

Wonderbox 特色

WonderBox 簡化了分享的程序，讓分享變成再簡單不過的行為，WonderBox 專利的網路穿透技術免除固定 IP 的需求、也不再對 IP 分享器做任何轉 Port 的繁複設定，只要接上網路，即刻與網際網路(Internet)接軌，不再需要像 DropBox 這類的網路硬碟，就可輕鬆隨時提供跨地域網路存取服務，容量不夠可 隨心所欲加上硬碟即可。輕鬆分享，讓外界的好友連接你專屬的 NAS。

免固定 IP、免對 IP 分享器做任何轉 Port 的繁複設定，Plug&Play

- 可設定多組帳號，並提供遠端連入服務
- 提供遠端管理機制可線上/離線傳送訊息與檔案
- 支援 Streaming 影音串流機制，邊下載邊播放
- 支援 iPhone、iPad、Android 等手持裝置使用
- 支援在各種桌面作業系統使用 Windows, Mac, Linux
- 可自行開發插件(plug-in)，擴充 WonderBox 功能

1. 目的:

利用 WonderBox 核心引擎機制, 開發客製化插件(plug-in)應用擴充功能。

2. 插件(plug-in)結構說明:

每組插件皆包含兩部份：

1. 設定檔

(假設 /wbox 為 WonderBox 安裝目錄)

(1) 設定檔必須存到 /wbox/conf/ 目錄下

(2) 設定檔必須是 json 格式

2. 執行檔

(假設 /wbox 為 WonderBox 安裝目錄)

(1) 執行檔存放的位置是依 設定檔 中 path 所指定的檔案路徑

先來看個簡單的範例

(假設 /wbox 為 WonderBox 安裝目錄)

設定檔: hello_world.json

所以其檔案含路徑 為 /wbox/conf/hello_world.json

檔案內容如下：

```
{
  "module-plugin": [
    {
      "name": "/unet/hello_world.DiskUsage",
      "path": "ext/hello_world/hello_world_disk_usage.sh",
      "allow": [ "user", "admin" ],
      "argument": "env",
      "disable": 0
    }
  ],
}
```

```

{
  "name": "/unet/hello_world.ProcessInfo",
  "path": "ext/hello_world/hello_world_process_info.sh",
  "allow": [ "user", "admin" ],
  "argument": "env",
  "disable": 0
}
]
}

```

設定檔中屬性的說明：

"name": "/unet/hello_world.ProcessInfo" 表示指令名稱, 且其中 『/unet/』 是必要的.
 "path": "ext/hello_world/hello_world_process_info.sh" 表示執行檔的完整檔案含路徑為
 /wbox/ext/hello_world/hello_world_process_info.sh
 "allow": ["user", "admin"] 表示允許授權使用者和管理者使用該指令。
 "argument": "env" 參數是透過環境變數傳遞給執行檔。
 "disable": 0 目前開啓可使用。

以上例子定義了兩個指令：

/unet/hello_world.DiskUsage 和 /unet/hello_world.ProcessInfo

其對應的執行檔分別為：

/wbox/ext/hello_world/hello_world_disk_usage.sh 和

/wbox/ext/hello_world/hello_world_process_info.sh

(記得要 chmod +x 使這二個執行檔成為可執行)

再假設此二執行檔的內容為如下之 Shell Script：

/wbox/ext/hello_world/hello_world_disk_usage.sh =>執行 df 指令

```
#!/bin/bash
```

```
df
if [ "$?" != "0" ]; then
  echo "df execute failed"
fi
```

/wbox/ext/hello_world/hello_world_process_info.sh =>執行 PS 指令，並可下參數

```
#!/bin/bash
```

```
if [ "$WBX_PARAM_KEYWORD" == "" ]; then
  echo "need parameter 'keyword'"
  exit 1;
fi
```

```
ps -ef|grep ${WBX_PARAM_KEYWORD}
if [ "$?" != "0" ]; then
  echo "'ps' execute failed"
fi
```

插件指令之使用方法：可用瀏覽器或其他方法(curl 指令)，透過 HTTP 協定即可使用

方法一:直接連結：(與 wonderbox 同網段或網路路徑可連通時)

http://192.168.1.100:8101/unet/hello_world.DiskUsage

方法二:透過 tunnel 連結 (**wonderbox 無法直接連結時**)

http://stwp26.cyberhood.net.tw/ucloud.a103089/unet/hello_world.DiskUsage

上述方法中的指令寫法說明如下:

192.168.1.100 => 是 Wonderbox 所在的 IP

8101 => Wonderbox 的 listen port，預設設定於 conf.json 中。

stwp26.cyberhood.net.tw => Wonderbox 連接的 tunnel server

103089 => 是 WonderBox 的 Service ID

其中，tunnel server 必須在使用前先詢問 Center (cloud.wonderbox.net.tw)，指令為
[http://cloud.wonderbox.net.tw/_rpc/v3/rpc.php?
__dbid=0&__ver=2&__style=csv&__rpcname=WonderCloud/OnlineInfo&controller_alias
=103089](http://cloud.wonderbox.net.tw/_rpc/v3/rpc.php?__dbid=0&__ver=2&__style=csv&__rpcname=WonderCloud/OnlineInfo&controller_alias=103089)

(103089 指 Wonderbox 的 Service ID)

回傳值：

```
0, "", "", 17856, 9815, "cloud.wonderbox.net.tw"
```

```
33, "61.219.98.65", "192.168.1.47", "stwp26.cyberhood.net.tw"
```

tunnel server 就是回傳值中的第 2 列的最後一欄

附帶說明：(a) 上述回傳值第 1 列的第一個值若不是 0，則表指令執行有誤，

(b) 上述指令必須在登入後才能使用，且寫法若搭配權限設定時，必須再加上其他必要參數，詳細說明請參考下方使用權限

3. 插件(plug-in)的使用權限

Wonderbox 的使用者分成三種：admin, user, anonymous，說明如下：

(a) admin: Wonderbox 預設之管理權限使用者，連線來自同網段、經過授權來自不同網段或透過 tunnel 連結的使用者。

(b) user: 在 wonderBox 檔案總管->權限管理->帳號管理(或是 WonderBox 裝置管理員->User Accounts) 中建立的使用者。

(c) anonymous: 訪客使用者。

系統會依照設定 **allow** 屬性，允許那些使用者可執行插件。插件本身可利用環境變數 **WBX_USER_ID** 得知執行使用者身份。

設定範例 1:

```
{  
  "name": "/enet/hello_world.DiskUsage",  
  "path": "ext/hello_world/hello_world_disk_usage.sh",  
  "allow": [ "user", "admin" ],  
  "argument": "env",  
  "disable": 0  
}
```

設定範例 2:

```
{
```

```
"name": "/unet/hello_world.DiskUsage",
"path": "ext/hello_world/hello_world_disk_usage.sh",
"allow": [ "admin" ],
"argument": "env",
"disable": 0
}
```

設定範例 3:

```
{
  "name": "/unet/hello_world.DiskUsage",
  "path": "ext/hello_world/hello_world_disk_usage.sh",
  "allow": [ "admin", "user", "anonymous" ],
  "argument": "env",
  "disable": 0
}
```

admin 的密碼可以設定或不設定，差別在於：

- * 沒有設定密碼，連線來自同網段直接授權，連線來自不同網段或透過 tunnel 者拒絕授權。
- * 有設定密碼，連線來自同網段、不同網段或透過 tunnel 者都會先詢問密碼。

密碼設定處：

- * WonderBox 系統設定>安全性>Admin 密碼
- * WonderBox 裝置管理員>Service>Admin Password

user 在使用前要先登入：

[http://cloud.wonderbox.net.tw/_rpc/v3/rpc.php?
__dbid=0&__ver=2&__style=csv&__rpcname=SystemEx/UserLogin3&&pop3_login=2&login_n
ame=103089_myusername&login_passwd=mypasswd](http://cloud.wonderbox.net.tw/_rpc/v3/rpc.php?__dbid=0&__ver=2&__style=csv&__rpcname=SystemEx/UserLogin3&&pop3_login=2&login_name=103089_myusername&login_passwd=mypasswd)

(參數說明：)

103089 指 WonderService 的 Service ID

myusername 使用者名稱

mypasswd 使用者密碼

回覆內容：

```
0, "", "", 681, 7226, "cloud.wonderbox.net.tw"  
"DJGBHVEKFYQMTLDEAUFHCTBUA8ETCDAEDHEJHHE9D2"
```

回傳內容中的 **DJGBHVEKFYQMTLDEAUFHCTBUA8ETCDAEDHEJHHE9D2** 授權字串(使用插件時會用到)

使用插件時，要多加一個參數 **u_str**

[http://stwp26.cyberhood.net.tw/ucloud.a103089/unet/hello_world.DiskUsage?
u_str=DJGBHVEKFYQMTLDEAUFHCTBUA8ETCDAEDHEJHHE9D2](http://stwp26.cyberhood.net.tw/ucloud.a103089/unet/hello_world.DiskUsage?u_str=DJGBHVEKFYQMTLDEAUFHCTBUA8ETCDAEDHEJHHE9D2)

或是加在 cookie 中也可以。

4. 插件(plug-in)的參數，分成系統參數與自訂參數。

當以環境變數的形式出現時，有以下

系統參數：(可於程式中自行運用)

```
WBX_USER_ID=15
WBX_CLIENT_IP=192.168.1.33
WBX_NAME=/unet/hello_world.DiskUsage
WBX_SERVICE_HOME=/home/hmc
WBX_SERVICE_ID=103033
```

WBX_USER_ID 指 user's ID，0 表示 admin，-1 表示 anonymous

WBX_CLIENT_IP 指 client's IP

WBX_NAME 為指令名稱

WBX_SERVICE_HOME 為 WonderBox 安裝的目錄

WBX_SERVICE_ID 為 WonderBox 的 Service ID

自訂參數在環境變數的形式中，名稱會有些改變，如：

`http://192.168.1.100:8001/unet/hello_world.ProcessInfo?keyword=Wonder`
傳遞給執行檔時，參數名稱會變為 **WBX_PARAM_KEYWORD**
也就是 `"WBX_PARAM_" + upper_case("keyword")`

5. 插件(plug-in)還有其他屬性：

`"time-out": 30000`

`"argument": "json", (option)可用值："standard", "json"(預設), "env"`

`"response": "text" (option)可用值："text"(預設), "pass-through"`

time-out 逾時時間(millisecond)，0 表示不等待，預設為 60 秒(60000 millisecond)

argument 參數傳遞給執行檔時的形式，`"standard", "json"(預設), "env"`。

`argument: "env"` 表示參數內容會放在環境變數。

如果參數值過大，可能會造成環境變數空間不足，這時就可以考慮改用

`argument: "json"` 參數會從 stdin 以 json 的格式傳入，參數名稱會保持原有的大小寫。

```
{
  "user-id": 15,
  "client-ip": "192.168.1.33",
  "name": "/unet/hello_world.DiskUsage",
  "service-home": "/home/hmc"
  "service-id": 103033,
  "parameters": {
    "keyword": "Wonder",
    "name2": "xxx",
    "name3": "xxx"
  }
}
```

`argument: "standard"` 是將所有參數都放在執行檔後面作為引數，實務上不如 `"env"` 來的方便，就不再贅述。

response 執行回覆的方式，`"text"(預設), "pass-through"`

`response: "text"` 表示指令(執行檔)執行完畢後，才會一併將輸出結果傳回 client。

但是有些指令執行較久，你會希望盡快看到執行中的輸出，或者輸出量大，若不即時傳

到 client，其實就是佔用記憶體空間，這時就可以改用
response: "pass-through" 一有輸出就即刻傳回 client。

使用 "pass-through" 要注意一點，就是 HTTP header 要自行產生，至少要以下
幾行(以 shell script 為例)：

```
echo "HTTP/1.1 200 OK"  
echo "Content-Type: text/plain"  
echo ""
```

至此描述的 WonderBox 插件(plug-in)可以說是指令插件(command plug-in or method plug-in)，
之後還會有其他類型的插件介紹：

- * 事件插件(event plug-in)
- * 規則插件(policy plug-in)
- * 內置網頁
- * 中繼網站

(待續).....

如有任何問題，歡迎來信討論指教...

官網: www.wonderbox.net.tw

郵件地址: charkhsu@kinghood.com