User -> cloudfront-> ALB -> EC2(nginx port 80) -> node apps port like 3000, 4000, 5000.

**ALB and Target Group Perspective**Great question! Understanding **why and how ports are exposed** for the **ALB and Target Group** is key to setting up your infrastructure correctly.

**🔁 Why We Expose Ports for ALB and Target Group**

**🔹 ALB Listener Port (e.g., 80 or 443)**

This is the **public-facing port** that users access in their browser.

* **Port 80** → for HTTP traffic
* **Port 443** → for HTTPS traffic (with SSL certificate)

💡 Think of this as the **entry gate** to your application.

**🔹 Target Group Port (e.g., 3000)**

This is the **port on your EC2 instance** (or container) where your app is actually running.

* In your case, your Node.js app is running inside a Docker container on **port 3000**.
* So, the **Target Group** must forward traffic to **port 3000** on the EC2 instance.

💡 This is the **destination port** where the ALB sends traffic after receiving it.

**🔁 How It Works Together**

Here’s a simplified flow:

User → ALB (port 80) → Target Group → EC2 (port 3000) → Docker Container (port 3000)

* ALB listens on **port 80**
* Forwards traffic to the **Target Group**
* Target Group sends traffic to **port 3000** on the EC2 instance
* EC2 forwards it to the **Docker container** (also listening on 3000)

**✅ Summary**

| **Component** | **Port** | **Purpose** |
| --- | --- | --- |
| ALB Listener | 80/443 | Public access (HTTP/HTTPS) |
| Target Group | 3000 | Internal app port on EC2 |
| Docker Container | 3000 | App running inside the container |

Would you like a visual diagram of this flow? Or help with setting up HTTPS on the ALB using ACM?

**While working on Cloudfront Perspective**  
You said:

with cloudfront distribution domain name with default behaviour I am able to access one node app with another behavior that is /legal/\* I am not able to access another second node app I attached alb as origin I am able to access this another second node app with alb DNS name:82/legal this is my nginx configuration server { listen 80; listen [::]:80; server\_name localhost; root /usr/share/nginx/html; location / { proxy\_pass # Forward requests to localhost:3000 proxy\_set\_header Host $host; # Preserve the original Host header proxy\_set\_header X-Real-IP $remote\_addr; # Forward the real IP address } location /gst { proxy\_pass # Forward requests to localhost:4000 proxy\_set\_header Host $host; # Preserve the original Host header proxy\_set\_header X-Real-IP $remote\_addr; # Forward the real IP address } } server { listen 82; listen [::]:82; server\_name localhost; root /usr/share/nginx/html; location /legal { proxy\_pass # Forward requests to localhost:4000 proxy\_set\_header Host $host; # Preserve the original Host header proxy\_set\_header X-Real-IP $remote\_addr; # Forward the real IP address rewrite ^/gst/(.\*)$ /$1 break; # Strip /gst from the path } } my first node app is working on port 3000 on amazon linux ec2 my second node app is working on port 5000 and I am using nginx where first node app is pointing to port 80 and second app is pointing to port 82 how to access second node app with cloudfront

**Copilot said:**

**Copilot**

Thanks for the detailed explanation! Based on your setup, here's a breakdown of the issue and how to fix it so that your second Node.js app (running on port 5000 and exposed via Nginx on port 82) is accessible via CloudFront using the /legal/\* behavior.

**✅ What’s Working**

* **First Node.js app** (port 3000) is accessible via CloudFront default behavior.
* **Second Node.js app** is accessible directly via ALB DNS name on port 82 using /legal.

**❌ What’s Not Working**

* CloudFront behavior for /legal/\* is **not** routing correctly to the second app.

**🔍 Likely Issues**

1. **CloudFront only supports HTTP/HTTPS on ports 80/443** — it **cannot** route to port 82 directly.
2. Your Nginx is listening on port 82 for /legal, but CloudFront can’t reach that port.
3. The rewrite rule in your Nginx config might be incorrect (rewrite ^/gst/(.\*)$ inside /legal block).

**✅ Recommended Fix**

**Option 1: Use a single Nginx server block on port 80**

Update your Nginx config so that both apps are accessible via port 80, which CloudFront can reach.

🔁 This way, both / and /legal/\* are served from port 80, and CloudFront can access them.

**🔧 CloudFront Configuration**

1. **Origin**: Use your ALB DNS name (which points to the EC2 with Nginx).
2. **Behaviors**:
   * **Default behavior**: Path pattern Default (\*), Origin: ALB, Forward all.
   * **Second behavior**: Path pattern /legal/\*, Origin: same ALB, Forward all.

Make sure to **forward all headers** and **disable caching** during testing.

**✅ Test URLs**

* https://your-cloudfront-domain/ → should go to Node app 1
* https://your-cloudfront-domain/legal/ → should go to Node app 2

Would you like help updating your Nginx config or CloudFront behavior settings step-by-step?