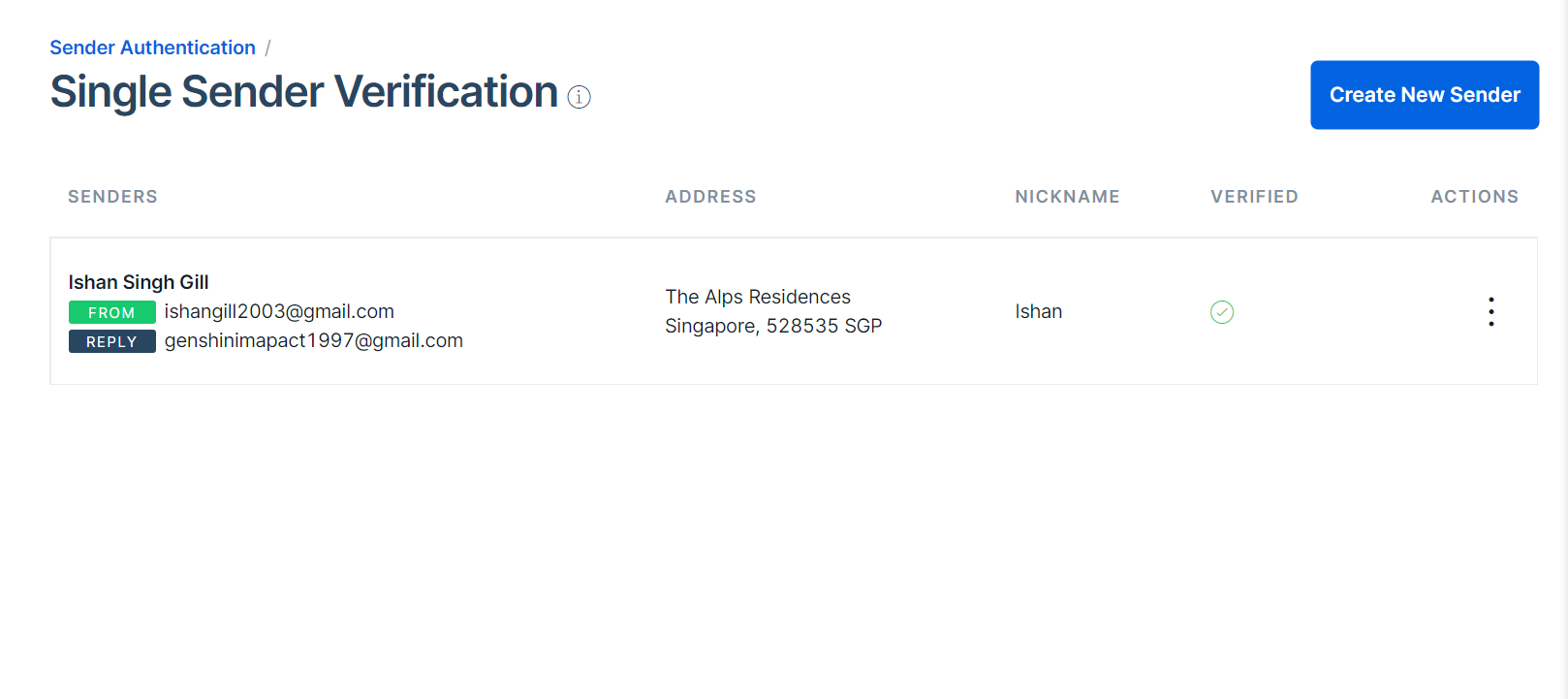
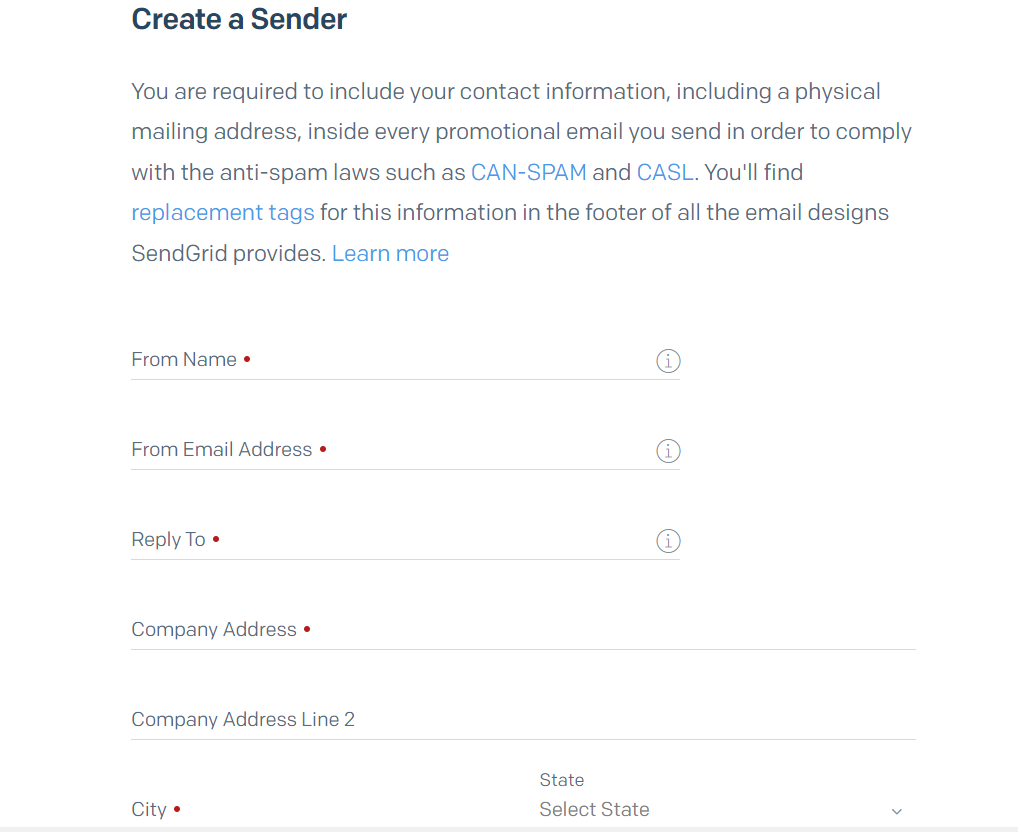
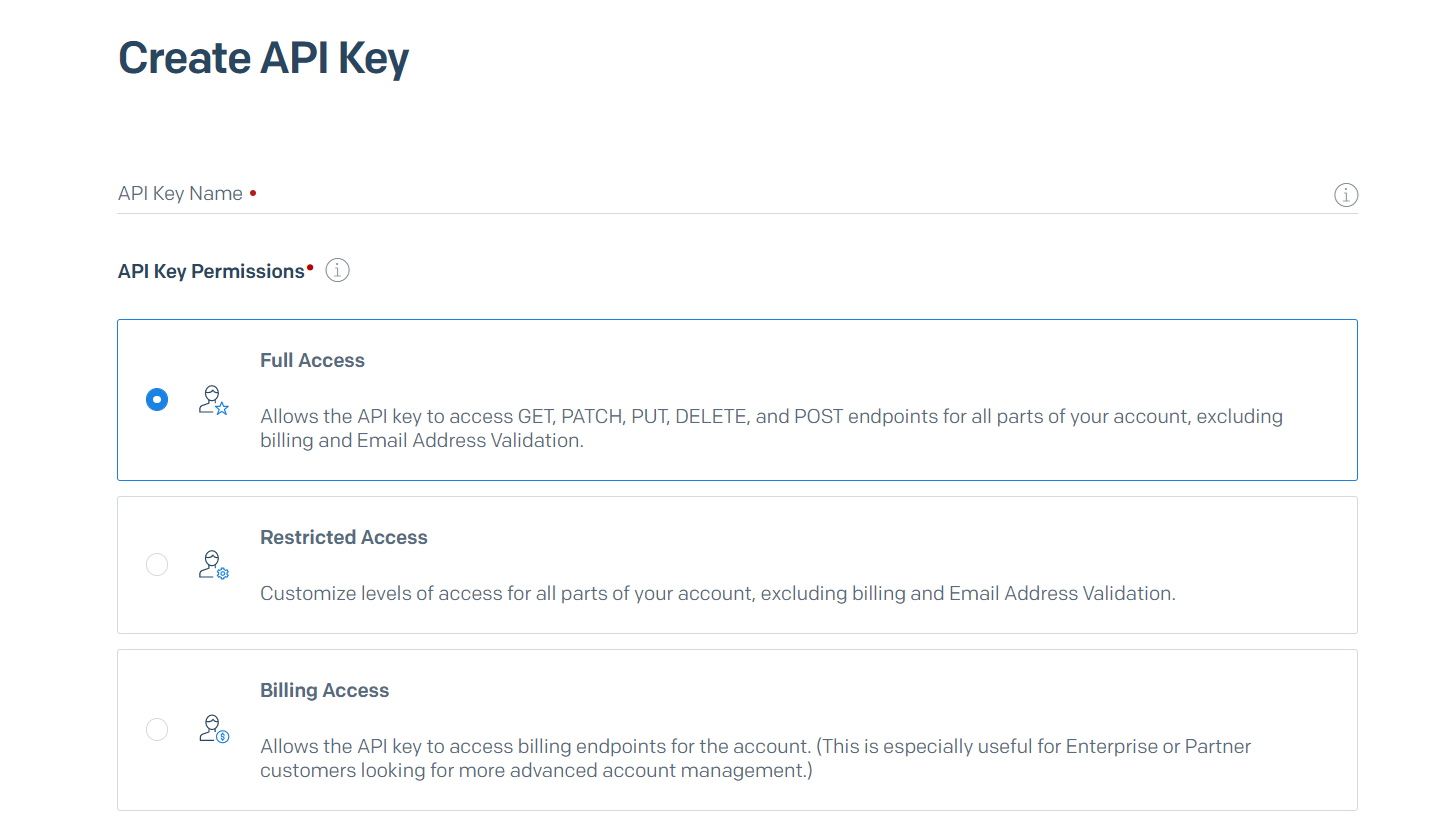
Sending a Test Email with the Gelato Network & SendGrid automatically every 25 seconds

* Set up a SendGrid Account (<https://app.sendgrid.com/login?redirect_to=%2F>)
  + Upon Login, Verify Single Sender Authentication under Settings by creating a new sender with your email address and relevant details

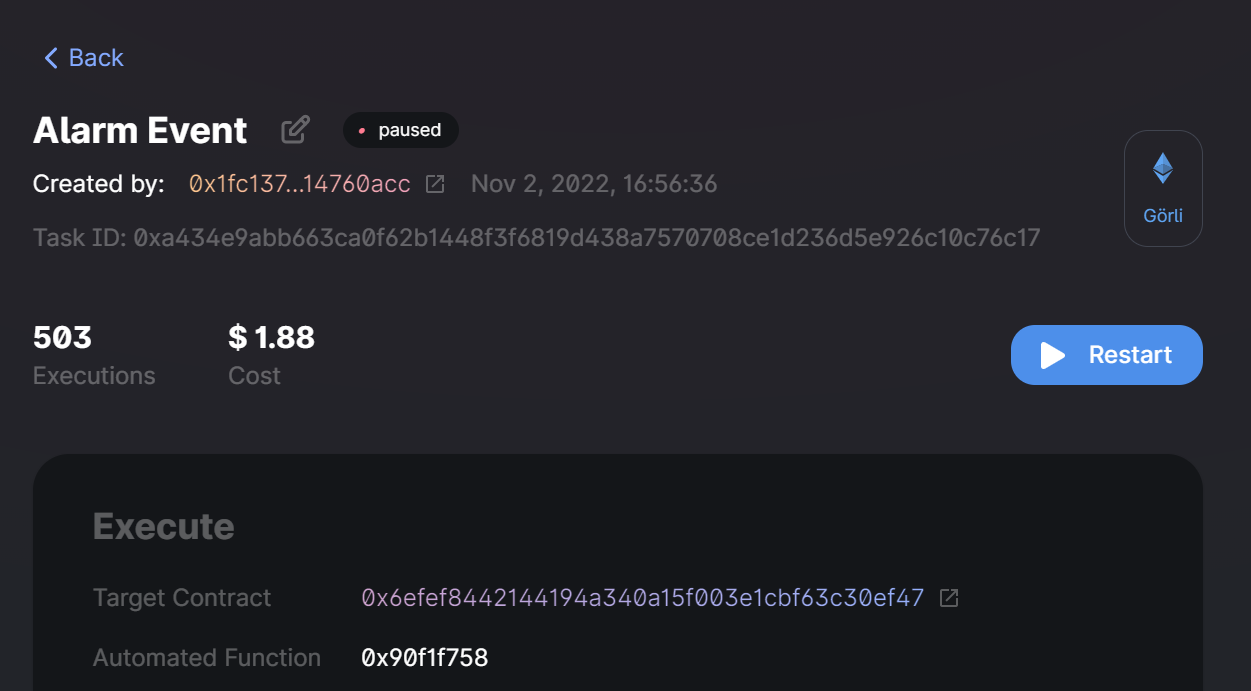




* + Create an API Key with full access and copy the key value



* Integrate with Node.js Environment + Gelato Network
  + Head to the Gelato Network and search for your task created previously and run it again (based of the Gelato Network Documentation) (Alarm Event)



* + Create a file called alarm.js and input the following backend codes along with your API Key from your SendGrid Account

const sgMail = require('@sendgrid/mail');

const API\_KEY =

'<API Key Value>';

sgMail.setApiKey(API\_KEY)

const { ethers } = require("hardhat");

// For Hardhat

const contract = require("../artifacts/contracts/Alarm.sol/Alarm.json");

//console.log(JSON.stringify(contract.abi));

// Provider

const alchemyProvider = new ethers.providers.AlchemyProvider(network="goerli", "jJopiviW6VYA5ngL2Z5gX45xhXwVTnRI");

// Signer

const signer = new ethers.Wallet("06b91f40bafe25bb97844ca576675d475ad4c197a4acab0f00a7dabf8326d480", alchemyProvider);

// Contract

const AlarmContract = new ethers.Contract("<Alarm Event’s target contract>", contract.abi, signer);

async function main() {

    abi = ["event Plus(uint plus)"];

    AlarmContract.on('Plus', (plus) => {

      console.log("Live Value:", plus.toNumber());

      if (plus != 25) {

        const message = {

          to: '<Secondary Email Address>',

          from: {

              name: 'Enterprise Insurance Agency',

              email: '<Primary Email Address>'

          },

          subject: 'Insurance Premium Payment Reminder',

          body: 'Insurance Premium Payment Reminder',

          html: '<h3>Please remember to pay your Insurance Premium</h3>',

      };

      sgMail

      .send(message)

      .then((response) => console.log('Email sent...'))

      .catch((error) => console.log(error.message));

      }

  })

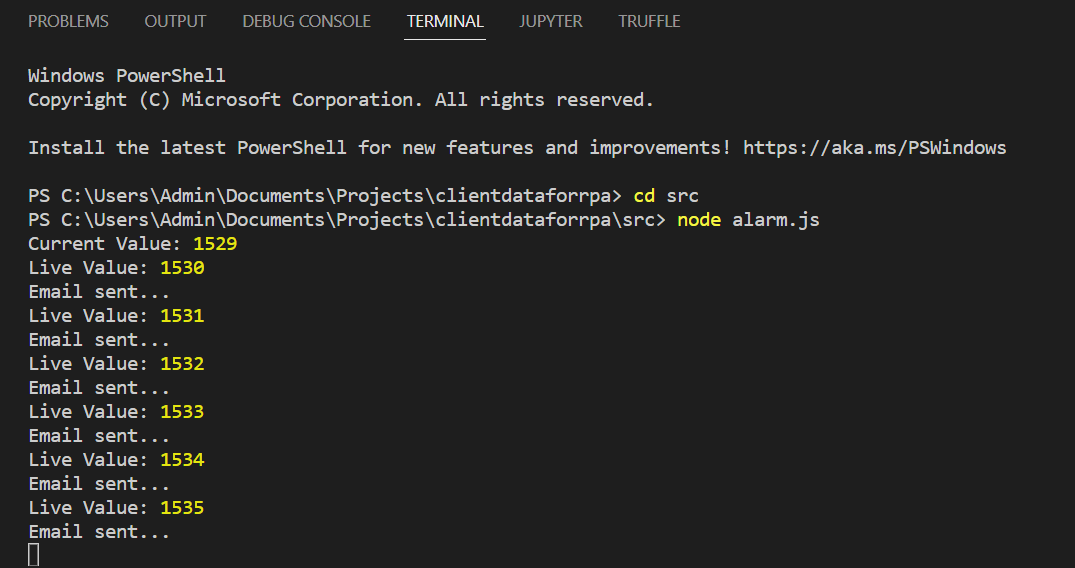
  const value = await AlarmContract.getInput();

  console.log("Current Value:", value.toNumber())

}

  main();

* Run the alarm.js file on console (a test email would be sent after each event)



-------------------------------------------------------------------------------------------------------

