ENQODE

Full Stack Developer - Assignment

Objectives

- **JavaScript:** Analyze and explain the output of a given JavaScript code snippet.
- **Build a Darkweb Scanner API**: Create a cURL-based API using a service like IntelX or native command-line tools for system interaction.
- **Create a "Zero-Trust Score" Widget**: Develop a Vue.js component that displays a company's Zero-Trust Score based on provided metrics.
- **Bonus Task**: Implement a backend service to collect data from the Zero-Trust Score JSON.

Assignment Details

Part 1: JavaScript

Question: What will the following JavaScript code output to the console?

```javascript
const array = [12, 10, 22, 5, 25];
for(var i = 0; i < array.length; i++){
    setTimeout(function(){
        console.log("The element in position " + i + " is: " + array[i]);
    }, 5000);
}
```

Part 2: Darkweb Scanner API

Requirements:

- Research a suitable API provider (e.g., IntelX), NPM package, or use command-line tools like OpenSSL and nslookup.
- Implement a cURL-based API that can query dark web data relevant to a specified domain or company.
- The API should return structured data that can be easily integrated into your application.
- Test your API using Postman or a similar tool.

Deliverables:

- A clear and concise README file explaining how to set up and use the API.
- Code demonstrating your API implementation.

Part 3: Zero-Trust Score Widget

Requirements:

- Using Vue.js 3.x and Vuetify, build a widget that visually represents the Zero-Trust Score from the provided JSON structure.
- The widget should include:
  - A display of the overall Zero-Trust Score.
  - Individual scores for each metric, along with descriptions.
  - A visual representation (e.g., progress bars) of each metric based on its score.
  - A risk category indicator.

Sample JSON Structure:

```
{
  "companyName": "FinTechSecure Ltd.",
  "ZeroTrustScore": 58.5,
  "metrics": {
    ...
  },
```

```
Unset
  "observableData": {
      "averageShannon

ShannonEntropyScore": 7.8,
      "firewallDetected": true,
      "DNSsecEnabled": true,
      "tlsVersion": "1.2",
      "certificateBitStrength": 2048,
      "securityHeadersImplemented": ["X-XSS-Protection", "X-Frame-Options"],
      "openPortsDetected": 12
    }
```

```
  "riskCategory": "Moderate Risk"
}
```

Deliverables:

- Vue.js component code, runs on Vuetify - play.vuetifyjs.com
- A demo or screenshot of the widget in action.
- A README explaining how to integrate and use the widget in a Vue application.

Part 4: Bonus - Data Collection Backend

Requirements:

- Build a backend using Node.js, Python, TypeScript, or JavaScript to collect the Zero-Trust Score data below:

```json
{

  "averageShannon

ShannonEntropyScore": 7.8,
  "firewallDetected": true,
  "DNSsecEnabled": true,
  "tlsVersion": "1.2",
  "certificateBitStrength": 2048,
  "securityHeadersImplemented": ["X-XSS-Protection", "X-Frame-Options"],
  "openPortsDetected": 12
}
```

- The backend should be capable of receiving the JSON data and storing it for future use (e.g., in a database or a file).
- Provide endpoints to:
  - Submit Zero-Trust Score data.
  - Retrieve stored data.

Deliverables:

- Backend service code.
- A README explaining how to set up and use the backend.
- Documentation for the API endpoints.

Submission Guidelines

- Your submission should be in a GitHub repository or a similar platform, with clear instructions on how to set up and run each component.
- Ensure that your code is well-organized and commented, adhering to industry best practices.
- Aim for clarity and maintainability in your code and documentation.

Deadline

Please complete and submit your assignment within one week of receiving this document.

Feel free to reach out if you have any questions or need clarifications. We look forward to seeing your work!