

前端開發指引

2022/06/27 by Wade

名詞解釋

- **Q8 Vista** 這是這套軟體完整的名稱, 包含底下部份
 - 控制台: 由 q8server + q8webui 組成
 - q8server.exe 這是後台, 執行在 windows server 上
 - q8webui 這是前端, 網頁放在 www/ 目錄下, 以 www/ 表示
 - q8s.db 這是資料庫, sqlite3 格式
 - terminal 這是公司產品, 利用 q8server + Q8webui 的設定開機並使用 RDS 提供的應用程式
 - RDS: 這是 Windows Server 的功能之一, 提供 terminal 遠端桌面及應用程式的伺服器
 - ADS 這是 windows server 的功能之一, 進行網域內帳號管理
- **Installer** 安裝程式, 一般開發測試不需要透過 Installer
- **q8server** 在 windows 執行的後台, 搭配前端 www/ 執行控制台功能。q8server 由 golang 寫成, 主要考量是效率, golang 套件都是 open source, 套件或文件資源取得容易
- **terminal** 即公司的產品, 為一種無硬碟的精簡型電腦, 其開機是由 q8server 負責傳送開機系統檔案, 包括 OS 及 firmware package
- **www/** 後面的斜線表示目錄(資料夾), 此即前端編譯後產生的目錄, 大部份時間以 Webui 稱呼, 安裝後以 arista-[www.fs](#) 存在
- **Webui** 即控制台的前端, 以 API 與 q8server 後台溝通, 其網頁檔在 www/ 或 arista-[www.fs](#)
- API 是前端與後台溝通界面, 即 url, 溝通時以 JSON 交換資料
- Q8Path 此即執行 q8server 所在目錄
- DocPath 此即資料庫存放位置

- tid 為 terminal id, 其他還有 app id, rds id 等, 識別 terminal
- terminal status 有各種狀態: BOOTING CREATING PENDING ACTIVE OFF ERROR
- ~ 符號表示使用者的家目錄, 例如 C:\Users\Administrator
- RDS 這是 windows server 提供的遠端桌面環境
- ADS 一般是每個應用程式同時只能提供一個帳戶讀取, 所以如果同一個應用程式需要供多個 terminals 使用, 需要 ADS 進行多帳號管理
- virtual machine 因為無法讓開發團隊或個人每個人都有 windows server 環境, 通常採用虛擬機, 但是對 windows 開發者而言, 這個是非必須的。
- 遠端桌面, 通常為了連到美國查問題, 會以 team viewer 進行遠端作業
- ssh 在遠端要連回公司, 採用遠端桌面相當耗資源, 通常我們都用 ssh 的方式連回公司進行遠端開發與檔案存取
- git 進行版本控制, git for windows 還提供 bash 環境, 相當於一個簡易的 Linux 環境, 與其他開發者比較接近。
- OS 由 q8server 傳給 terminal 進行開機用, 它是 embedded linux, 由 buildroot 建構而成
- firmware package 簡稱 fp, 系統開機後, 會取得各種工具, 集成 fp, 主體也是利用 buildroot 環境, 大部份是 python 或 bash 程式

修改網頁及測試

1. 修改網頁一般用 [visual studio code](#)
2. 版本控制一律使用 [git](#), 並請只加入修改的檔案, 嚴禁 `git add .` 或 `git add *`
3. 測試時要模擬 terminal, 請參考「模擬 terminal」
4. 前端除錯需要搭配瀏覽器的 Console 及 React 擴充套件, 建議採用 [Firefox](#) 或 [Chrome](#)
5. 如果有需要, 請以命令列方式執行 q8server, 參考「q8server 執行方式」, 可以看到更多後台的資訊。必要時修改 q8server 以利除錯。
6. 在 windows service時 log 在『[事件簿](#)』中, 在 Linux 執行的話log在命令視窗中
7. 修改網頁最重要的是編譯, 請參考「React 編譯環境」
8. **因為沒有實際的 RDS 及 terminal, 因此 WebUI 建立的資料可以隨便亂打**
9. **git commit 只加修改的檔, 嚴禁盲目全加, 每次 commit 要提供註解**

React 編譯環境

1. 最重要的是 NodeJS
 - a. 透過 [npm](#) 進行 nodejs 的套件管理
 - b. Q8 Vista 並非使用最新的 NodeJS, 一般以 [nvm 進行 npm 的版本管理](#)
2. Node 使用 14, 詳情參考右圖
 - a. 以 nvm 安裝時: `nvm i 14.19` 或 `nvm i 14` 或 `nvm i 14.19.0`
3. 編譯指令應該是 `npm run build`
4. 原開發者有提供 Mock Server, 如果要執行假資料驗證的話, 應該是 `npm run mock-server` 或 `npm run dev` 或 `npm run dev2` 之所以是『或』是因為不鼓勵所以我也沒研究
5. 編譯完應該會產生 q8server/www
 - a. 一種方式是將 www/ 複製到 q8server.exe 所在地
 - b. 另一種方式是將 q8server.exe 複製到 q8server/
 - c. 以上兩種方式的精神是相同的

```
$ npm version
{
  q8webui: '0.0.1',
  npm: '6.14.16',
  ares: '1.18.1',
  brotli: '1.0.9',
  cldr: '40.0',
  icu: '70.1',
  llhttp: '2.1.4',
  modules: '83',
  napi: '8',
  nghttp2: '1.42.0',
  node: '14.19.0',
  openssl: '1.1.1m',
  tz: '2021a3',
  unicode: '14.0',
  uv: '1.42.0',
  v8: '8.4.371.23-node.85',
  zlib: '1.2.11'
}
```

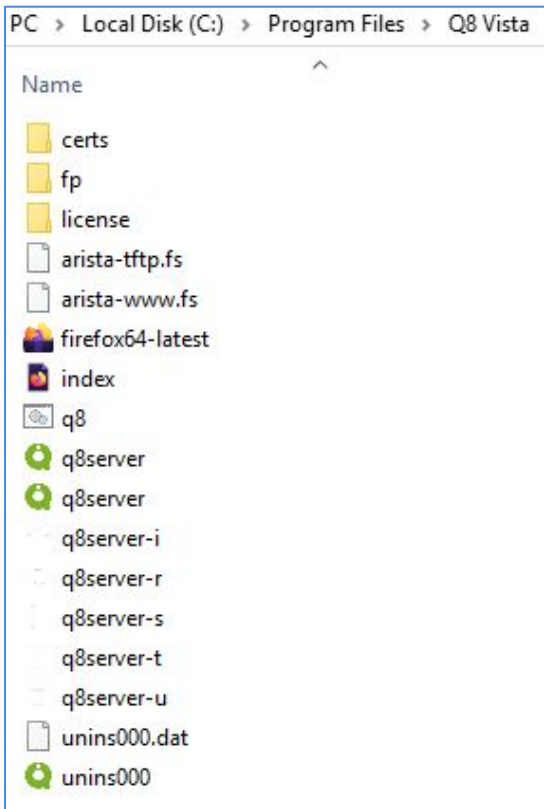
Run Q8Server for Windows

1. 透過 Installer 即安裝程式，它的主要功用有幾項
 - a. 將執行時必要的檔案，放在 C:\Program Files\Q8 Vista\ 參考「安裝後的檔案結構」
 - b. 安裝過程詢問資料庫的處理，參考「資料庫」
 - c. 安裝過程詢問是否建立開始功能表，參考「q8.bat」
 - d. 最後啟動 q8server 服務，參考「q8server 服務」
2. 就上述功能來看，我們可以手動執行上述功能，不一定要透過 Installer
 - a. 執行時必要的有 q8server.exe + www/ + 資料庫
 - b. 其中 q8server.exe 必須跟 www/ 放在一起，資料庫則無法變動位置
 - c. q8server 執行時，有兩種方式，參考「q8server執行方式」，會影響資料庫的位置
 - d. 網頁有修改後，不一定要重新 啟動 q8server，瀏覽器建議關閉分頁重新載入或 Refresh
3. 參考「修改網頁及測試」

Run Q8Server for Linux

1. 在 Linux 下不必 Installer, 等同 Windows 命令視窗模式
2. 執行時在根目錄(root/)必要的有 q8server + www/ + 資料庫
 - a. q8server 執行的目錄即為 Q8Path, 在 Linux 下等同 DocPath
 - b. 網頁有修改後, 不一定要重新 啟動 q8server, 瀏覽器建議關閉分頁重新載入或 Refresh
3. 參考「修改網頁及測試」

安裝後的檔案結構



1. 比較重要的有

- a. q8server.exe 這個執行檔, 預設以 service(服務) 的方式執行
- b. arista-[www.fs](#) 這是 www/ 虛擬磁碟檔, 參考「虛擬磁碟檔」
- c. index.html 這個檔如果不存在, 會以 www/index.html 替代
- d. q8.bat 這個算是多重目的管理檔, 參考「q8.bat」
- e. unins000.exe 反安裝程式, 需配合 unins000.dat

2. 其他

- a. certs/ 是提供 https 的金鑰
- b. fp/ 是提供 terminal 開機的系統檔, 即 firmware package
- c. arista-tftp.fs 是 tftp/ 虛擬磁碟檔, 參考「虛擬磁碟檔」
提供 terminal 開機的 OS
- d. *.ico 為圖示

模擬 terminal

1. 請安裝 [git for windows](#), 建議安裝時除了勾選『On the Desktop』外, 一律以預設值安裝(一直按Next), 安裝後圖示如右。
2. git for windows 除了 git 做版本控制外, 提供
 - a. [bash](#) 環境, 這種環境相當重要, 參考「[bash簡單教學](#)」
 - b. [curl](#) 提供命令列 web client, 供模擬 terminal 時利用 API 與 q8server 溝通
 - c. [vim\(vi\)](#)是 linux 下最常用的編輯工具, 如果有嵌入式系統有興趣務必學習
3. 請 git clone [webui-dist](#), 參考裡面的
 - a. docs/
 - i. [simulate-terminal.pdf](#) 在 windows 下模擬 terminal說明文件
 - ii. [test api](#): 由網頁取得各種 q8server 內部資料
 - iii. [Git-shell-and-資料庫.pdf](#): 探討 q8server 內部資料庫
 - b. scripts/
 - i. terminal.sh 提供模擬 terminal 的各種動作
 - ii. 最新的 q8server 不允許 terminal 回報 OFF, 因此得用 terminal-force.sh 來實現



q8server 執行方式

1. q8server時是依據『使用者身分』取得寫入權，因為有二種執行方式，其使用者身分是不同的，影響 DocPath不同
2. 安裝後預設以服務方式執行，DocPath 在 c:\Windows\System32\config\systemprofile\AppData\Local\Q8Vista\
3. 以命令列方式執行時，DocPath 在 ~\AppData\Local\Q8Vista\
4. 兩種方式的切換請參考「q8.bat」

q8server 服務

1. [q8server 服務](#)模式的管理, 請參考「q8.bat」
2. 以服務方式執行 q8server 時, 它的使用者身分是『系統』, 而用 命令模式 執行時, 它的身分是登入帳號, 會影響 DocPath, 請參考「q8server 執行方式」
3. q8server 以服務方式執行時, 無命令視窗即時看到執行每個 API 時的訊息, 請參考[事件檢視器](#)
4. 有時 q8server 當掉, 外在表現就是怎麼都不聽話, 此時請先停用再啟用, 或是重啟 q8server 服務, 請參考 q8.bat -r 或 q8server.exe restart
5. **如果是以命令模式啟動 q8server, 請以『管理者身分』執行.**

q8.bat

1. 算是 q8server.exe 的過濾器, q8.bat 的大部份功能是透過 q8server.exe 完成
2. 無參數直接執行時
 - a. 檢查 firefox, 必要時會先安裝,
 - b. 利用 firefox 的訪客模式瀏覽 <http://localhost:8088>(將來會改成 <https://localhost:8085>)
3. 幾種執行參數
 - a. q8.bat -f 檢查, 必要時安裝 firefox
 - b. q8.bat -d 反安裝(移除) firefox
 - c. q8.bat -r 重新啟動 q8server 服務, 同 q8server.exe restart
 - d. q8.bat -s 啟動 q8server 服務, 同 q8server.exe start
 - e. q8.bat -t 停止 q8server 服務, 同 q8server.exe stop
 - f. q8.bat -i 安裝 q8server 服務, 同 q8server.exe install
 - g. q8.bat -u 移除 q8server 服務, 同 q8server.exe uninstall
4. 停用 q8server 服務後, 在命令視窗執行q8server.exe此即命令模式

資料庫

1. 首先參考「q8server 執行方式」了解 DocPath 的目錄位置
2. 資料庫名稱一律是 q8s.db, 它是 [sqlite3](#), 請依瀏覽器安裝擴充套件, 例如 [Chrome](#) 或 [Firefox](#)
3. [sqlite3 的語法](#) 簡單易學, 但是只用來驗證 API, 建議以 API 的 payload 或 response 為準
4. 有時候會想要清空資料庫, 做法
 - a. 停止 q8server(請依 服務或命令列方式停用)
 - b. 刪除/更名/移動 q8s.db
5. 同理, 可以手動備份資料庫
6. 最後再強調一下, 不同執行方式的 DocPath 不同

1. *sqlite3 q8s.db* 可以進行資料庫操作, 或是透過擴充套件進行
2. *delete from term;* 清空 terminal
3. 同理 *delete from app;* 清空 app
4. *select * from term;* 取得所有 terminal 設定
5. *select * from term WHERE ObjId=1;* 取得 tid=1 的設定

虛擬磁碟檔

1. 目前主要有兩個
 - a. arista-[www.fs](#) 此即前端網頁所在
 - b. arista-tftp.fs 此即terminal 開機時 OS 檔所在
2. q8server 執行時會載入 arista-tftp.fs 中的 OS 檔，以利加速開機時傳檔
通常不會更換此 OS 檔，如果真的有要更換的話，請先停用 q8server
3. q8server 執行時，會載入 index.html，其它頁面則是動態載入，因此更換 www/ 內容時，只需瀏覽器 refresh 或關閉分頁重新載入，不必重新啟動 q8server
4. q8server 載入網頁以實際 www/ 優先，找不到才會搜尋 arista-[www.fs](#)。
因此不必特意刪除 arista-[www.fs](#) 就可以更新 WebUI
5. 承上，如果把 www/ 刪除，q8server 就又以 arista-[www.fs](#) 內容為主。

bash 簡單教學

1. bash 與 windows cmd 最大不同在於路徑是以 / 分隔, cmd 是以 \ 分隔
2. bash script, 例如 terminal.sh, 是純文字檔,
 - a. 在 git bash 中需要透過 `chmod a+x` 賦與執行權才能執行
 - b. 執行 bash script 時, 需要明確指定路徑, 例如 `./terminal.sh`
3. bash script 除了前面提到的要執行權外, 通常第一行是 `#!/bin/bash`
如果第一行不是 `#!/bin/bash`, 則執行時要加上 bash, 例如 `bash terminal.sh`
前面提到的第一行行首一定是 # 開頭, 請勿留空白
4. bash 環境中請善用 Tab 及上下左右鍵
5. 參考 [bash 參考手冊](#)

參考資料

1. 源碼

- a. 如果你是在公司上班, 可以
 - i. 產生 openssh 金鑰, 把公開金鑰 mailto:wade.chen@aristaipc.com
 - ii. 等通知之後才能進行下一步
 - iii. git clone ssh://q8vista@192.168.1.160/home/q8vista/Q8_Vista/bare/Q8webui
- b. 如果你無法存取公司網路, 可以
 - i. 申請 github 帳號, 把帳號 mailto:wade.chen@aristaipc.com
 - ii. git clone <https://github.com/wade-fs/webui>
 - iii. 等通知才能 push, 但是 github 政策要 push 要申請 token

2. 最新的 Installer 每2週 release 一次

- a. v10: [在這兒下載](#)

3. 如何取得 Jenkins 的檔(日期要自己替換)

- a. 可以下載的有 q8server.exe www.tar.xz Q8_Vista-20220627.exe
- b. curl <http://192.168.1.158:8080/userContent/20220627/q8/q8server.exe> -o q8server.exe