Your team wants only paid user should be able to download video , how can you achieve this in CloudFront.

Problem

- By default, **anyone with a CloudFront URL** can access your content.
- You want **only authorized users** (e.g., paid subscribers) to download/watch videos.
- Solution → Use **Signed URLs / Signed Cookies** in CloudFront.

✅ **Part 1: How to configure CloudFront**

Step 1: Make your origin (S3) private

- Don't allow public access to videos.
- Attach **OAC (Origin Access Control)** so CloudFront alone can fetch from S3.

Step 2: Enable Trusted Signers

- In CloudFront distribution → choose **Require Signed URLs**.
- This means: CloudFront will serve the file **only if a valid signature/token is present**.

Step 3: Generate Keys

- Create a CloudFront **key pair** (public/private).
- Store **public key** in CloudFront as a *trusted key group*.
- Keep **private key** safe in your backend (e.g., API service).

Step 4: Generate Signed URLs

- Your backend app generates a **signed URL** for a user who is allowed.
- Example:

- **https://d123.cloudfront.net/videos/movie.mp4?Expires=1735647600&Signature=xyz&Key-Pair-Id=ABCD1234**
- Expires → When URL stops working.
- Signature → Cryptographic proof.
- Key-Pair-Id → Identifies which CloudFront public key to use.

👉 CloudFront checks the signature. If valid → serves video. If not → 403 Forbidden.

Summary

- **How to configure?**
  - Make S3 private, enable CloudFront **signed URLs**, create key pairs, and let only signed requests access content.
- **DevOps automation?**
  - Store private key securely.
  - Build a small service that generates signed URLs.
  - CI/CD pipeline ensures CloudFront + backend configs are deployed and tested.

👉 Real-world: OTT platforms (like Netflix, Hotstar) use this pattern for secure video delivery.

By default, what happens if you share a CloudFront URL with someone?
a) It can only be accessed by authorized users.
b) Anyone with the URL can access the content.
c) CloudFront blocks all requests without a signed URL.
d) Access is restricted by default.

b) Anyone with the URL can access the content. ✅

What is the first step in securing S3 content for CloudFront signed URLs?
a) Enable public access to S3 bucket.
b) Use IAM users to generate URLs.
c) Make the origin (S3) private and attach an Origin Access Control (OAC).
d) Configure Lambda@Edge for authentication.

c) Make the origin (S3) private and attach an Origin Access Control (OAC). ✅

In CloudFront, what does enabling "Require Signed URLs" ensure?
a) All traffic is encrypted with SSL.
b) Only requests with a valid signature/token can access content.
c) Requests are cached for a longer duration.
d) The origin server validates user credentials.

b) Only requests with a valid signature/token can access content. ✅

Which component must be stored in CloudFront as part of a trusted key group?
a) Private Key
b) Public Key
c) Session Token
d) Access Key ID

b) Public Key ✅