

CloudFrontConfig

August 17, 2026

CloudFrontConfig

Kind: Interface**Source:** `src/adapter/lambda-edge/handler.ts`**Part of:** [Adapter](#)

`CloudFrontConfig` describes CloudFront metadata extracted from a Lambda@Edge event. It carries the distribution identity, event type, and request identifier so handler code can route, log, or correlate requests.

Properties

Property	Type
<code>distributionDomainName</code>	<code>string</code>
<code>distributionId</code>	<code>string</code>
<code>eventType</code>	<code>string</code>
<code>requestId</code>	<code>string</code>

Diagram

```
graph LR
    Event[Lambda@Edge Event] --> Config[CloudFrontConfig]
    Config --> Domain[distributionDomainName]
    Config --> Distribution[distributionId]
    Config --> Type[eventType]
    Config --> Request[requestId]
    Config --> Handler[Edge Handler Logic]
```

Usage

```
import type { CloudFrontConfig } from "./handler";

function logCloudFrontRequest(config: CloudFrontConfig): void {
  console.log({
    distributionId: config.distributionId,
    domainName: config.distributionDomainName,
    eventType: config.eventType,
    requestId: config.requestId,
  });
}

const config: CloudFrontConfig = {
  distributionDomainName: "example.cloudfront.net",
  distributionId: "E123ABC",
  eventType: "viewer-request",
  requestId: "request-id",
};

logCloudFrontRequest(config);
```

AI Coding Instructions

- Keep all `CloudFrontConfig` fields as strings when mapping Lambda@Edge event data.
- Read `distributionId` and `distributionDomainName` from the CloudFront event configuration.
- Pass `requestId` into logs or error context to correlate handler activity.
- Treat `eventType` as event metadata and avoid assuming a single CloudFront trigger type.

How it works

`CloudFrontConfig` is an exported TypeScript interface for the `cf.config` object in a Lambda@Edge event. It requires four string fields: `distributionDomainName`, `distributionId`, `eventType`, and `requestId`. [src/adapters/lambda-edge/handler.ts:62-67]

A `CloudFrontEvent` stores this object at `cf.config`, and `CloudFrontEdgeEvent` is an object whose `Records` array contains those events. [src/adapters/lambda-edge/handler.ts:69-79] The Lambda@Edge adapter also re-exports `CloudFrontConfig` as a type from its public entry point. [src/adapters/lambda-edge/index.ts:6-14]

When `handle()` processes an event, it passes the first record's configuration to `app.fetch()` in the environment/bindings object as `config`. [src/adapters/lambda-

edge/handler.ts:124-141] Its `distributionDomainName` is also used as the request URL host when the first `host` request header is absent. [src/adapter/lambda-edge/handler.ts:164-170]

The interface declares no optional fields, runtime validation, defaults, or error handling. [src/adapter/lambda-edge/handler.ts:62-67]