

View online

Download PDF

Workshop: The bees are important: use SDKs wisely

Paolo Insogna

Node.js TSC, Principal Engineer @ **Platformatic**

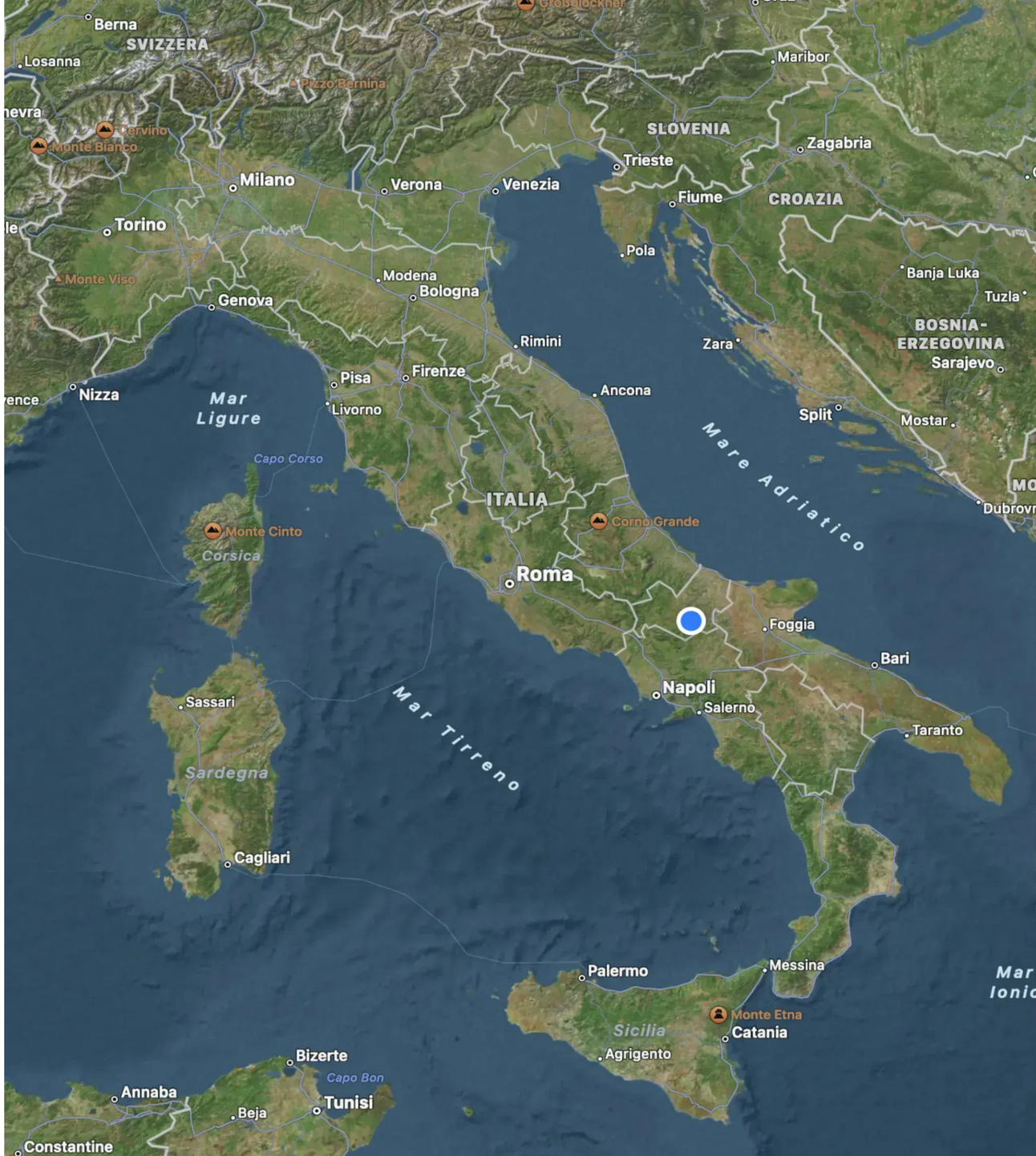
Hello, I'm **Paolo**!



Node.js Technical Steering Committee Member

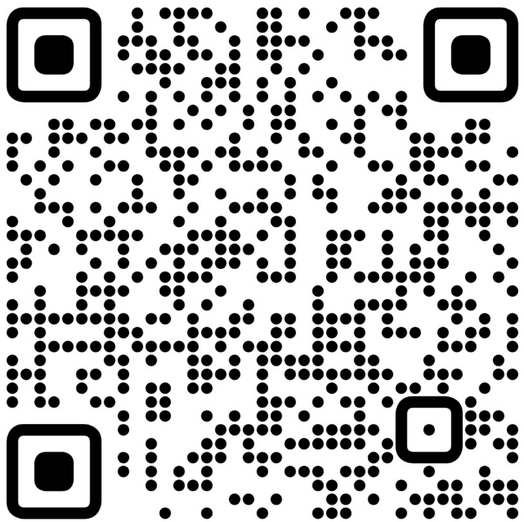
Platformatic Principal Engineer

`paoloinsogna.dev` `ShogunPanda` `p_insogna` `pinsogna`



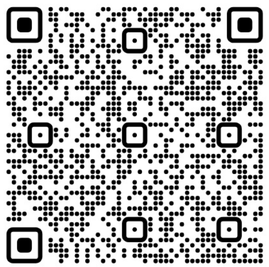
What are we talking about?

Let's start with a brief introduction.



https://talks.paoloinsogna.dev/bees

The details lie in the documentation



AWS S3 PutObject REST API

Documentation



AWS Signature Version 4

Calculations for the Authorization Header

Let's get to the action!



Setup

No dependencies in **`package.json`**, Node.js already has our backs ...

```
{
  "name": "direct",
  "version": "1.0.0",
  "main": "index.js",
  "type": "module",
  "scripts": {},
  "dependencies": {}
}
```

The skeleton

```
1   import { createHash, createHmac } from 'node:crypto'
2   import { readFile } from 'node:fs/promises'
3
4   const awsS3UnsignedPayload = 'UNSIGNED-PAYLOAD'
5
6   function sha256(contents) {
7     return createHash('sha256').update(contents).digest('hex')
8   }
9
10  function hmacSha256(key, contents) {
11    return createHmac('sha256', key).update(contents).digest()
12  }
13
14  // Implements the Authenticating Requests (AWS Signature Version 4) algorithm.
15  function signS3Upload(keyId, accessKey, region, method, rawUrl, headers, payloadHash) {
16    // CanonicalRequest calculation
17    // StringToSign calculation
18    // Signature calculation
19  }
```

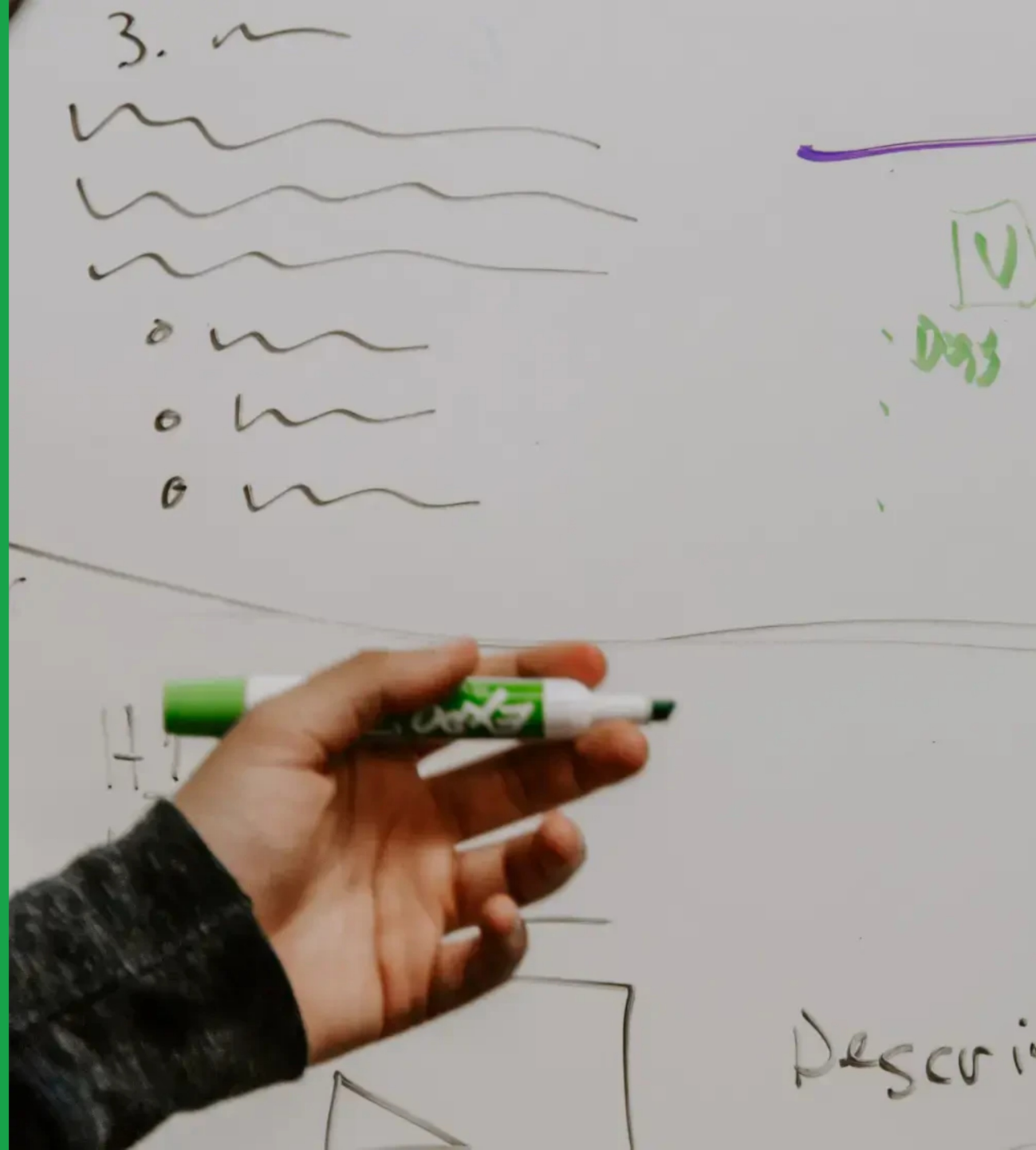
Task 1: Create a Canonical Request

Enjoy! 😈



Task 2: Create a String to Sign

Enjoy! 😂



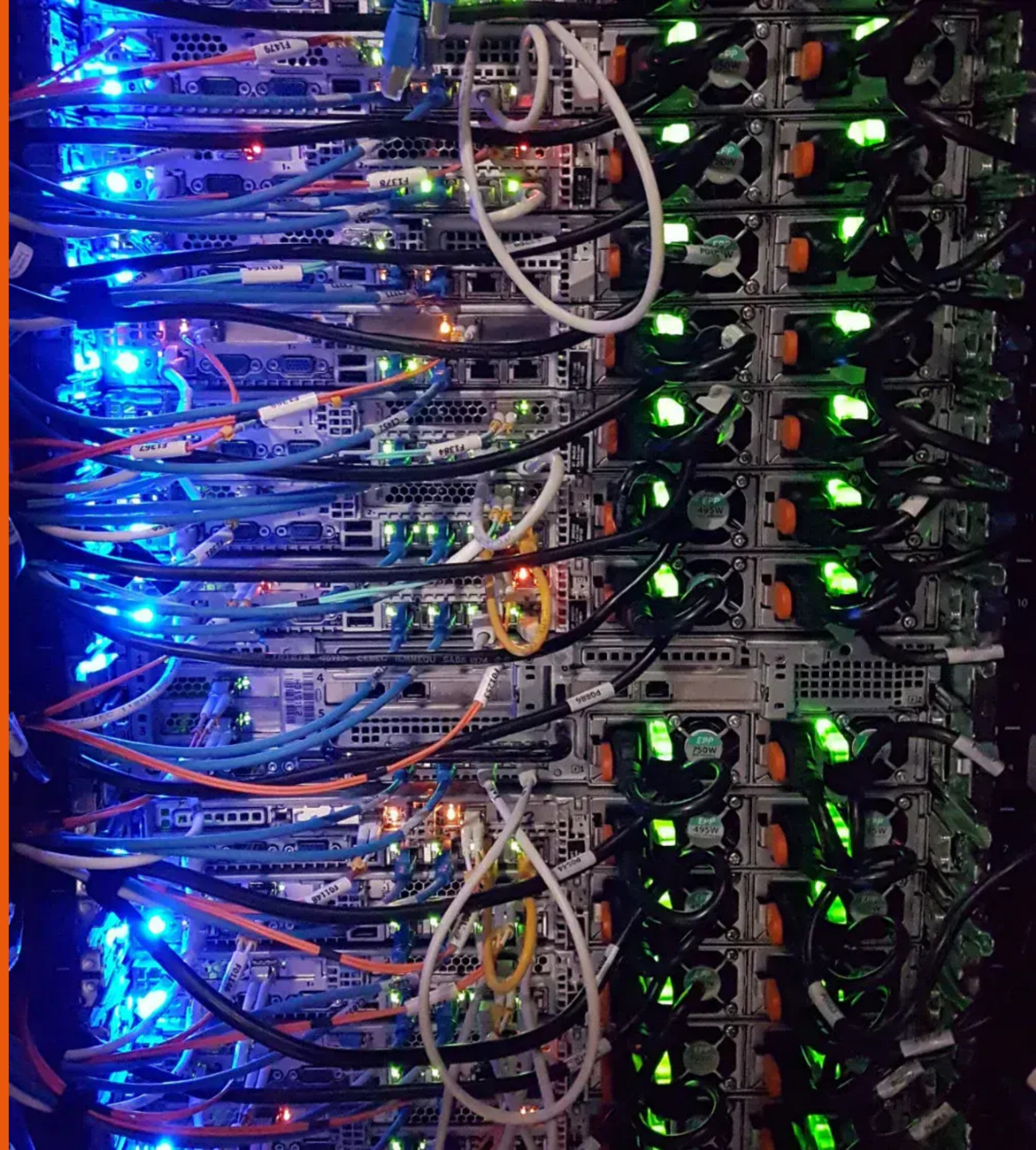
Task 3: Calculate Signature

Enjoy! 😶



Task 4: REST API call

Enjoy! 😎



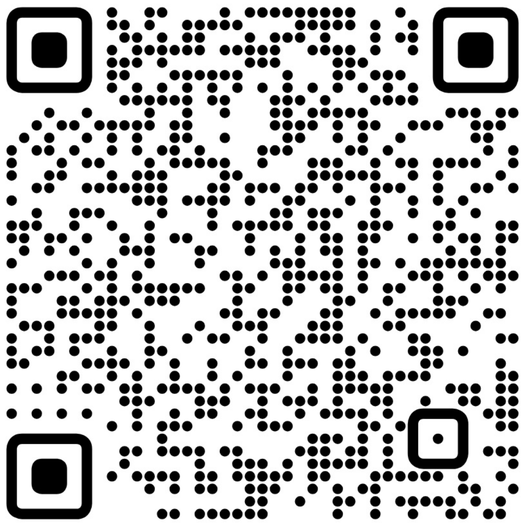
Task 5: Main function

Enjoy! 🍺



Take home

You can find all the code on GitHub:



https://github.com/ShogunPanda/workshops-bees



Thank you!

Paolo Insogna

Node.js TSC, Principal Engineer

@p_insogna

paolo.insogna@platformatic.dev

Platformatic