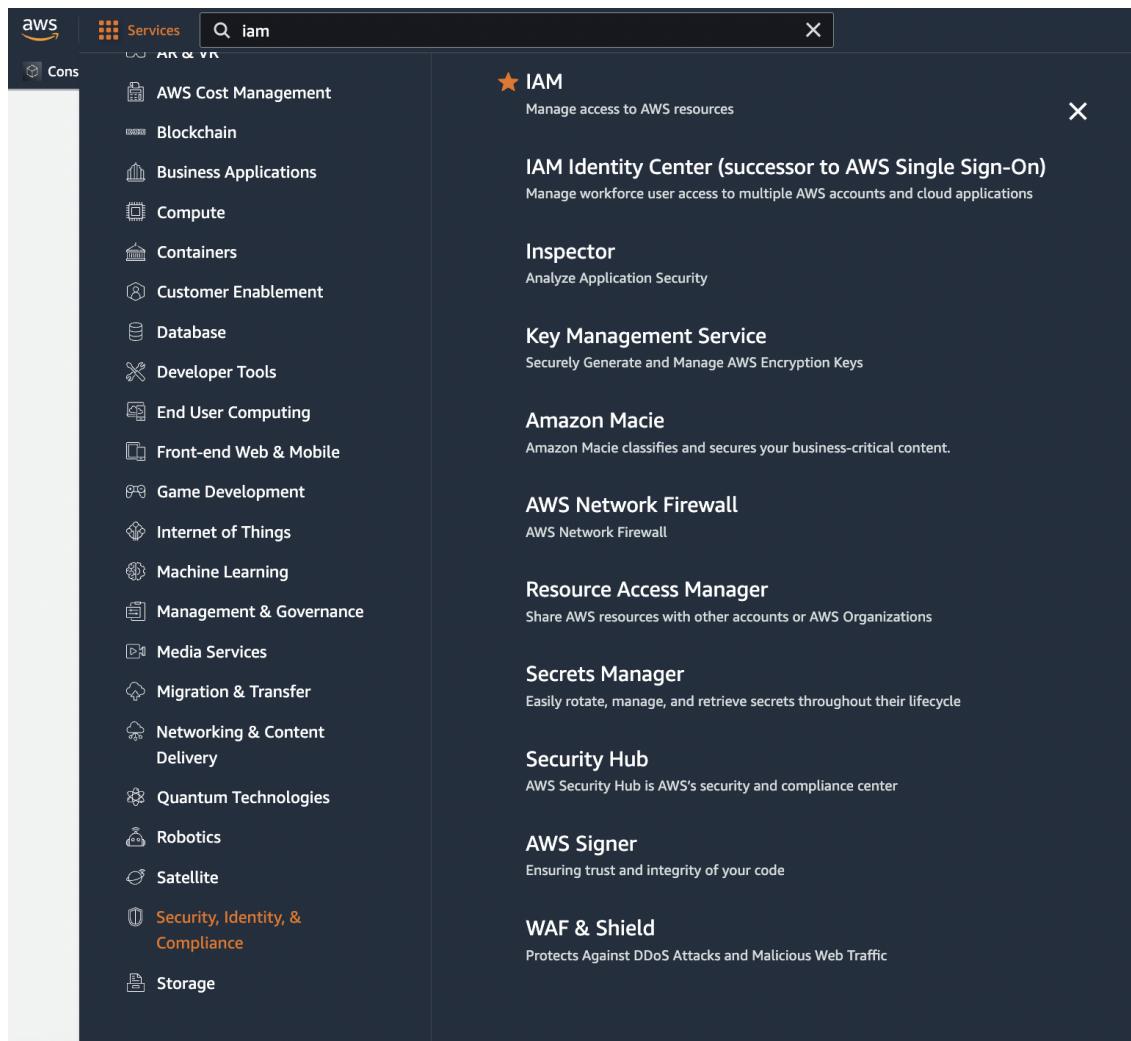
— New to AWS? —

Create a new AWS account

Select “Root User” and key in the email address and password.

For your convenience, the AWS sign-in page uses a browser cookie to remember your user name and account information. If you previously signed in as a different user, choose Sign in to a different account near the bottom of the page to return to the main sign-in page.

Click Services -> Security, Identity, & Compliance -> IAM. Or enter “iam” in the search box.



Select Users -> Add Users

The screenshot shows the AWS Identity and Access Management (IAM) service dashboard. At the top, there's a search bar labeled "Search for serv". Below it, a navigation bar includes links for "Console Home", "RDS", "IAM", and "S3". The main content area has a title "Identity and Access Management (IAM)" with a close button. A search bar "Search IAM" is present. The left sidebar contains a navigation menu:

- Dashboard**
- Access management**
 - User groups
 - Users
 - Roles
 - Policies
 - Identity providers
 - Account settings
- Access reports**
 - Access analyzer
 - Archive rules
 - Analyzers
 - Settings
- Credential report
- Service control policies (SCPs)

IAM > Users

Users (1) Info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

[Edit](#) [Delete](#) [Add users](#)

<input type="checkbox"/>	User name	Groups	Last activity	MFA	Password a...	Active key
<input type="checkbox"/>	olinkuser1	None	6 hours ago	None	40 days ago	40 days

Key in “User Name”, Tick Access type: “Programmatic access” and , then click “Next Permissions”

Add user

1 2 3 4 5

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name*

[+ Add another user](#)

Select AWS access type

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Select AWS credential type*

Access key - Programmatic access

Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

Password - AWS Management Console access

Enables a **password** that allows users to sign-in to the AWS Management Console.

* Required

[Cancel](#)

[Next: Permissions](#)

You may copy permissions from an existing user.

Add user

1 2 3 4 5

Set permissions

 Add user to group

 Copy permissions from existing user

 Attach existing policies directly

Select an existing user from which to copy policies and group membership.

Copy permissions from existing user

Showing 1 result		
User name	Groups	Attached policies
olinkuser1	None	AmazonRDSDataFullAccess and 4 more

Set permissions boundary

[Cancel](#) [Previous](#) [Next: Tags](#)

Or Search and select the following permissions: RDSDDataFullAccess, RDSFullAccess, CloudFrontReadOnlyAccess, S3FullAccess, getUserpolicy.

▼ Permissions policies (5 policies applied)

[Add permissions](#)

[+ Add inline policy](#)

Policy name ▾	Policy type ▾	
Attached directly		
▶ AmazonRDSDDataFullAccess	AWS managed policy	✖
▶ AmazonRDSFullAccess	AWS managed policy	✖
▶ CloudFrontReadOnlyAccess	AWS managed policy	✖
▶ AmazonS3FullAccess	AWS managed policy	✖
▶ getUserpolicy	Managed policy	✖

“Add Tags” are optional, Click “Next: Review”. Then, Click “Create User”.

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name olinkuser2

AWS access type Programmatic access - with an access key

Permissions boundary Permissions boundary is not set

Permissions summary

The following groups and policies will be copied from the selected existing user and attached to the user shown above.

Type	Name
Managed policy	AmazonRDSDDataFullAccess
Managed policy	AmazonRDSFullAccess
Managed policy	CloudFrontReadOnlyAccess
Managed policy	AmazonS3FullAccess
Managed policy	getUserpolicy

Tags

[Cancel](#)

[Previous](#)

[Create user](#)

Click “Download .csv”, save it to the folder oLink installed, then click “Close”.

Add user

- 1
- 2
- 3
- 4
- 5

Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://327767585884.signin.aws.amazon.com/console>

Download .csv

User	Access key ID	Secret access key
olinkuser2		***** Show

Close

The Access credentials will be used in this screen in the olink application.

oLink

AWS Credentials | Close

Profile List	<input type="text"/>	Db Server	<input type="text"/>
Profile Name	<input type="text"/>	DB User Name	<input type="text"/>
Profile Id	<input type="text"/>	DB Password	<input type="text"/>
Profile Key	<input type="text"/>	Create Profile	
File Bucket	<input type="text"/>	Secret access key	

Create S3 Bucket

sign in to the AWS console using “Root User”. Click “All Services” -> S3. Or search S3 in the search box.

The screenshot shows two views of the AWS Management Console:

Top View (Storage Section):

- Left sidebar: Identity and Access Management (IAM), Access management, Users, Roles, Policies, Quantum Technologies, Robotics, Satellite, Security, Identity, & Compliance, Storage.
- Search bar: Search for services, features, blogs, docs, and more [Option+S].
- Main content: Storage section with links to AWS Backup, EFS, AWS Elastic Disaster Recovery, FSx, S3 (highlighted with a star), S3 Glacier, and Storage Gateway.

Bottom View (Search Results for S3):

- Left sidebar: Identity and Access Management (IAM), Dashboard, Access management, User groups, Users, Roles, Policies.
- Search bar: Search results for 'S3'.
- Main content: Services section showing results for S3, S3 Glacier, and Athena.

Click Button “Create Bucket”

The screenshot shows the AWS S3 Buckets page. At the top left, it says "Amazon S3 > Buckets". Below that is an "Account snapshot" section with a link to "View Storage Lens dashboard". The main area is titled "Buckets (3)" with an "Info" link. It shows three buckets: "olinkuser2-s3" (size 0), "olinkuser2-test" (size 0), and "olinkuser2-test2" (size 0). There are buttons for "Copy ARN", "Empty", "Delete", and a large orange "Create bucket" button. Below the buckets is a search bar with placeholder text "Find buckets by name" and a page navigation area showing "1" of "1". The table headers are "Name", "AWS Region", "Access", and "Creation date".

In order for olink application to work smoothly, use this name convention.

- **Naming rule: [username] + “-S3”**
- **[username] has to be the one who is supposed to have access to this bucket**
- **Example: olinkuser2-s3**

In Bucket name, enter a DNS-compliant name for your bucket.

Also, the bucket name must

- Be unique across all of your Amazon S3.
- Bucket names must be between 3 (min) and 63 (max) characters long.
- Bucket names can consist only of lowercase letters, numbers, dots (.), and hyphens (-).
- Bucket names must begin and end with a letter or number.
- Bucket names must not contain two adjacent periods.
- Avoid including sensitive information, such as account numbers, in the bucket name. The bucket name is visible in the URLs that point to the objects in the bucket.
- [MORE](#)

After you create the bucket, you can't change its name. In Region, select “US East (N. Virginia) us-east-1”

Create bucket Info

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name

abc-ded-fde

Bucket name must be globally unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

US East (N. Virginia) us-east-1



Copy settings from existing bucket - optional

Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Keeping “Block All Public Access” is recommended.

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Block public access to buckets and objects granted through new access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

Block public access to buckets and objects granted through any access control lists (ACLs)

S3 will ignore all ACLs that grant public access to buckets and objects.

Block public access to buckets and objects granted through new public bucket or access point policies

S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

Block public and cross-account access to buckets and objects through any public bucket or access point policies

S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Click Button “Create Bucket”

Save the bucket name as S3 Bucket in the oLink Management Console Tab.

USE “cloudfont.net” domain to hide and protect S3 bucket

Advantage:

Prevent S3 from being discovered

More Comprehensive settings are available to protect the buckets

Use your own domain name later on (Amazon Route53)

With CloudFront Functions, you can write lightweight functions in JavaScript for high-scale, latency-sensitive CDN customizations.

URL appears simpler

Steps:

Go to CloudFront service and click “Create Distribution”.

On the next screen, Choose an Origin such as a S3 bucket.

The screenshot shows the AWS CloudFront 'Create distribution' wizard. At the top, there's a navigation bar with links for Console Home, RDS, IAM, S3, Lambda, API Gateway, EC2, CloudFront, and VPC. Below the navigation bar, the main title is 'Create distribution'. On the left, there's a sidebar with a 'Create' button. The main content area is titled 'Origin' and contains a section for 'Origin domain'. It says 'Choose an AWS origin, or enter your origin's domain name.' Below this is a search input field with the placeholder 'Choose origin domain'. A list of 'Amazon S3' origins is shown, with three items listed: 'olinkky.s3.amazonaws.com', 'olinkuser1-s3.s3.amazonaws.com', and 'olinkuser1s3.s3.amazonaws.com'. The entire interface has a light gray background with blue highlights for interactive elements.

For “Origin Access”, select “Origin access control settings (recommended)” and click “Create control setting” and give it a name e.g. “olinks-OAI”.

Click “Create Distribution” to save and here is the resulting Domain Name and URL to access the private bucket.

<https://d1699jf5nad3mc.cloudfront.net/show.htm>

Distribution domain name “d1699jf5nad3mc.cloudfront.net”

Distribution ID “[E16VQLAFI1IZTZ](#)”

Distribution origin name “myolinkforuser1”

Select the ID and click “Edit” to “Edit origin”

E16VQLAFI1IZTZ

General Origins Behaviors Error pages Geographic restrictions Invalidations Tags

Origins

Filter origins by property or value

Origin name	Origin domain	Origin path	Origin type	Origin Shield region	Origin ...
myolinkforuser1	olinkuser1-s3.s3.us-east-1.amazonaws.com		S3	-	E2358BU.

In section “Origin access”, select “Origin access control settings (recommended)”, and select from the drop down “Origin access control” (e.g. “olinks-OAI”). Or click button “Create control setting”.

Origin access | Info

Public
Bucket must allow public access.

Origin access control settings (recommended)
Bucket can restrict access to only CloudFront.

Legacy access identities
Use a CloudFront origin access identity (OAI) to access the S3 bucket.

Origin access control

Select an existing origin access control (recommended) or create a new configuration.

olinks-OAI

Create control setting

ⓘ You must allow access to CloudFront using this policy statement. Learn more about giving CloudFront permission to access the S3 bucket [\[\]](#).

Go to S3 bucket permissions [\[\]](#)

The click “Copy Policy”. The bold text below shows the copied bucket policy.

Bucket policy (permission)

```
{  
  "Version": "2008-10-17",
```

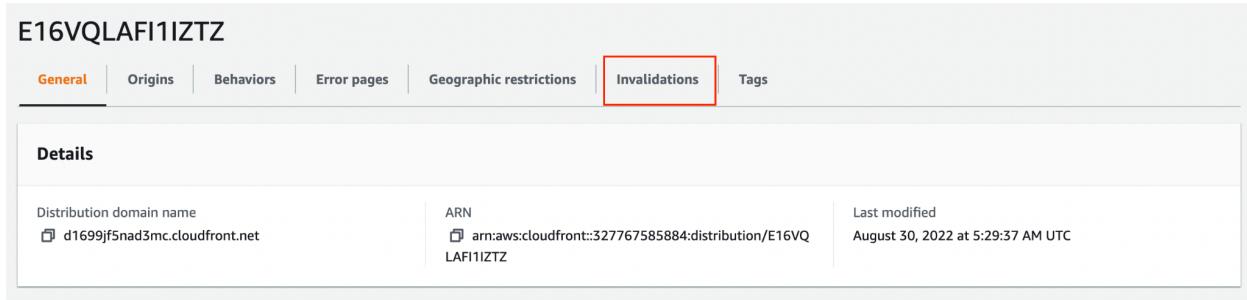
```
"Id": "PolicyForCloudFrontPrivateContent",
"Statement": [
    {
        "Sid": "AllowCloudFrontServicePrincipal",
        "Effect": "Allow",
        "Principal": {
            "Service": "cloudfront.amazonaws.com"
        },
        "Action": "s3:GetObject",
        "Resource": "arn:aws:s3:::olinkuser1-s3/*",
        "Condition": {
            "StringEquals": {
                "AWS:SourceArn":
                    "arn:aws:cloudfront::327767585884:distribution/E16VQLAFI1IZTZ"
            }
        }
    }
],
{
    "Version": "2008-10-17",
    "Id": "PolicyForCloudFrontPrivateContent",
    "Statement": [
        {
            "Sid": "s3policyforcf",
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudfront.amazonaws.com"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::olinkuser1-s3/*",
            "Condition": {
                "StringLike": {
                    "AWS:SourceArn":
                        "arn:aws:cloudfront::327767585884:distribution/*"
                }
            }
        }
    ]
}
```

```
}
```

Public Access Policy

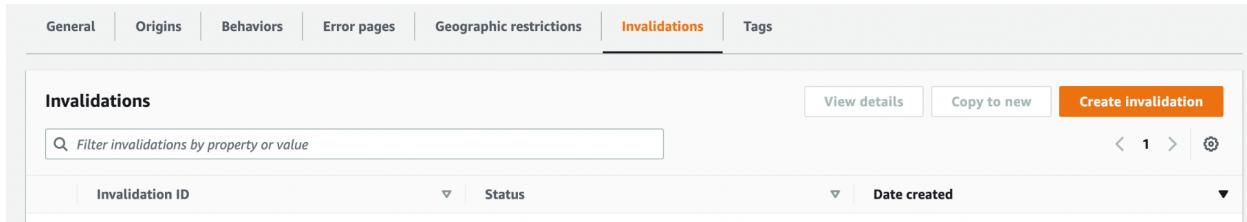
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::olinkuser1-s3/*"
    }
  ]
}
```

In order for CloudFront to refresh as S3 bucket updates its content, please add “Invalidation”.



The screenshot shows the AWS CloudFront console interface. At the top, there's a navigation bar with tabs: General, Origins, Behaviors, Error pages, Geographic restrictions, Invalidations (which has a red box around it), and Tags. Below the tabs, there's a section titled 'Details' containing fields for 'Distribution domain name' (d1699jf5nad3mc.cloudfront.net) and 'ARN' (arn:aws:cloudfront::327767585884:distribution/E16VQ LAFI1IZTZ). To the right, it shows 'Last modified' as 'August 30, 2022 at 5:29:37 AM UTC'.

Click “Create Invalidation” button



The screenshot shows the 'Invalidations' tab selected in the CloudFront distribution configuration. At the top, there's a search bar labeled 'Filter invalidations by property or value'. Below it, there are columns for 'Invalidation ID', 'Status', and 'Date created'. On the right side, there are buttons for 'View details', 'Copy to new', and a prominent orange 'Create invalidation' button.

Type in “/*” in the “Add object paths” field and click “Create invalidation”.

Create invalidation

Object paths

Add object paths

Add the path for each object that you want to remove from the CloudFront cache. You can use wildcards (*).

/

 To add object paths individually, use the [standard editor](#).

Cancel

Create invalidation

CloudFront Background: Amazon CloudFront is a fast content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency, high transfer speeds, all within a developer-friendly environment.
<https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/Introduction.html>

REFERENCE

AWS CloudFront CDN for S3 Tutorial

<https://www.youtube.com/watch?v=-DDGYzKtNwc>

React App on AWS S3 with Static Hosting + Cloudfront

<https://www.youtube.com/watch?v=mls8tiil3uc>

Developer's Guide to Cloudfront and static web

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/AmazonCloudFront_DevGuide.pdf

Cloudfront Pricing

1 TB of data transfer out

10,000,000 HTTP or HTTPS Requests

2,000,000 CloudFront Function Invocations

USER creation and permissions

List of permissions to be added:

The screenshot shows the AWS IAM User Summary page for a user named 'olinkuser1'. The user ARN is arn:aws:iam::327767585884:user/olinkuser1. The creation time is 2022-07-28 07:56 PDT. The 'Permissions' tab is selected. There are five policies applied: AmazonRDSDataFullAccess, AmazonRDSFullAccess, CloudFrontReadOnlyAccess, AmazonS3FullAccess, and getUserpolicy. The 'getUserpolicy' is a Managed policy.

Policy name	Policy type
AmazonRDSDataFullAccess	AWS managed policy
AmazonRDSFullAccess	AWS managed policy
CloudFrontReadOnlyAccess	AWS managed policy
AmazonS3FullAccess	AWS managed policy
getUserpolicy	Managed policy

“AmazonRDSDataFullAccess”

“AmazonRDSFullAccess”

“CloudFrontReadOnlyAccess”

“AmazonS3FullAccess”

“getUserpolicy”: Allow GetUser (Read) from IAM service

```
ListDistributions()
    ListDistributionResponse.DistributionList.Items[0]
        Amazon.CloudFront.Model.DistributionSummary
            Origins.Items[0]

<Amazon.CloudFront.Model.Origin>((<Amazon.CloudFront.Model.Distributi
onSummary>(dl.DistributionList.Items).Items[0]).Origins.Items).Items[0]).Do
mainName = "olinkuser1-s3.s3.us-east-1.amazonaws.com"
<Amazon.CloudFront.Model.Origin>((<Amazon.CloudFront.Model.Distributi
onSummary>(dl.DistributionList.Items).Items[0]).Origins.Items).Items[0]).Id
= "myolinkforuser1"
(new
System.Collections.Generic.Mscorlib_CollectionDebugView<Amazon.Clu
dFront.Model.Origin>((new
System.Collections.Generic.Mscorlib_CollectionDebugView<Amazon.Clu
dFront.Model.DistributionSummary>(dl.DistributionList.Items).Items[0]).Orig
ins.Items).Items[0]).OriginShield???
(new
System.Collections.Generic.Mscorlib_CollectionDebugView<Amazon.Clu
dFront.Model.Origin>((new
System.Collections.Generic.Mscorlib_CollectionDebugView<Amazon.Clu
dFront.Model.DistributionSummary>(dl.DistributionList.Items).Items[0]).Orig
ins.Items).Items[0]).S3OriginConfig???
```