

個人狀態頁面

看起來你安裝的 `vnstat` 版本不支援 `-u` 參數。讓我們按照正確的步驟來配置 `vnstat` 並設置帶寬監控 API。

步驟 1：配置 `vnstat`

為你的網絡接口初始化 `vnstat`：

```
sudo vnstat -i eth0
```

步驟 2：等待數據收集

`vnstat` 需要時間來收集數據。定期檢查狀態：

```
sudo vnstat -l
```

一段時間後，驗證數據收集：

```
sudo vnstat -d
```

步驟 3：創建 API 以暴露帶寬數據

安裝 Flask：

```
pip install Flask
```

創建一個 Python 腳本 (`bandwidth_api.py`)：

```
from flask import Flask, jsonify
from flask_cors import CORS
import subprocess

app = Flask(__name__)
CORS(app) # 為所有路由啟用 CORS

@app.route('/bandwidth', methods=['GET'])
```

```
def get_bandwidth():
    # 運行 vnstat 命令以獲取 eth0 的 5 分鐘間隔流量統計數據
    result = subprocess.run(['vnstat', '-i', 'eth0', '-5', '--json'], capture_output=True, text=True)
    data = result.stdout

    # 將捕獲的數據作為 JSON 響應返回
    return jsonify(data)

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000)
```

運行腳本：

```
python bandwidth_api.py
```

步驟 4：與你的博客集成

使用以下 HTML 和 JavaScript 來獲取並顯示帶寬數據：

```
document.addEventListener('DOMContentLoaded', function () {
    fetch('https://www.lzwjava.xyz/bandwidth')
        .then(response => response.json())
        .then(data => {
            var bandwidthData = JSON.parse(data);

            // 創建一個時間容器
            var timesContainer = document.createElement('div');

            var currentUtcTime = new Date();
            var currentLocalTime = new Date(currentUtcTime.getTime());

            var pUtcTime = document.createElement('p');
            pUtcTime.textContent = `UTC 時間: ${currentUtcTime.toUTCString()}`;
            timesContainer.appendChild(pUtcTime);

            var pLocalTime = document.createElement('p');
            pLocalTime.textContent = `我的本地時間: ${currentLocalTime.toString()}`;
```

```

timesContainer.appendChild(pLocalTime);

// 將時間容器附加到狀態 div
document.getElementById('status').appendChild(timesContainer);

// 創建一個表格
var table = document.createElement('table');
table.border = '1';
table.style.borderCollapse = 'collapse';
table.style.width = '100%';

// 創建表格頭部
var thead = document.createElement('thead');
var tr = document.createElement('tr');
var headers = ['時間', '流量 (KB/s)', '狀態'];
headers.forEach(headerText => {
    var th = document.createElement('th');
    th.textContent = headerText;
    tr.appendChild(th);
});
thead.appendChild(tr);
table.appendChild(thead);

// 創建表格主體
var tbody = document.createElement('tbody');

// 處理流量數據
var fiveMinuteData = bandwidthData.interfaces[0].traffic.fiveminute.reverse();
fiveMinuteData.forEach(interval => {
    var tr = document.createElement('tr');

    var dateTime = new Date(Date.UTC(interval.date.year, interval.date.month - 1, interval.date.day, interval.date.hour, interval.date.minute, interval.date.second));
    var timeDifference = Math.round((currentUtcTime - dateTime) / (1000 * 60)); // 時間差 (分鐘)

    var tdTimeDifference = document.createElement('td');
    tdTimeDifference.textContent = timeDifference + ' 分鐘前';

```

```

tr.appendChild(tdTimeDifference);

var averageTraffic = (interval.rx + interval.tx) / 2; // 計算 RX 和 TX 的平均值
var tdTrafficKBs = document.createElement('td');
var trafficKBs = (averageTraffic / (5 * 60 * 1024)).toFixed(2); // 轉換為 KB/s
tdTrafficKBs.textContent = trafficKBs;
tr.appendChild(tdTrafficKBs);

var tdStatus = document.createElement('td');
tdStatus.textContent = trafficKBs > 5 ? '在線' : '離線';
tdStatus.className = trafficKBs > 5 ? 'status-online' : 'status-offline';
tr.appendChild(tdStatus);

tbody.appendChild(tr);
});
table.appendChild(tbody);

// 將表格附加到狀態 div
document.getElementById('status').appendChild(table);
})
.catch(error => {
  console.error('獲取帶寬數據時出錯:', error);
});
});

```

將 `http://your-droplet-ip:5000/bandwidth` 替換為你的 droplet 的 IP 地址。

其他考慮

- **安全性**：確保你的 API 是安全的。考慮添加身份驗證。
- **性能**：監控帶寬可能會消耗大量資源。確保你的 droplet 有足夠的資源。
- **可靠性**：添加錯誤處理和重試邏輯以處理 API 不可用的情況。

通過遵循這些步驟，你可以在博客上創建一個狀態頁面，根據你的 DigitalOcean droplet 的帶寬使用情況來指示你是否在線。