

Internet

Note: This diagram does not cover details like scaling and fault tolerance. For improved scalability and fault tolerance, the system can be deployed with kubernetes. Kubernetes enables automatic scaling, load balancing, and rolling updates etc. Additional security measures such as **mutual TLS** and **JWT validation** will be implemented and requests will be sent with jwt token for authentication .

lan

Written in C, responsible for proactively sending alarms data directly to the external server. But won't be able to receive requests from outside the LAN

ACAP Applications

Runs on

Runs on

Runs on

Runs on

Cameras/Speakers



http/https

http/https

http/https

http/https

LAN Server

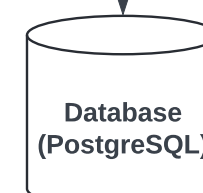
Can be a physical machine, responsible for forward requests, manage LAN devices, including requests to control device status like toggle active, scheduling, confidence adjustment, live video streaming, etc. Will require some way to store small amount of data, like public key for validating jwt token, scheduling configuration to perform the scheduling

Connect To

http/https with jwt token as authentication

Send alarm metadata, snapshot, camera identifier etc through requests to external server

Send alarm metadata, snapshot, camera identifier etc through requests to external server



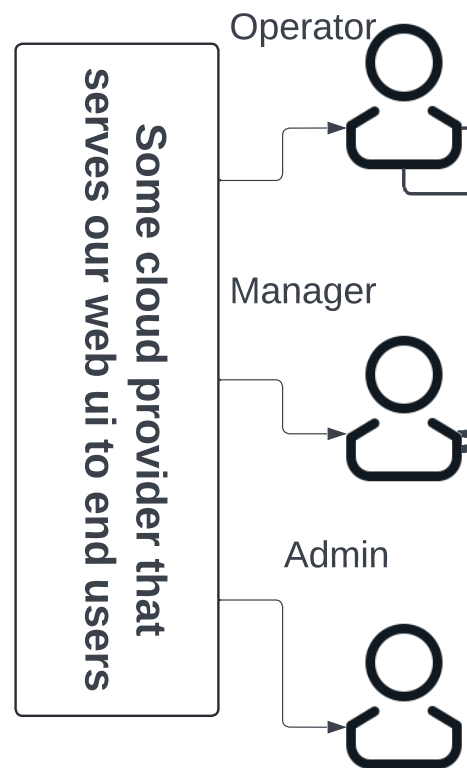
Database (PostgreSQL)

Connect to

External Server

handle authentication, user management, alarm management, communicate with database

Store Alarm metadata, user information, temporary video clips, etc



Request video live streaming

Manage Schedules

Confirm intruder/Cancel alarm

Retreive statistics data, web ui render data into charts/diagram

Control camera/system parameters, manage accounts

Some cloud provider that serves our web ui to end users