NAU NORTHERN ARIZONA UNIVERSITY

School of Informatics, Computing, and Cyber Systems

Site Locking and Alerting Mechanism for Remote Facilities (SLAM-Doors)

Team: Christopher Son, Aiden Seay, Preston Smith, Ryan Todd

Client: Benjamin Walker, **General Dynamics Mission Systems**, Scottsdale, AZ

Team Mentor: Bailey Hall

Motivation

General Dynamics Mission Systems supports the U.S. Coast Guard Rescue 21 network for real-time distress calls and mission coordination.

Current systems face:

- Unstable networks and **high latency**
- **Security vulnerabilities**/limited monitoring
- Power outages

SLAM-Doors delivers a **secure**, **resilient**, and **remotely managed** solution that operates even under extreme conditions.

GENERAL DYNAMICS Mission Systems

U.S. COAST GUARD RESCUE 21 SAVING LIVES IN THE 21st CENTURY

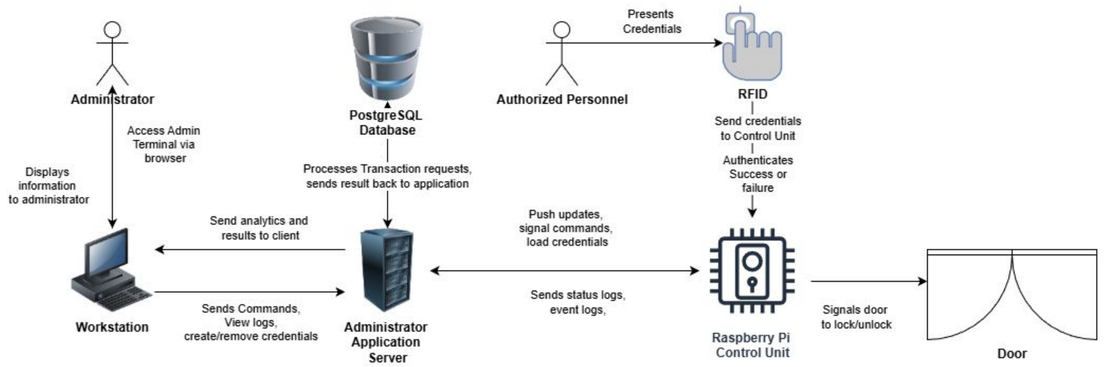
Plans, Goals, Stretch Goals

- **Plans:** Use AWS to host a server and database for the system and admin portal to **pull key information** and **post** it to the **admin portal**.
- **Goals:** Develop a **secure** system where personnel scan a **RFID card** to gain access to authorized sites and **automatically update** activity to the server.
- **Stretch Goals:** Implement **NFC** protocols to also allow access from **smartphones**.

Our Proposed Solution

Each door lock is controlled by a Raspberry Pi that hosts a lightweight server. This server verifies user access when an RFID chip is scanned and communicates securely with the central system. Using Docker ensures consistent deployments and simplifies remote updates without on-site maintenance.

On the server side, the backend is hosted by AWS, which manages all communication, authentication, and data storage. A PostgreSQL database stores user credentials and event logs. The frontend, built in React, provides operators with a realtime dashboard for monitoring lock status managing users, and reviewing system events.



Project Requirements

Intrusion Detection:
- Detect and report entry attempts
- Three states: no/auth/authorized
- Real-time operator notifications
- Live access status display
- Event logging for audits

Facility Integration:
- Intrusion detection at facilities
- Continuous sensor status reporting

Communication
- Operate during high latency
- Auto-recover after outages
- Safe remote software upgrades
- Live access status display
- Report entries via secure IP

Security:
- Protected operator authentication
- Block internal/external intrusion
- Encrypted TLS/HTTPS communication
- Secure SNMPv3 status reporting

Feasibility

Key Technologies:
- **Raspberry Pi:** This technology will act as the control unit for physical locations
- **Docker:** Every control unit will be running a Docker instance of Linux
- **GitHub:** Version control platform
- **AWS:** Provides infrastructure for the solution
- **React:** Responsive frontend framework
- **Node.JS:** Programming language used for backend development
- **PostgreSQL:** Database technology used to hold log data, users, credentials, and updates.

Planned Technologies

Raspberry Pi

docker

node JS

React

aws

PostgreSQL