

Fresh-Soft-9303 • 2 dgn geleden

Nice concept. Here's a few cents from someone who worked in these areas:

1. Verification Node:

- ensure your data payload is not "heavy", try to process as much of the image locally, or else you risk impacting your broker's performance (also increase message size limits on mqtt if you like)

- you can combine the "Lock" with the verification node, makes your architecture simpler, there's no need to have more components when your R-Pi already has sufficient GPIOs. Yes, you'll need a driver and maybe a small battery, but the reduction in software coding across multiple languages, wiring, distributed architecture, etc. is worth (sometimes).

2) Hub

- Suggest moving the "facial recognition service" onto the verification node, or at least big chunks of it. MQTT isn't best used for that purpose. If your architecture is super necessary you can use BLE transfer (pending distance), and then as soon as you get the image trigger the processing on the hub.

3) Cloud Application / Google

- I would try and combine those as much as possible. Would even opt-in for serverless and be done with it.

👍 2 🗨 Beantwoorden 🏆 Prijs geven ➦ Delen ...

InFamouss01 OP • 2 dgn geleden

Thank you for your feedback.

1. The verification node is currently sending 360p images at 30fps to the hub for the kiosk_service, which runs an HTTP server to provide a live preview of the webcam. Additionally, it sends 1080p images every X seconds for the facial recognition. So far, I haven't noticed any performance issues, but I will keep this in mind.

I'm not sure if I could combine the lock and the verification node. The verification node is mounted outside, in front of the door, together with the pinpad and the camera (think of ring camera with a pinpad) . The lock itself is a lightweight microcontroller with a servo, running on a battery. Which will be placed inside the lock cylinder on the inside of the door, looking something like this:

<https://imgur.com/a/soVNNa5>

2. In my head, I wanted the system to be scalable for future upgrades (adding more node types). It felt more logical to make the hub the "brain" of the entire system, processing all data and making decisions (sending the request to open the lock). The nodes, would remain simple devices that either collect data or perform actions based on the hub's instructions.

Ideally, I want run the verification node off battery, but I haven't tested this yet, so I'm not sure if it will be viable.

👍 1 🗨 Beantwoorden 🏆 Prijs geven ➦ Delen ...

Fresh-Soft-9303 • 2 dgn geleden

You're welcome.

- Thanks for clarifying. HTTP servers are better to process data. From the image I thought you were sending a decoded image over MQTT and worried about bottlenecks. If you want to take it a step further you can run an ffmpeg client/server to stream this better. 30 fps is a lot, 15-20 fps should be reasonable (I have seen designs for casinos with like 20fps as minimum to detect slight-of-hand cheating), so if you're only doing facial recognition lower could be better.

- RE: lock combined with verification node, I get that design (thanks for explaining), makes sense to keep it distributed like that. Good for scalability.

- It will be challenging to run verification nodes on battery, you'll need to either recharge often or run wires. And... if you're running wires already you might want to reconsider the previous step of combining the lock/verification nodes together (through wiring - esp32 + motor will be close to the power source inside the home)

Good luck

👍 1 🗨 Beantwoorden 🏆 Prijs geven ➦ Delen ...



publiclyinnocent • 34 min. geleden

- What authentication method do you use for the REST API?
- How does a user authenticate when accessing the system on the local network?
- What fallback options exist if any of the components goes down?
- How do you manage firmware updates for the microcontrollers?

3 Beantwoorden Prijs geven Delen ...

Overwegingen & Feedback

- MQTT broker niet overladen met webcam feed payload
- Lock & Verification nodes samenvoegen (Geen optie)
- Facial recognition op de verification node uitvoeren (Geen MQTT transfers nodig)
- Serverless oplossing in plaats van een Cloud backend
- Ffmpeg client/server voor webcam streaming?
- 15-20 FPS (of nog minder) webcam feed zou voldoende moeten zijn.
- HUB -> Backend authenticatie onderzoeken/implementeren
- Hub -> Mobile App authenticatie onderzoeken / implementeren
- Fallback opties
- Microcontroller firmware updates pipeline