Road Map to React Native Mobile App

DOP Side: DVCO Side:

HTTPS [POST]

Task Queue

DOP Gateway

Communicate through MQTT Protocol

Decrypt Message

Encrypt Message

Broker

Publisher

Subscriber

HTTPS [POST, GET]

HTTPS [POST, GET]

Proxy

Worker

Broker

Clients

**Requirements (Libraries):**

Axios (HTTPS [POST, GET])

Mqtt.js MQTT Protocol (Mosquitto broker)

React-Native-Websockets

Blowfish.ts (Blowfish ecb Algorithm)

Crc16 for integrity, base64 for encoding

Seed and pseudorandom generator for keys

**Idea for a simple mobile application:**

Flow of Requests:

1. Client Side (Sending HTTPS Request to DOP Gateway)

- Step 1: User enters message in the input field.

- Step 2: User taps "Send Request."

- Step 3: The app sends an HTTPS POST request to the DOP Gateway with the entered data.

- Step 4: DOP Gateway processes the request and responds back, displaying a success or error message on the app.

2. Pub/Sub Side (Communication via Broker and Proxy)

Publishing Flow:

- Step 1: User types a message in the input field and encrypt it then presses "Publish."

- Step 2: The app publishes the message to the broker using MQTT.

Subscribing Flow:

- Step 3: The app, as a subscriber, listens for message from the broker and decrypt it then displays them in a list.

Proxy Communication:

1. Message Generation (Pub/Sub)

- Step 1: The subscriber or publisher generates a message randomly.

2. Sending Message to Proxy

- Step 2: The app sends this generated message to the proxy via an HTTPS request.

3. Proxy Response

- Step 3: The proxy server processes the message and sends back a response.

- The app receives this response and displays it in a section labeled "Proxy Responses."

Example in App

We can see messages being logged in real-time as "Messages Sent to Proxy."

As responses come back, they appear in the "Proxy Responses" section.