**Step 1: AWS Configuration**

1. **Create AWS Account**: Make sure you have an AWS account. If you don't, create one at [AWS](https://aws.amazon.com).
2. **Create IAM Role and Policy**:
   * Go to the IAM (Identity and Access Management) console.
   * Create a new role with the necessary permissions to access AWS Chime SDK. Attach the AmazonChimeSDK and AmazonChimeSDKMediaPipelines policies to this role.
3. **AWS Chime SDK**:
   * Go to the [Amazon Chime SDK Console](https://console.aws.amazon.com/chime/home).
   * Create a new application. Note the application ID and region.
4. **AWS Lambda Functions (Optional)**:
   * If you need server-side logic for managing meetings, attendees, and signaling, set up Lambda functions.
   * Use AWS API Gateway to expose the Lambda functions as RESTful APIs.

**Step 2: Raspberry Pi Configuration**

1. **Set Up Raspberry Pi**:
   * Ensure your Raspberry Pi is set up with Raspbian OS. Update it using:

bash

Copy code

sudo apt update && sudo apt upgrade

1. **Install Node.js**:
   * You need Node.js to run the Chime SDK code. Install Node.js by running:

bash

Copy code

curl -fsSL https://deb.nodesource.com/setup\_16.x | sudo -E bash -

sudo apt-get install -y nodejs

1. **Install AWS CLI**:
   * Install AWS CLI to interact with AWS services.

bash

Copy code

sudo apt install awscli

aws configure

* + Provide your AWS credentials during configuration.

1. **Install Required Libraries**:
   * You will need libraries for handling media input/output:

bash

Copy code

sudo apt-get install libasound2-dev

**Step 3: Code to Connect AWS Chime SDK on Raspberry Pi**

1. **Create a Node.js Application**:
   * Set up your Node.js project:

bash

Copy code

mkdir chime-raspberry

cd chime-raspberry

npm init -y

npm install aws-sdk amazon-chime-sdk-js express

1. **Create a Server Application (server.js)**:
   * This server will manage meetings and attendees.

javascript

Copy code

const express = require('express');

const { ChimeSDKMeetingsClient, CreateMeetingCommand, CreateAttendeeCommand } = require('@aws-sdk/client-chime-sdk-meetings');

const app = express();

const port = 3000;

const chimeClient = new ChimeSDKMeetingsClient({ region: 'us-east-1' });

app.use(express.json());

app.post('/createMeeting', async (req, res) => {

try {

const meetingResponse = await chimeClient.send(new CreateMeetingCommand({ ClientRequestToken: `meeting-${Date.now()}` }));

res.json({ Meeting: meetingResponse.Meeting });

} catch (error) {

res.status(500).send(error.message);

}

});

app.post('/createAttendee', async (req, res) => {

try {

const { MeetingId, ExternalUserId } = req.body;

const attendeeResponse = await chimeClient.send(new CreateAttendeeCommand({ MeetingId, ExternalUserId }));

res.json({ Attendee: attendeeResponse.Attendee });

} catch (error) {

res.status(500).send(error.message);

}

});

app.listen(port, () => {

console.log(`Server running at http://localhost:${port}`);

});

1. **Run the Server**:
   * Start the server on Raspberry Pi:

bash

Copy code

node server.js

**Step 4: Connect Mobile Phone Browser**

1. **Create a Client Application (index.html)**:
   * This HTML file will connect to your Node.js server to join the Chime meeting.

html

Copy code

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Chime Meeting</title>

</head>

<body>

<button id="joinMeeting">Join Meeting</button>

<script>

document.getElementById('joinMeeting').addEventListener('click', async () => {

// Create Meeting

const meetingResponse = await fetch('http://<Raspberry\_Pi\_IP>:3000/createMeeting', { method: 'POST' });

const meeting = await meetingResponse.json();

// Create Attendee

const attendeeResponse = await fetch('http://<Raspberry\_Pi\_IP>:3000/createAttendee', {

method: 'POST',

headers: { 'Content-Type': 'application/json' },

body: JSON.stringify({ MeetingId: meeting.Meeting.MeetingId, ExternalUserId: 'user@domain.com' })

});

const attendee = await attendeeResponse.json();

console.log('Joined Meeting:', meeting, attendee);

});

</script>

</body>

</html>

1. **Access the HTML Page on Mobile**:
   * Host the index.html file using a simple HTTP server on the Raspberry Pi or upload it to a static website hosting service like AWS S3.
   * Access the page from your mobile browser and connect to the Raspberry Pi’s IP address.

**Step 5: Test and Debug**

* Make sure all AWS credentials are correctly set up.
* Ensure network configurations allow connections from your mobile device to the Raspberry Pi.
* Check permissions on your AWS account and roles to access the Chime SDK.