**修改紀錄**

|  |  |
| --- | --- |
| 修改項目 | 修改日期 |
| 新增私有鏈架設 | 5/26 |
| 新增私有鏈介面 | 5/27 |
| 刪除私有鏈架設、介面 | 5/30 |
| 新增區塊鏈上傳、下載架構 | 5/30 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

目錄

[Blockchain Upload Architecture 3](#_Toc41765211)

[Server、Edge Server 3](#_Toc41765212)

[Node.js 3](#_Toc41765213)

[BlockChain 4](#_Toc41765214)

[交易內容參數詳解 4](#_Toc41765215)

[Blockchain Upload Architecture 4](#_Toc41765216)

[Server、Edge Server說明： 4](#_Toc41765217)

[Node.js說明： 4](#_Toc41765218)

[BlockChain說明： 5](#_Toc41765219)

[Blockchain Download Architecture 6](#_Toc41765220)

[Blockchain Download Architecture 6](#_Toc41765221)

[Server、Edge Server 6](#_Toc41765222)

[Node.js 6](#_Toc41765223)

[BlockChain 7](#_Toc41765224)

[Blockchain Download Architecture 7](#_Toc41765225)

[Edge Server說明： 7](#_Toc41765226)

[Node.js說明： 7](#_Toc41765227)

[BlockChain說明： 8](#_Toc41765228)

# Blockchain Upload Architecture

## Server、Edge Server

主伺服器、邊緣端，將要上傳至區塊鏈的資料提供給Node.js

|  |  |  |
| --- | --- | --- |
| 欄位類型 | 欄位名稱 | 用途/詳解 |
| File | User Request | .txt檔，內容存放者使用者要調閱的時段 |
| File | Metadata.js | 每台Edge有一個至數個，存放內容為Metadata，用於上傳至區塊鏈保護 |
| file | Transaction Hash | .txt檔，存放每筆上傳Metadata至區塊鏈時，產生的交易Hash |
| var | fs | require(“fs”)，可讀取指定的檔案內容、生成檔案並寫入資料 |

## Node.js

利用Node.js的web3模組可將本地環境與區塊鏈環境連通

|  |  |  |
| --- | --- | --- |
| 欄位類型 | 欄位名稱 | 用途/詳解 |
| var | web3 | require(“web3”) |

## BlockChain

|  |  |  |
| --- | --- | --- |
| 欄位類型 | 欄位名稱 | 用途/詳解 |
| 參數 | eth.sendTransaction | JavaScript-Eth語法，用於向以太坊發起交易，可附帶data資料 |

## 交易內容參數詳解

透過JavaScript命令列模式檢視指定的區塊交易資訊

# Blockchain Upload Architecture

## Server、Edge Server說明：

初始化指令。

|  |  |
| --- | --- |
| **Server、Edge Server** | |
| 輸入 | 指定目錄內的.txt資料 |
| 輸出 | 資料內容輸出成.js檔並執行 |
| 參數 | var fs=require("fs") |
| 方法 | 透過nodejs的fs模組讀取指定目錄內的檔案資料 |

## Node.js說明：

|  |  |
| --- | --- |
| **Node.js** | |
| 輸入 | 1. Server、Edge Server 提供需要上傳的資料 2. 發起交易後產生的Transaction Hash(Hash以16進制表示，長度恆為256Bits，由區塊鏈提供) |
| 輸出 | 1. 將資料以16進制上傳至區塊鏈 2. 以.txt檔的形式存放至Server或Edge Server的指定資料夾內 |
| 參數 | 1. var Web3 = require("web3") var web3 = new Web3() web3.setProvider(web3.providers.HttpProvider("http://localhost:8545")); 2. var fs=require("fs") |
| 方法 | 透過web3模組，連通本地與區塊鏈環境，可直接與區塊鏈做溝通。 |

## BlockChain說明：

|  |  |
| --- | --- |
| **BlockChain** | |
| 輸入 | 由web3連通區塊鏈接點發起新的交易(欲上傳的資料(以16進制表示)) |
| 輸出 | 產生Transaction Hash(Hash以16進制表示，長度恆為256Bits) |
| 參數 | eth.sendTransaction({})  -JavaScript-Eth語法，用於向以太坊發起交易，可附帶data資料 |
| 方法 | 利用eth.sendTransaction，將資料逐筆上傳至區塊鏈，交易被打包後，會產生出該筆交易的Transaction Hash |

# Blockchain Download Architecture

# Blockchain Download Architecture

## Server、Edge Server

主伺服器、邊緣端，將Transaction hash提供給Node.js

|  |  |  |
| --- | --- | --- |
| 欄位類型 | 欄位名稱 | 用途/詳解 |
| file | Transaction Hash | .txt檔，存放每筆上傳Metadata至區塊鏈時，產生的交易hash |
| File | Metadata.txt | 每台Edge有一個至數個，存放內容為Metadata，用於上傳至區塊鏈保護 |
| var | fs | require(“fs”)，可讀取指定的檔案內容、生成檔案並寫入資料 |

## Node.js

利用Node.js的web3模組可將本地環境與區塊鏈環境連通

|  |  |  |
| --- | --- | --- |
| 欄位類型 | 欄位名稱 | 用途/詳解 |
| var | web3 | require(“web3”) |
| String[] | data | 存放目錄底下所有Transaction Hash |

## BlockChain

|  |  |  |
| --- | --- | --- |
| 欄位類型 | 欄位名稱 | 用途/詳解 |
|  |  |  |

# Blockchain Download Architecture

## Edge Server說明：

初始化指令。

|  |  |
| --- | --- |
| **Edge Server** | |
| 輸入 | 從Server獲取調閱要求 |
| 輸出 | 輸出Transaction Hash(Hash以16進制表示，長度恆為256Bits)給Node.js |
| 參數 | 無 |
| 方法 | 透過nodejs的fs模組讀取指定目錄內的檔案資料 |

## Node.js說明：

|  |  |
| --- | --- |
| **Node.js** | |
| 輸入 | 由Edge Server提供Transaction Hash |
| 輸出 | 輸出Transaction Hash(Hash以16進制表示，長度恆為256Bits)至區塊鏈 |
| 參數 | 1. var Web3 = require("web3") var web3 = new Web3() web3.setProvider(web3.providers.HttpProvider("http://localhost:8545")); 2. web3.eth.getTransaction(“transaction hash”).input -JavaScript-Eth語法，利用Transaction hash，檢視該筆交易的input(附帶資料，也就是Metadata(Hex)) |
| 方法 | 透過web3模組，連通本地與區塊鏈環境，可直接與區塊鏈做溝通。 |

## BlockChain說明：

|  |  |
| --- | --- |
| **BlockChain** | |
| 輸入 | eth.getTransaction(“transaction hash”).input |
| 輸出 | 顯示Metadata(Hex) |
| 參數 | 無 |
| 方法 | 無 |