

# S Plus Gaming 串接文件

## 修訂紀錄

版本	日期	修訂紀錄	作者
2.0.0	2024/05/27	初始版本. 顯示API參數字數限制 新增串接流程圖範例	Shawn
2.0.1	2024/05/30	LaunchGame新增參數showBackButton控制按鈕顯示	Shawn
2.0.2	2024/09/10	6.CheckWagerRecords 查詢注單列表wagerdata資料新增參數buyFreeGame, winRatio	Shawn
2.0.3	2024/09/26	新增單一錢包文件要求代理商實作API D.小遊戲下注預先扣款, E.小遊戲下注派彩結果	Shawn
2.0.3	2024/10/07	8. LaunchGame 玩家登入遊戲 新增參數rGameId, 為小遊戲額外必填參數	Shawn
2.0.4	2024/10/14	8. LaunchGame 玩家登入遊戲 移除參數rGameId 6. CheckWagerRecords 新增RoundId, BetInfo, BetOdds, ExpectedWinAmount, WinResult回傳欄位 新增BetInfo欄位說明 單一錢包文 D.小遊戲下注預先扣款新增WinAmount, E.小遊戲下注派彩結果新增BetAmount	Shawn
2.0.5	2024/10/21	新增部分遊戲支援韓文提供韓文語系參數ko-kr	Shawn
2.0.6	2024/10/24	8. LaunchGame 玩家登入遊戲 移除參數category 新增BetInfo欄位說明	Shawn
2.0.7	2024/11/01	新增體育遊戲資料結構說明 新增API 21. GetTokenGameToken 取得token遊戲的token	Shawn

# 目錄

## S Plus Gaming 串接文件

- 修訂紀錄
- 目錄
- 前言
- 名詞解釋
- 時區

## 串接流程

### 代理驗證流程

1. 申請代理帳號
2. 建立代理帳號
3. 給予'代理帳號','代理密碼','JWT密碼'
4. 組建參數
5. 呼叫1.AGLogin
6. 返回'accesstoken'

### 使用者登入遊戲流程

1. 呼叫1.AGLogin
2. 返回'accesstoken'
3. 呼叫2.CreateMember
4. 返回'username'
5. 呼叫(8.LaunchGame)
6. 返回'launchgameurl'

## 加密方法

### API 請求規範

### API 回傳規格

## API列表

### 1. AGLogin 代理登入拿到access token

Request

Response

### 2. CreateMember 創建使用者

Request

Response

### 3. GetGameList 獲取遊戲列表

Request

Response

### 4. CheckMemberBalance 查詢玩家餘額

Request

Response

### 5. GetTransactionsRecords 獲取轉帳紀錄

Request

Response

### 6. CheckWagerRecords 查詢注單列表

Request

Response

7. CheckWagerRecordDetail 查詢注單詳細資料

Request

Response

8. LaunchGame 玩家登入遊戲

Request

Response

9. CheckGameReport 查詢遊戲統計

Request

Response

10. CheckPlayerReport 查詢玩家注單統計

Request

Response

11. Deposit 存款

Request

Response

12. Withdraw 取款

Request

Response

13. KickOutMem 踢玩家

Request

Response

14. KickAndBlockedOutMem 踢玩家跟黑名單

Request

Response

15. GetTransactionsRecordDetail 獲取單筆轉帳紀錄

Request

Response

16. GetGameVersion 獲取遊戲版本號

Request

Response

17. LaunchLobby 啟動遊戲大廳

Request

Response

18. UnblockMem 移除玩家黑名單

Request

Response

19. GetVirtualOrder 獲取可視化投注訂單的詳細信息

Request

Response

20. LaunchLinkGame 啟動單機版遊戲網址

Request

Response

21. GetTokenGameToken 取得token遊戲的token

Request

Response

## 單一錢包接口文檔

文件目的

傳輸規範

安全規範

MD5簽名規範

1. 單一錢包接口說明平台方要提供的 API 接口

A. 玩家當前餘額

Reuquest

Response

B. 老虎機專用 SlotSpin

Reuquest

Response

C. 老虎機失敗注單通知

Reuquest

Response

D. 小遊戲下注預先扣款

Reuquest

Response

E. 小遊戲下注派彩結果

Reuquest

Response

2. 第三方遊戲額外提供的接口

A. 查詢失敗或是超時的下注

Reuquest

Response

B. 恢復失敗的下注

Reuquest

Response

### 3. JP 說明

#### 附件

Response & Error code 對應表

Error message 對應表

語系說明

語系對應表

Category 說明

Category對應表

History 說明

Slot 圖標對應表

Slot 消除類 (深海,埃及,瑪雅)

陣列裡面為該次的遊戲次數,如果有免費遊戲,該陣列裡面會有對多筆紀錄

Slot 連線類 (電音,封神)

resultList如果有免費遊戲,該陣列裡面會有對多筆紀錄

富貴捕魚 魚隻結果對應表

富貴捕魚 子彈類對應表

富貴捕魚

子彈資料結構,每一筆遊戲記錄最多只會紀錄100發子彈資訊

deadFishes 資料結構

BetInfo說明

視訊真人龍虎鬥

視訊真人百家樂

體育足球

體育籃球

體育賽馬

體育賽狗

金豬

獅子兔子

龍虎鬥(Tiger Dragon)

上和下(Up & Down)

天梯(Paricle Ladder)

射龍門(Pently Kick)

賽馬(Horse Racing)

體育遊戲資料結構說明：

體育遊戲 ID

足球 890 主結構：Sport1Bets

子結構：Sport1Bet

JSON 範例

籃球 891 主結構：

子結構：Sport2Bet

JSON 範例

賽馬 892 主結構：

子結構：Sport3Bet

JSON 範例

賽狗 893 主結構：

子結構：Sport4Bet

JSON 範例

遊戲 ID 890 的 BetInfo 說明

遊戲 ID 891 的 BetInfo 說明

遊戲 ID 892 的 BetInfo 說明

遊戲 ID 893 的 BetInfo 說明

# 前言

## 名詞解釋

名詞	解釋
營運商	提供S Plus Gaming系統提供遊戲對接的遊戲營運商。
代理商	與S Plus Gaming系統對接的客戶。
玩家	代理商底下的客戶。玩家為S Plus Gaming遊戲的用戶
JWT	JSON Web Token

# 時區

系統表示所有時間都使用UTC+0，GMT+0。

- 中國大陸/台灣地區應使用UTC+8。
- 泰國/越南地區應使用UTC+7。
- 日本/韓國地區應使用UTC+9。
- 美國東部時區標準時間（EST）為UTC-5，夏令時間（EDT）為UTC-4。
- 美國西部時區標準時間（PST）為UTC-8，夏令時間（PDT）為UTC-7。
- 印度地區應使用UTC+5.5。

# 串接流程

## 代理驗證流程

代理商第一次串接的流程介紹，申請代理商戶後，打 AGLogin API 正確取得 access token 後，token 存到 http header Authorization Bear {{accesstoken}}，在其他需要驗證代理的 API 都需攜帶 token 在 Header 中。





## 1. 申請代理帳號

透過與營運商洽談申請代理商戶，填寫申請資料。

## 2. 建立代理帳號

審核通過後，營運商會建立代理商的代理帳號相關資料。

## 3. 給予'代理帳號','代理密碼','JWT密碼'

營運商會給代理商，代理帳號、代理密碼、JWT密碼，串接API必要資料。

## 4. 組建參數

請參閱[加密方法](#)

## 5. 呼叫1.AGLogin

請參閱[AGLogin\\_代理登入](#)

## 6. 返回'accesstoken'

取得accesstoken後，呼叫其他API都需攜帶accesstoken在Header帶入  
'Authorization: Bearer {{accesstoken}}'

## 使用者登入遊戲流程

代理商底下使用者登入遊戲流程範例，已經成功建立會員可以跳過第3.步驟。



## 1. 呼叫1.AGLogin

請參閱[AGLogin](#) 代理登入

## 2. 返回'accesstoken'

取得accesstoken後，呼叫其他API都需攜帶accesstoken在Header帶入  
'Authorization: Bearer {{accesstoken}}'

## 3. 呼叫2.CreateMember

請參閱[CreateMember](#) 創建使用者，已經成功建立會員可以跳過此步驟。

## 4. 返回'username'

成功呼叫後，會在供應商建立代理商底下會員username

## 5. 呼叫(8.LaunchGame)

請參閱[LaunchGame](#) 玩家登入遊戲，

## 6. 返回'launchgameurl'

成功呼叫後，會返回該會員的遊戲登入網址launchgameurl

# 加密方法

**Header 與 Body 皆使用 JWT 做驗證**

**JWT採用HS256演算法**

1. Header - JWT token 可以從1. AGLogin的access token獲取(呼叫1. AGLogin不需攜帶)
2. POST 的 Body 在傳輸的時候會給代理商 JWT 的 secret 作為資料傳輸的加密金鑰， 將要傳 JSON 的 Body 加密後傳輸
3. POST 的 Body 要使用 raw 的 Text 來傳送，以下是 postman 傳送圖片範例

cURL 傳送範例

```
curl --location --request POST 'https://node-public-api.splusgaming.net/v1' \
--header 'Content-Type: text/plain' \
--data-raw
'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJJSZXF1ZXN0Ijp7InVzZXJuYW1lIjoidGVzdFNvZnQiLCJwYXNzd29yZCI6IjEyMzQ1NiJ9LCJBY3Rpb24iOiJBRE0xvZ2luIn0.Pq1dmi5R3aZGKqT27yiZZV5xB40_e2I607EUTNwjbq'
```

## API 請求規範

皆採取 POST 方式請求，請求 Body 的規範第一層固定是：

```
{  
  "Request": 內含創建、讀取請求的資料，  
  "Action": 指定哪一隻API，為 API 名稱 ex: "AGLogin"  
}
```

## API 回傳規格

回傳的token解密完後，payload的訊息第一層固定為 RespCode , Status , Action , Data .

1. RespCode 對應附件的Response & Errcode對應表格
2. Status 成功success, 失敗fail
3. Action 為對應request的Action
4. Data 內含回傳的相關資料

```
{  
  "RespCode": "5000001",  
  "Status": "fail",  
  "Action": "KickOutMem",  
  "Data": {  
    "username": "testMember01",  
    "userid": "1234567"  
  }  
}
```

## API列表

### 1. AGLogin 代理登入拿到access token

POST {{api\_server\_url}}/v1

## Request

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string(50)	Y	代理帳戶名稱
password	body	string(50)	Y	代理密碼

### Request in body format

```
{
  "Request": {
    "username": "testAgent01",
    "password": "thisispassword"
  },
  "Action": "AGLogin"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
accessToken	body	string	Y	為之後代理需要帶在Authroization的 access token。 過期時效為取得後+12小時

### Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "AGLogin",
  "Data": {
    "accessToken":
    "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1Y1NpdGVJZCI6IjI0DY50CIsImIhdC
    IMTcxNjM2NDY5NCwiZXhwIjoxNzE2NDA3ODk0fQ.56VIy-XHUmMVv0gD7_y-
    ZxOZjhQ9EfMmBf_aFSW7uGk"
  }
}
```

## Response 500 Internal Server Error 內部系統錯誤

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "AGLogin",
  "Data": {
    "accessToken": null
  }
}
```

## 2. CreateMember 創建使用者

POST {{api\_server\_url}}/v1

### Request

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string(50)	Y	會員帳戶

### Request in body format

```
{
  "Request": {
    "username": "testAgent01"
  },
  "Action": "CreateMember"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
userid	body	string	Y	會員ID.同playerId
username	body	string	Y	會員帳戶
userenable	body	string	Y	會員激活狀態(y:是, n:否)
balance	body	string	Y	會員餘額

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "CreateMember",
  "Data": {
    "userdata": {
      "userid": "920020",
      "username": "test23",
      "userenable": "Y",
      "balance": "0"
    }
  }
}
```

## Response 4040099 DUPLICATE\_USER 使用者重複

```
{
  "RespCode": "4040099",
  "Status": "fail",
  "Action": "CreateMember",
  "Data": {
    "userdata": {
      "userid": "null",
      "username": "null",
      "userenable": "null",
      "balance": "null"
    }
  }
}
```

## Response 500 Internal Server Error 內部系統錯誤

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "CreateMember",
  "Data": {
    "userdata": {
      "userid": "null",
      "username": "null",
      "userenable": "null",
      "balance": "null"
    }
  }
}
```

## 3. GetGameList 獲取遊戲列表

POST {{api\_server\_url}}/v1



## Request

參數名稱	參數種類	參數型態	必填	參數說明
page	body	string(50)	Y	頁碼
limit	body	string(50)	option	一頁有幾筆
langx	body	string(5)	Y	遊戲地方語系名稱. <a href="#">提供的語系</a>

### Request in body format

```
{
  "Request": {
    "page": "1",
    "limits": "15",
    "langx": "zh-cn"
  },
  "Action": "GetGameList"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
totalpage	body	string	Y	總頁數
page	body	string	Y	頁碼
gamelist	body	array	Y	遊戲列表

### Explanation of parameters in the gamelist

參數名稱	類型	數據類型	是否必填	描述
gameid	body	string	Y	遊戲編號
gamename	body	string	Y	遊戲標題
gametad	body	string	Y	遊戲標籤 0: 熱門遊戲 1: 推薦遊戲 2: 新遊戲
gamestatus	body	string	Y	遊戲狀態 0: 正常運行 -1: 維護中
version	body	string	Y	遊戲版本

Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "GetGameList",
  "Data": {
    "totalpage": "1",
    "page": "1",
    "gamelist": [
      {
        "gameid": "18",
        "gamename": "slot name",
        "gamepopular": "[0]",
        "gamestate": "0",
        "version": "1.0.0"
      }
    ]
  }
}
```

Response 500 Internal Server Error 內部系統錯誤

```
{
```

```
"RespCode": "5000001",
"Status": "fail",
"Action": "GetGameList",
  "Data": {
    "gamelist": [{
      "gameid": "null",
      "gamename": "null",
      "gamepopular": "null",
      "gamestate": "null",
      "gameplatform": "null",
      "version": "null"
    }]
  }
}
```

## 4. CheckMemberBalance 查詢玩家餘額

POST {{api\_server\_url}}/v1

### Request

參數名稱	類型	數據類型	是否必填	描述
username	body	string(50)	是	會員帳號
playerId	body	string(50)	選擇性	會員ID.同userid

### Request in body format

```
{
  "Request": {
    "username": "test1",
    "playerId": "test1"
  },
  "Action": "CheckMemberBalance"
}
```

## Response

參數名稱	類型	數據類型	是否必填	描述
username	body	string	是	會員帳號
balance	body	string	是	會員餘額
currency	body	string	是	貨幣單位 棄用. 幣別請參照申請代理帳號設定
playerId	body	string	選填	會員ID

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "CheckMemberBalance",
  "Data": {
    "balance": "1.23",
    "currency": "RMB",
    "username": "test1",
    "playerId": "test1",
  }
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "CheckMemberBalance",
  "Data": {
    "balance": "null",
    "currency": "null",
    "username": "null",
    "playerId": "null"
  }
}
```

## 5. GetTransactionsRecords 獲取轉帳紀錄

POST {{api\_server\_url}}/v1

### Request

參數名稱	類型	數據類型	是否必填	描述
starttimestamp	body	string(13)	是	開始時間. 採用Unix Timestamp(Milliseconds) 例：1586422527681(Thu Apr 09 2020 08:55:27 GMT+0000)
endtimestamp	body	string(13)	是	結束時間. 採用Unix Timestamp(Milliseconds) 例：1587030250328(Thu Apr 16 2020 09:44:10 GMT+0000)
page	body	string(50)	是	頁碼
username	body	string(50)	選填	會員帳號
limits	body	string(50)	選填	每頁記錄數，如果未提供參數，默認為25。

## Request in body format

- 查詢時間區段內的所有轉帳記錄

```
{
  "Request": {
    "starttimestamp": "1586422527681",
    "endtimestamp": "1587030250328",
    "page": "1",
  },
  "Action": "GetTransactionsRecords"
}
```

- 查詢時間區段內某個會員轉帳記錄

```
{
  "Request": {
    "username": "test1",
    "starttimestamp": "1586422527681",
    "endtimestamp": "1587030250328",
    "page": "1",
  },
  "Action": "GetTransactionsRecords"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
transuserid	body	string	Y	會員轉帳唯一序號
transrecid	body	string	Y	交易回饋的紀錄 id
transpayno	body	string	Y	存提交易時定義的 id
transamount	body	string	Y	交易金額
transdate	body	string	Y	交易時間
transpayway	body	string	Y	存款"1",提款"-1"
transstatus	body	boolean	Y	是否成功
transmessage	body	string	Y	錯誤訊息

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "transdata": [
      {
        "transuserid": "661553336",
        "transrecid": "767123647",
        "transpayno": "2222333",
        "transamount": "2.96",
        "transdate": "1715336991645",
        "transpayway": "-1",
        "transstatus": true,
        "transmessage": ""
      }
    ]
  },
  "Action": "GetTransactionsRecords"
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Data": {
    "transdata": [
      {
        "transuserid": "null",
        "transrecid": "null",
        "transpayno": "null",
        "transamount": "null",
        "transdate": "null",
        "transpayway": "null"
      }
    ],
  },
  "Action": "GetTransactionsRecords"
}
```

## 6. CheckWagerRecords 查詢注單列表

```
POST {{api_server_url}}/v1
```

*此 API 每 30 秒僅允許一次調用，每次最多返回 1000 條記錄。*

### Request



參數名稱	參數種類	參數型態	必填	參數說明
username	body	string(50)	option	會員帳號. 帶 username 為取該會員注單列表，不帶則取得所有會員注單列表
starttimestamp	body	string(13)	Y	開始時間. 採用Unix Timestamp(Milliseconds) 例：1586422527681(Thu Apr 09 2020 08:55:27 GMT+0000)
endtimestamp	body	string(13)	Y	結束時間. 採用Unix Timestamp(Milliseconds) 例：1587030250328(Thu Apr 16 2020 09:44:10 GMT+0000)
page	body	string(50)	Y	頁碼
langx	body	string(50)	Y	語言 為以下三種 'en-us', 'zh-tw', 'zh-cn'

## Request in body format

- 查詢所有會員注單列表

```
{
  "Request": {
    "starttimestamp": "1585635151793",
    "endtimestamp": "1585635602099",
    "page": "1",
    "langx": "zh-cn"
  },
  "Action": "CheckWagerRecords"
}
```

- 查詢單一會員注單列表

```
{
  "Request": {
    "username": "test1",
    "starttimestamp": "1585635151793",
    "endtimestamp": "1585635602099",
    "page": "1",
    "langx": "zh-cn"
  },
  "Action": "CheckWagerRecords"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
totalpage	body	string	Y	總頁數
page	body	string	Y	第幾頁
wagerdata	body	object	Y	注單資料

### wagerdata 內的參數說明

參數名稱	參數種類	參數型態	必填	參數說明
wagerid	body	string	Y	注單單號
gamerecordid	body	string	Y	局號
username	body	string	Y	會員名稱
userid	body	string	Y	會員編號
ecsiteid	body	string	Y	代理商編號
gameid	body	string	Y	遊戲編號
beforebetbalance	body	string	Y	投注前金額
betamount	body	string	Y	下注金額

validamount	body	string	Y	有效下注金額
winamount	body	string	Y	派彩金額，遊戲方回給玩家金額+房主返水
resultamount	body	string	Y	損益金額+房主返水
settle	body	string	Y	'1'：有結果, '0':沒結果
result	body	string	Y	注單結果 "L": 輸, "W": 贏, "D": 取消, "N": 打平, "0" 沒結果
afterbetbalance	body	string	Y	投注後金額
wagerdate	body	string	Y	注單日期
device	body	string	Y	下注裝置 手機"M"，網頁"S"
gametype	body	string	Y	麻將遊戲類型 1: 單人; 2: 多人; 3: 多人對戰; 4: 好友房 沒有是'null'
kkBeforBalance	body	string	Y	KK錢包扣款前餘額
kkAfterBalance	body	string	Y	KK錢包扣款後餘額
recordtype	body	string	Y	資料列類型 Game: 遊戲紀錄, CompetitionReward: 比賽獎金, CompetitionOrder: 比賽報名費
buyFreeGame	body	boolean	Y	有沒有買免費遊戲
winRatio	body	decimal	Y	贏分倍率
RoundId	body	string	Y	局號
BetInfo	body	string	Y	下注資訊. 提供特定遊戲, 請參照 <a href="#">BetInfo說明</a>
BetOdds	body	float64	Y	下注賠率
ExpectedWinAmount	body	float64	Y	預計贏的金額
WinResult	body	int	Y	贏的結果 1: 贏，2: 不贏，3: 和局

注意:

- 好友房相關資料時

- winamount 會加入房主返水
- resultamount 會加入房主返水
- 比賽場相關資料時
  - 沒有 gameRecordId, beforeBetBalance, afterBetBalance, validamount 等值
  - betAmount 為入場費
  - winAmount 為 獎金或是返還入場費
  - recordtype 為比賽報名費、比賽獎金時的 wagerid 內容為比賽訂單相關的 Unique Id

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "totalpage": "1",
    "page": "1",
    "wagerdata": [
      {
        "wagerid": "0EV000000000000000067",
        "gamerecordid": "GR-17202885CCD-1-36",
        "username": "test2",
        "userid": "20200438202",
        "ecsiteid": "34863",
        "gameid": "18",
        "beforebetbalance": "980100",
        "betamount": "100",
        "validamount": "100",
        "winamount": "0",
        "resultamount": "-100",
        "settle": "1",
        "result": "L",
        "afterbetbalance": "980000",
        "wagerdate": "1587464892762",
        "device": "S",
```

```
        "gametype": "0",
        "kkBeforBalance": "99900",
        "kkAfterBalance": "100000",
        "recordtype": "Game"
    },
    {
        "wagerid": "0EV0000000000000000065",
        "gamerecordid": "GR-1720D59EC16-2-B0",
        "username": "test2",
        "userid": "20200438202",
        "ecsiteid": "34863",
        "gameid": "18",
        "beforebetbalance": "980300",
        "betamount": "100",
        "validamount": "100",
        "winamount": "0",
        "resultamount": "-100",
        "settle": "1",
        "result": "L",
        "afterbetbalance": "980200",
        "wagerdate": "1587464892760",
        "device": "S",
        "gametype": "3",
        "kkBeforBalance": "99900",
        "kkAfterBalance": "100000",
        "recordtype": "Game"
    },
]
},
"Action": "CheckWagerRecords"
}
```

## Response 500 Internal Server Error

```
{
    "RespCode": "5000001",
```

```
"Status": "fail",
"Data": {
  "totalpage": "1",
  "page": "1",
  "wagerdata": [
    {
      "wagerid": "null",
      "gamerecordid": "null",
      "username": "null",
      "userid": "null",
      "ecsiteid": "null",
      "gameid": "null",
      "beforebetbalance": "null",
      "betamount": "null",
      "validamount": "null",
      "winamount": "null",
      "resultamount": "null",
      "settle": "null",
      "result": "null",
      "afterbetbalance": "null",
      "wagerdate": "null",
      "device": "null",
      "kkBeforBalance": "null",
      "kkAfterBalance": "null",
      "recordtype": "null",
    },
  ],
},
"Action": "CheckWagerRecords"
}
```

## 7. CheckWagerRecordDetail 查詢注單詳細資料

POST {{api\_server\_url}}/v1

此 API 每 30 秒僅允許一次調用，每次最多返回 1000 條記錄。

## Request

參數名稱	參數種類	參數型態	必填	參數說明
wagerid	body	string(64)	Y	注單編號

### Request in body format

```
{
  "Request": {
    "wagerid": "0E7000000002F0HM0004"
  },
  "Action": "CheckWagerRecordDetail"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
wagerdetaildata	body	object	Y	注單詳細資料
gamerecorddata	body	object	Y	該注單的遊戲紀錄

### wagerdetaildata 內的參數說明

參數名稱	參數種類	參數型態	必填	參數說明
gamerecordid	body	string	Y	遊戲紀錄編號
ecsiteid	body	string	Y	代理商編號
userid	body	string	Y	會員編號
gameid	body	string	Y	遊戲編號
bettimestamp	body	string	Y	下注時間
betamount	body	string	Y	下注金額
validamount	body	string	Y	下注有效金額
winamount	body	string	Y	下注金額結果+房主返水(好友房)
result	body	string	Y	下注結果L: 輸、W: 贏、D: 取消、N: 沒結果、O: 打平
balance	body	string	Y	餘額

#### gamerecorddata 內的參數說明



參數名稱	參數種類	參數型態	必填	參數說明
wagerid	body	string	Y	注單編號
gameid	body	string	Y	遊戲編號
themeid	body	string	Y	廳館編號
roomid	body	string	Y	房間編號
gametype	body	string	Y	遊戲類型 1: 單人; 2: 多人; 3: 多人對戰; 4: 好友房 沒有是'null'
mergedgamerecordid	body	string	Y	好友房場次號, 一般遊戲是'null'
starttimestamp	body	string	Y	下注開始時間 Unix Epoch timestamp in milliseconds
endtimestamp	body	string	Y	下注結束時間 Unix Epoch timestamp in milliseconds
history	body	string	Y	遊戲定義的結果, 以JSON格式顯示。詳細說明在附件

注意:

大眾&紅中 history 中 胡牌只會有一張所以 winnerTile 為一整數 (Ex: 1)

血戰&血流 history 中 胡牌可能有多張牌 winnerTile 為一陣列 (Ex:[13,13,13])

好友房 winamount 會加入房主返水

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "wagerdetaildata": {
      "gamerecordid": "1585635151793-G2-183-149",
      "ecsiteid": "34863",
      "userid": "34867",
      "gameid": "18",
```

```
        "bettimestamp": "1585635331906",
        "betamount": "4",
        "validamount": "4",
        "winamount": "192",
        "result": "W",
        "balance": "1065900"
    },
    "gamerecorddata": {
        "wagerid": "0E7000000002F0HM0004",
        "gameid": "18",
        "themeid": "1",
        "roomid": "1",
        "gametype": "2",
        "mergedgamerecordid": "xxxxxx",
        "starttimestamp": "1585635151793",
        "endtimestamp": "1585635151793",
        "history": {},
    }
},
"Action": "CheckWagerRecordDetail"
}
```

## Response 500 Internal Server Error

```
{
    "RespCode": "5000001",
    "Status": "fail",
    "Data": {
        "wagerdetaildata": {
            "gamerecordid": "null",
            "ecsiteid": "null",
            "userid": "null",
            "gameid": "null",
            "bettimestamp": "null",
            "betamount": "null",
            "validamount": "null",

```

```
        "winamount": "null",
        "result": "null",
        "balance": "null"
    },
    "gamerecorddata": {
        "gameid": "null",
        "themeid": "null",
        "roomid": "null",
        "gametype": "null",
        "starttimestamp": "null",
        "endtimestamp": "null",
        "history": "null",
    }
},
"Action": "CheckWagerRecordDetail"
}
```

## 8. LaunchGame 玩家登入遊戲

POST {{api\_server\_url}}/v1

### Request

參數名稱	類型	數據類型	是否必填	描述
username	body	string(50)	是	會員帳號
gameid	body	string(50)	是	遊戲編號
amount	body	string(10)	否	目前遊戲金額僅允許一次玩一款遊戲, 並將其帶入單個遊戲。
roomNumber	body	string(6)	是	如果是好友房間, 需要帶入房間號碼 (六位數字), 如果不是, 帶入 '0'
langx	body	string(5)	否	遊戲地方語系名稱, 如果指定不正確或未指定, 將統一設置為 "en-us". 提供的語系
category	body	string(3)	否	<del>指定大廳遊戲類型的參數. 提供的遊戲類型</del>
showBackButton	body	boolean	否	是否顯示返回上一頁按鈕. true:顯示;false:不顯示 目前只適用"加密貨幣"類型遊戲

## Request in body format

```
{
  "Request": {
    "username": "test13",
    "gameid": "18",
    "amount": "100",
    "roomNumber": "0",
    "langx": "zh-cn",
    "currency": "CNY",
    "category": "0",
  },
  "Action": "LaunchGame"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string	Y	使用者帳戶名稱
userid	body	string	Y	使用者id
launchgameurl	body	string	Y	登入遊戲的網址(會攜帶lang, currency, cateogry的欄位)
usertoken	body	string	Y	登入遊戲的token(會攜帶lang, currency, cateogry的欄位)

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "LaunchGame",
  "Data": {
    "launchgameurl": "xxxxxx/url/?.....",
    "username": "test13",
    "userid": "34891",
    "usertoken": "xxxxxxtoken"
  }
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "LaunchGame",
  "Data": {
    "launchgameurl": "null",
    "username": "test13",
    "userid": "34891",
    "usertoken": "null"
  }
}
```

## 9. CheckGameReport 查詢遊戲統計

POST {{api\_server\_url}}/v1

### Request

參數名稱	參數種類	參數型態	必填	參數說明
gameid	body(50)	string	option	遊戲編號 沒有帶則查詢所有遊戲
starttimestamp	body(13)	string	Y	開始時間. 採用Unix Timestamp(Milliseconds) 例：1586422527681(Thu Apr 09 2020 08:55:27 GMT+0000)
endtimestamp	body(13)	string	Y	結束時間. 採用Unix Timestamp(Milliseconds) 例：1587030250328(Thu Apr 16 2020 09:44:10 GMT+0000)

### Request in body format

- 查詢所有遊戲下注統計

```
{
  "Request": {
    "starttimestamp": "1584356400000",
    "endtimestamp": "1584359999999"
  },
  "Action": "CheckGameReport"
}
```

#### ■ 查詢單一遊戲下注統計

```
{
  "Request": {
    "gameid": "1",
    "starttimestamp": "1584356400000",
    "endtimestamp": "1584359999999"
  },
  "Action": "CheckGameReport"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
gameid	body	string	option	遊戲編號 或 'All' 所有遊戲
betcount	body	string	Y	查詢區間的總筆數
betamount	body	string	Y	查詢區間的總下注金額
validbetamount	body	string	Y	查詢區間的總有效下注金額
payoutamount	body	string	Y	查詢區間的總贏分
result	body	string	Y	查詢區間的總淨分

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "gameid": "All",
    "betcount": "1003",
    "betamount": "21798.58",
    "validbetamount": "39980.1966",
    "payoutamount": "251879.5643",
    "result": "39980.1966"
  },
  "Action": "CheckGameReport"
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Data": {
    "gameid": "null",
    "betcount": "null",
    "betamount": "null",
    "validbetamount": "null",
    "payoutamount": "null",
    "result": "null"
  },
  "Action": "CheckGameReport"
}
```

## 10. CheckPlayerReport 查詢玩家注單統計

POST {{api\_server\_url}}/v1



## Request

參數名稱	參數種類	參數型態	必填	參數說明
ecUserId	body(50)	string	option	ecSite會員編號.同username 有帶 ecUserId 為查詢該會員統計; 沒帶 ecUserId 則查詢所有會員統計
playerId	body(50)	string	option	會員ID.同userid 有帶 playerId 為查詢該會員統計; 沒帶 playerId 則查詢所有會員統計
starttimestamp	body(13)	string	Y	開始時間. 採用Unix Timestamp(Milliseconds) 例：1586422527681(Thu Apr 09 2020 08:55:27 GMT+0000)
endtimestamp	body(13)	string	Y	結束時間. 採用Unix Timestamp(Milliseconds) 例：1587030250328(Thu Apr 16 2020 09:44:10 GMT+0000)

### Request in body format

- 查詢所有會員下注統計

```
{
  "Request": {
    "starttimestamp": "1584356400000",
    "endtimestamp": "1584359999999"
  },
  "Action": "CheckPlayerReport"
}
```

- 查詢單一會員下注統計



## Response

參數名稱	參數種類	參數型態	必填	參數說明
ecuserid	body	string	option	ecsite會員名稱 或 'All' 所有會員
playerId	body	string	option	會員名稱 或 'All' 所有會員
betcount	body	string	Y	查詢區間的總筆數
betamount	body	string	Y	查詢區間的總下注金額
validbetamount	body	string	Y	查詢區間的總有效下注金額
payoutamount	body	string	Y	查詢區間的總贏分
result	body	string	Y	查詢區間的總淨分

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "ecuserid": "All",
    "playerId": "All",
    "betcount": "1003",
    "betamount": "291859.7609",
    "validbetamount": "291859.7609",
    "payoutamount": "251879.5643",
    "result": "39980.1966"
  },
  "Action": "CheckPlayerReport"
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
```

```
"Status": "fail",
"Data": {
  "ecuserid": "null",
  "playerId": "null",
  "betcount": "null",
  "betamount": "null",
  "validamount": "null",
  "payoutamount": "null",
  "result": "null"
},
"Action": "CheckPlayerReport"
}
```

## 11. Deposit 存款

後台會紀錄轉帳失敗紀錄

1. 餘額不足。
2. 玩家還在遊戲中，不允許轉帳

不會紀錄的情境有

1. API 參數錯誤
2. 找不到玩家
3. 找不到廠商
4. 其他未知錯誤

```
POST {{api_server_url}}/v1
```

### Request

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string(50)	Y	會員帳號
amount	body	string(32)	Y	儲值金額
payno	body	string(64)	Y	付款單號
transferid	body	string(64)	Y	可反查的細單id. 可同付款單號

## Request in body format

```
{
  "Request": {
    "username": "test1",
    "amount": "125",
    "payno": "34863-3",
    "transferid": "xxxx-xxx-xxx",
  },
  "Action": "Deposit"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string	Y	會員帳號
txid	body	string	Y	轉帳編號
payway	body	string	Y	存款"1",提款"-1"
currency	body	string	Y	貨幣單位
reqamount	body	string	Y	存款金額
balance	body	string	Y	存款後餘額
payno	body	string	Y	付款單號
trxMessage	body	string	Y	錯誤訊息
trxStatus	body	boolean	Y	存款是否成功

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "moneydata": {
      "username": "dddaaa3",
      "txid": "766990385",
      "payway": "1",
      "currency": "TWD",
      "reqamount": "2000",
      "balance": "249839",
      "payno": "test1111",
      "trxMessage": "null",
      "trxStatus": true
    }
  },
  "Action": "Deposit"
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Data": {
    "moneydata": {
      "username": "null",
      "txid": "null",
      "payway": "null",
      "currency": "null",
      "reqamount": "null",
      "balance": "null",
      "payno": "null",
      "trxMessage": "null",
      "trxStatus": false
    }
  },
  "Action": "Deposit"
}
```

## 12. Withdraw 取款

後台會紀錄轉帳失敗紀錄

1. 餘額不足。
2. 玩家還在遊戲中，不允許轉帳

不會紀錄的情境有

1. API 參數錯誤
2. 找不到玩家
3. 找不到廠商
4. 其他未知錯誤

POST {{api\_server\_url}}/v1

## Request

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string(50)	Y	會員帳戶
amount	body	string(32)	Y	儲值金額
payno	body	string(64)	Y	付款單號
transferid	body	string(64)	Y	可反查的詳細訂單id. 可同付款單號

## Request in body format

```
{
  "Request": {
    "username": "test1",
    "amount": "125",
    "payno": "34863-3",
    "transferid": "xxxx-xxx-xxx",
  },
  "Action": "Withdraw"
}
```

## Response



參數名稱	參數種類	參數型態	必填	參數說明
username	body	string	Y	會員帳號
txid	body	string	Y	轉帳編號
payway	body	string	Y	存款"1",提款"-1"
currency	body	string	Y	貨幣單位
reqamount	body	string	Y	取款金額
balance	body	string	Y	取款後餘額
payno	body	string	Y	付款單號
trxMessage	body	string	Y	錯誤訊息
trxStatus	body	boolean	Y	存款是否成功

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Message": "",
  "Data": {
    "moneydata": {
      "username": "dddaaa3",
      "txid": "767016956",
      "payway": "-1",
      "currency": "TWD",
      "reqamount": "91.96",
      "balance": "249747.04",
      "payno": "2222333",
      "trxMessage": "null",
      "trxStatus": true
    }
  },
  "Action": "Withdraw"
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Data": {
    "moneydata": {
      "username": "null",
      "txid": "null",
      "payway": "null",
      "currency": "null",
      "reqamount": "null",
      "balance": "null",
      "payno": "null",
      "trxMessage": "null",
      "trxStatus": false
    }
  },
  "Action": "Withdraw"
}
```

## 13. KickOutMem 踢玩家

```
POST {{api_server_url}}/v1
```

### Request

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string(50)	Y	使用者帳戶名稱

### Request in body format

```
{
  "Request": {
    "username": "testMember01",
  },
  "Action": "KickOutMem"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string	Y	會員暱稱
userid	body	string	Y	會員ID. 同playerId

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "KickOutMem",
  "Data": {
    "username": "testMember01",
    "userid": "1234567"
  }
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "KickOutMem",
  "Data": {
    "username": "testMember01",
    "userid": "1234567"
  }
}
```

## 14. KickAndBlockedOutMem 踢玩家跟黑名單

POST {{api\_server\_url}}/v1

### Request

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string(50)	Y	使用者帳戶名稱

### Request in body format

```
{
  "Request": {
    "username": "testMember01",
  },
  "Action": "KickAndBlockedOutMem"
}
```

### Response

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string	Y	會員暱稱
userid	body	string	Y	會員ID. 同playerId

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "KickOutMem",
  "Data": {
    "username": "testMember01",
    "userid": "1234567"
  }
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "KickOutMem",
  "Data": {
    "username": "testMember01",
    "userid": "1234567"
  }
}
```

# 15. GetTransactionsRecordDetail 獲取單筆轉帳紀錄

POST {{api\_server\_url}}/v1

## Request

參數名稱	參數種類	參數型態	必填	參數說明
transpayno	body	string	Y	轉帳單號

## Request in body format

```
{
  "Request": {
    "transpayno": "34863-8"
  },
  "Action": "GetTransactionsRecordDetail"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
transuserid	body	string	Y	會員轉帳唯一序號
transrecid	body	string	Y	交易回饋的紀錄 id
transpayno	body	string	Y	存提交易時定義的 id
transamount	body	string	Y	交易金額
transdate	body	string	Y	交易時間
transpayway	body	string	Y	存款"1",提款"-1"
transstatus	body	boolean	Y	是否成功
transmessage	body	string	Y	錯誤訊息

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "transdata": [
      {
        "transuserid": "661553336",
        "transrecid": "767123647",
        "transpayno": "2222333",
        "transamount": "2.96",
        "transdate": "1715336991645",

```

```
        "transpayway": "-1",
        "transstatus": true,
        "transmessage": ""
    },
]
},
"Action": "GetTransactionsRecordDetail"
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "50000001",
  "Status": "fail",
  "Data": {
    "transdata": {
      "transuserid": "null",
      "transrecid": "null",
      "transpayno": "null",
      "transamount": "null",
      "transdate": "null",
      "transpayway": "null"
    },
  },
  "Action": "GetTransactionsRecordDetail"
}
```

## 16. GetGameVersion 獲取遊戲版本號

```
POST {{api_server_url}}/v1
```

## Request

參數名稱	參數種類	參數型態	必填	參數說明
gameid	body	string(5)	Y	遊戲編號

### Request in body format

```
{
  "Request": {
    "gameid": "1"
  },
  "Action": "GetGameVersion"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
version	body	string	Y	遊戲版號

### Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "version": "1.0.0",
  },
  "Action": "GetGameVersion"
}
```



## 17. LaunchLobby 啟動遊戲大廳

POST {{api\_server\_url}}/v1

### Request

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string(50)	Y	會員名稱 帶 username 為取該會員領取返水紀錄，必帶
userid	body	string(50)	option	會員編號 帶 userid 為取該會員領取返水紀錄，可以不帶
langx	body	string(5)	Y	語系

### Request in body format

```
{
  "Request": {
    "username": "test1",
    "langx": "zh-tw"
  },
  "Action": "LaunchLobby"
}
```

### Response

參數名稱	參數種類	參數型態	必填	參數說明
launchlobbyurl	body	string	Y	登入遊戲的網址(會攜帶lang, currency, cateogry的欄位)

### Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "LaunchLobby",
  "Data": {
    "launchlobbyurl": "xxxxxx/url/?....."
  }
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "LaunchLobby",
  "Data": {
    "launchgameurl": "null"
  }
}
```

# 18. UnBlockedMem 移除玩家黑名單

POST {{api\_server\_url}}/v1

## Request

參數名稱	參數種類	參數型態	必填	參數說明
username	body(50)	string	Y	使用者帳戶名稱

## Request in body format

```
{
  "Request": {
    "username": "testMember01",
  },
  "Action": "UnBlockedMem"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
username	body	string	Y	會員暱稱
userid	body	string	Y	會員 id

## Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Action": "UnBlockedMem",
  "Data": {
    "username": "testMember01",
    "userid": "1234567"
  }
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "UnBlockedMem",
  "Data": {
    "username": "testMember01",
    "userid": "1234567"
  }
}
```

## 19. GetVirtualOrder 獲取可視化投注訂單的詳細信息

目前只提供Slot Game

```
GET {{game_records_url}}/web/slot/$gamerecordid
```

### Request

參數名稱	類型	數據類型	是否必填	描述
\$gamerecordid	參數	string	是	遊戲回合id

### Response

#### Response 200 OK

如果 \$gamerecordid 正確，將顯示虛擬投注訂單；否則，將顯示系統錯誤。

## 20. LaunchLinkGame 啟動單機版遊戲網址

```
POST {{api_server_url}}/v1
```

## Request

參數名稱	類型	數據類型	是否必填	描述
gameid	body	string	是	遊戲編號. 目前可帶入遊戲只限 411,413,414,415,416,417,418,419,421,422,423
redirectUrl	body	string(200)	是	遊戲內導向真錢版網址

```
{
  "Request": {
    "gameid": "413",
    "redirectUrl": "https://www.google.com"
  },
  "Action": "LaunchLinkGame"
}
```

## Response

參數名稱	參數種類	參數型態	必填	參數說明
launchgameurl	body	string	Y	遊戲網址

### Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Data": {
    "launchgameurl": "https://aaa.bbb/ccc/ddd/?single=abcdefg"
  },
  "Action": "LaunchLinkGame"
}
```

### Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "LaunchLinkGame",
  "Data": {
    "launchgameurl": "null"
  }
}
```

## 21. GetTokenGameToken 取得token遊戲的token

POST {{api\_server\_url}}/v1

### Request

參數名稱	類型	數據類型	是否必填	描述
gameId	body	string	是	遊戲編號. 目前可帶入遊戲只限888,889

```
{
  "Request": {
    "gameId": "888"
  },
  "Action": "GetTokenGameToken"
}
```

### Response

參數名稱	參數種類	參數型態	必填	參數說明
gameId	body	string	Y	遊戲編號. 依照帶入參數返回
token	body	string	Y	token 遊戲使用

### Response 200 OK

```
{
  "RespCode": "2000001",
  "Status": "success",
  "Message": "",
  "Data": {
    "gameId": "888",
    "token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJhZ2VudCI6ImlNQSEQ5NyIsInVzZXIiOiIyZzM1ZGxlanltZXJxenBjc2NoNmhhIiwiaWF0IjoiYXBvbGxvZ2FtZTc3Ny5jb20iLCJnYW1lIjoib2UiLCJpcCI6IjEwLjg1LjAuMjU0IiwiaWF0IjoxNzUyMjQ3MDYxLCJleHAiOiE3MzA0NTA2NjF9.ExXY5YPD3-qGaAmbf2sPUoY-388erRq3hTw9i9ema64"
  },
  "iat": 1730447086
}
```

## Response 500 Internal Server Error

```
{
  "RespCode": "5000001",
  "Status": "fail",
  "Action": "GetTokenGameToken",
  "Data": {
    "gameId": "",
    "token": ""
  }
}
```

# 單一錢包接口文檔

單一錢包 API 不需要在 POST 的 Body 在傳輸的時候會給代理商 JWT 的 secret 作為資料傳輸的加密金鑰， 將要傳 JSON 的 Body 加密後傳輸

# 文件目的

- 本文檔介紹平台接入遊戲系統流程、接口參數規範、加密驗簽規則，並於文末提供各語言版本串接範例。

## 傳輸規範

- 使用TCP/IP作為傳輸層協議
- 使用HTTP作為應用層協議
- 傳遞參數格式皆為json(須使用Content-type規範header)

## 安全規範

- 所有接口必須校驗 MD5 sign 簽名值正確無誤,再處理業務邏輯
- 建議平台將遊戲系統通知主機加入信任名單,去防範其餘外來主機偽裝調用接口

## MD5簽名規範

傳送的 body 的部分內容為

```
{
  "PlayerId": "12345",
  "Xid": "sdf32-sdf23r-sf32r",
  "GameId": "15",
  "BetAmount": 100000(Game currency uses a 1:10000 ratio),
  "WinAmount": 300000(Game currency uses a 1:10000 ratio),
  "StartTimestamp": 123141245214,
  "Sign": "58df2b36d1edb86c37f6f5cb363587b0"
}
```

依序取出 Xid + PlayerId + GameId + BetAmount(轉成 string) + WinAmount(轉成 string) + StartTimestamp(轉成 string)

以上範例為



```
"12345" + "sdf32-sdf23r-sf32r" + "15" + "100000" + "300000" + "123141245214"
```

變成完整的 string 串接在一起 12345sdf32-sdf23r-sf32r15100000300000123141245214

再經過 Md5 之後會輸出 58df2b36d1edb86c37f6f5cb363587b0

若無BetAmount或WinAmount則略過該參數的加密排序

效驗過才算是有效的單，效驗不過，請不要執行!!!

## 1. 單一錢包接口說明平台方要提供的 API 接口

### A. 玩家當前餘額

玩家在平台的當前餘額，不需要 MD5 效驗。

#### Reuquest

- Method: POST
- URL: 平台提供自定義接口位置
- Headers : Content-Type:application/json
- Body:

```
{  
  "playerId": "12345",  
  "username": "player1",  
}
```

欄位	屬性	說明
playerId	string	遊戲平台玩家 Id
username	string	玩家的暱稱

## Response

- Body:

```
{
  "resCode": "0",
  "data": {
    "balance": 50000,
  }
}
```

欄位	屬性	說明
resCode	string	0: 成功、1: 失敗
balance	number	玩家當前在平台上的餘額(需要乘上 10000，再傳送過來)

## B.老虎機專用 SlotSpin

專門老虎機專用的接口

## Reuquest

- Method: POST
- URL: 平台提供自定義接口位置
- Headers：Content-Type:application/json
- Body:

```
{
  "PlayerId": "12345",
  "Xid": "sdf32-sdf23r-sf32r",
  "GameId": "15",
  "BetAmount": 100000(Game currency uses a 1:10000 ratio),
  "WinAmount": 300000(Game currency uses a 1:10000 ratioT),
  "StartTimestamp": 123141245214,
  "Sign":"58df2b36d1edb86c37f6f5cb363587b0",
}
```

```
"EndTimeStamp": 123141245214,
"EcSiteId": "12",
"Nickname": "player1",
"PlayerProfit": 200000,
"SingleWalletType": 1,
"GameRecordId": "GR-234fsaf-faaf3r3",
"History": {},
"jpXid": "sdf-sdf23t-gjnas",
"jpTimeStamp": 123141245214,
"jpLevel": 0,
"jpHitValue": 3243.342,
"jp_Value_New_J1": 112144.244,
"jp_Value_New_J2": 6222.244,
"jp_Value_New_J3": 634.22,
"jpIsHit": false,
"buyFreeGame": false
}
```

參數名稱	數據類型	描述
PlayerId	string	遊戲平台玩家 ID
Xid	string	請求唯一代碼
GameId	string	遊戲編號
BetAmount	number	本輪投注金額（實際金額需除以 10000；遊戲貨幣比例為 1:10000）
WinAmount	number	本輪中獎金額（實際金額需除以 10000；遊戲貨幣比例為 1:10000）
StartTimeStamp	number	開始投注時間，Unix 13 位數字
EndTimeStamp	number	結束投注時間，Unix 13 位數字
Sign	string	MD5 驗證碼（參考 MD5 驗證說明）
EcSiteId	string	遊戲平台中的代理平台 ID

Nickname	string	玩家暱稱
PlayerProfit	number	玩家的淨利潤（本輪中獎金額減去本輪投注金額；實際金額需除以 10000；遊戲貨幣比例為 1:10000）
SingleWalletType	number	內部使用，平台不使用
GameRecordId	string	本輪遊戲訂單號
History	object	遊戲中每筆投注的詳細信息
jpXid	string	jp ID
jpTimestamp	number	jp 中獎時間
jpLevel	number	jp 中獎級別 0: JackpotPlatinum, 1: JackpotGold, 2: JackpotSilver
jpHitValue	number	jp 中獎金額
jp_Value_New_J1	number	現在 jp 值 JackpotPlatinum
jp_Value_New_J2	number	現在 jp 值 JackpotGold
jp_Value_New_J3	number	現在 jp 值 JackpotSilver
jpIsHit	bool	是否中 jp
buyFreeGame	bool	是否購買免費遊戲

## Response

- Body:

```
{
  "resCode": "0",
  "data": {
    "afterBalance": 50000,
    "beforeBalance": 80000,
  }
}
```

參數名稱	數據類型	描述
resCode	string	0: 成功 1: 失敗
afterBalance	number	玩家處理後的餘額（在傳輸之前需乘以 10000）
beforeBalance	number	玩家處理前的餘額（在傳輸之前需乘以 10000）

## C.老虎機失敗注單通知

如果老虎機有失敗的注單，會透過這個持續打這個接口，每 1 分鐘打一次，持續到超過四天的注單就不會再通知。

如果收到接收成功，也不會再持續通知這筆注單。

失敗注單通知以Xid為準，只會在呼叫恢復失敗的下注時候提供新的GameRecordId

## Reuquest

- Method: POST
- URL: 平台提供自定義接口位置
- Headers : Content-Type:application/json
- Body:

```
{
  "PlayerId": "12345",
  "Xid": "sdf32-sdf23r-sf32r",
  "GameId": "15",
  "BetAmount": 100000(Game currency uses a 1:10000 ratio),
  "WinAmount": 300000(Game currency uses a 1:10000 ratioT),
  "StartTimestamp": 123141245214,
  "Sign":"58df2b36d1edb86c37f6f5cb363587b0",
  "EndTimestamp": 123141245214,
  "EcSiteId": "12",
  "Nickname": "player1",
  "PlayerProfit": 200000,
  "SingleWalletType": 1,
```

```
"GameRecordId": "GR-234fsaf-faaf3r3",
"History": {},
"jpXid": "sdf-sdf23t-gjnas",
"jpTimestamp": 123141245214,
"jpLevel": 0,
"jpHitValue": 3243.342,
"jp_Value_New_J1": 112144.244,
"jp_Value_New_J2": 6222.244,
"jp_Value_New_J3": 634.22,
"jpIsHit": false,
"buyFreeGame": false
}
```

參數名稱	數據類型	描述
PlayerId	string	遊戲平台玩家 ID
Xid	string	請求唯一代碼
GameId	string	遊戲編號
BetAmount	number	本輪投注金額（實際金額需除以 10000；遊戲貨幣比例為 1:10000）
WinAmount	number	本輪中獎金額（實際金額需除以 10000；遊戲貨幣比例為 1:10000）
StartTimestamp	number	開始投注時間，Unix 13 位數字
EndTimestamp	number	結束投注時間，Unix 13 位數字
Sign	string	MD5 驗證碼（參考 MD5 驗證說明）
EcSiteId	string	遊戲平台中的代理平台 ID
Nickname	string	玩家暱稱
PlayerProfit	number	玩家的淨利潤（本輪中獎金額減去本輪投注金額；實際金額需除以 10000；遊戲貨幣比例為 1:10000）
SingleWalletType	number	內部使用，平台不使用

GameRecordId	string	本輪遊戲訂單號. 只會回空字串，只會在恢復失敗的下注時候提供新的GameRecordId
History	object	遊戲中每筆投注的詳細信息
jpXid	string	jp ID
jpTimestamp	number	jp 中獎時間
jpLevel	number	jp 中獎級別 0: JackpotPlatinum, 1: JackpotGold, 2: JackpotSilver
jpHitValue	number	jp 中獎金額
jp_Value_New_J1	number	現在 jp 值 JackpotPlatinum
jp_Value_New_J2	number	現在 jp 值 JackpotGold
jp_Value_New_J3	number	現在 jp 值 JackpotSilver
jpIsHit	bool	是否中 jp
buyFreeGame	bool	是否購買免費遊戲

## Response

### ■ Body:

```
{
  "resCode": "0"
}
```

欄位	屬性	說明
resCode	string	0: 成功、1: 失敗

## D.小遊戲下注預先扣款

針對部分小遊戲下注採取兩階段，"預先扣款"以及"派彩結果"

如代理商為「單一錢包」且授權遊戲編號為「414」「416」「419」則需要實作此API

## Reuquest

- Method: POST
- URL: 平台提供自定義接口位置
- Headers : Content-Type:application/json
- Body:

```
{
  "PlayerId": "12345",
  "Xid": "sdf32-sdf23r-sf32r",
  "GameId": "15",
  "BetAmount": 100000(Game currency uses a 1:10000 ratio),
  "WinAmount": 0,
  "StartTimestamp": 123141245214,
  "Sign":"58df2b36d1edb86c37f6f5cb363587b0",
  "EcSiteId": "12",
  "Nickname": "player1",
  "SingleWalletType": 1,
  "GameRecordId": "GR-234fsaf-faaf3r3",
}
```



參數名稱	數據類型	描述
PlayerId	string	遊戲平台玩家 ID
Xid	string	請求唯一代碼
GameId	string	遊戲編號
BetAmount	number	本輪投注金額（實際金額需除以 10000；遊戲貨幣比例為 1:10000）
WinAmount	number	固定為0
StartTimestamp	number	開始投注時間，Unix 13 位數字
Sign	string	MD5 驗證碼（參考 MD5 驗證說明）
EcSiteId	string	遊戲平台中的代理平台 ID
Nickname	string	玩家暱稱
SingleWalletType	number	內部使用，平台不使用
GameRecordId	string	本輪遊戲訂單號

## Response

- Body:

```
{
  "resCode": "0",
  "data": {
    "afterBalance": 50000,
    "beforeBalance": 80000,
  }
}
```

參數名稱	數據類型	描述
resCode	string	0: 成功 1: 失敗
afterBalance	number	玩家處理後的餘額（在傳輸之前需乘以 10000）
beforeBalance	number	玩家處理前的餘額（在傳輸之前需乘以 10000）

## E.小遊戲下注派彩結果

針對部分小遊戲下注採取兩階段，"預先扣款"以及"派彩結果"

如代理商為「單一錢包」且授權遊戲編號為「414」「416」「419」則需要實作此API

### Reuquest

- Method: POST
- URL: 平台提供自定義接口位置
- Headers : Content-Type:application/json
- Body:

```
{
  "PlayerId": "12345",
  "Xid": "sdf32-sdf23r-sf32r",
  "GameId": "15",
  "BetAmount": 0,
  "WinAmount": 300000(Game currency uses a 1:10000 ratioT),
  "Sign":"58df2b36d1edb86c37f6f5cb363587b0",
  "EndTimestamp": 123141245214,
  "EcSiteId": "12",
  "Nickname": "player1",
  "PlayerProfit": 200000,
  "SingleWalletType": 1,
  "GameRecordId": "GR-234fsaf-faaf3r3"
}
```

參數名稱	數據類型	描述
PlayerId	string	遊戲平台玩家 ID
Xid	string	請求唯一代碼
GameId	string	遊戲編號
BetAmount	number	固定為0
WinAmount	number	本輪中獎金額（實際金額需除以 10000；遊戲貨幣比例為 1:10000）
EndTimestamp	number	結束投注時間，Unix 13 位數字
Sign	string	MD5 驗證碼（參考 MD5 驗證說明）
EcSiteId	string	遊戲平台中的代理平台 ID
Nickname	string	玩家暱稱
PlayerProfit	number	玩家的淨利潤（本輪中獎金額減去本輪投注金額；實際金額需除以 10000；遊戲貨幣比例為 1:10000）
SingleWalletType	number	內部使用，平台不使用
GameRecordId	string	本輪遊戲訂單號

## Response

- Body:

```
{
  "resCode": "0",
  "data": {
    "afterBalance": 50000,
    "beforeBalance": 80000,
  }
}
```

參數名稱	數據類型	描述
resCode	string	0: 成功 1: 失敗
afterBalance	number	玩家處理後的餘額（在傳輸之前需乘以 10000）
beforeBalance	number	玩家處理前的餘額（在傳輸之前需乘以 10000）

## 2. 第三方遊戲額外提供的接口

### A.查詢失敗或是超時的下注

查詢因為 API 超時或是其他因素造成失敗的下注結果有哪些，可以進一步協助玩家將遊戲紀錄恢復

#### Reuquest

- Method: GET
- URL: `{{api_server_url}}/SingleWalletGetRecord?playerId="123123"&startTime=12312321414&endTime=123341424`
- Headers : Content-Type:application/json

params	屬性	說明
playerId	string	會員 Id
startTime	number	搜尋下注開始的時間(非必要填選)
endTime	number	搜尋下注結束的時間(非必要填選)

#### Response

- Body:

```
{
  "statusCode": "2000001",
  "message": "SUCCESS",
  "data": [Array of Failed Bet Data]
}
```

欄位	屬性	說明
statusCode	string	"2000001": 成功、其他: 失敗原因。
message	string	API 訊息
data	array	失敗的下注資料陣列

## B.恢復失敗的下注

當玩家下注在第三方顯示失敗，但是在平台方有成功的時候，可以根據 `GameRecordId` 來強制要求第三方遊戲將此遊戲紀錄標記成下注成功，讓玩家的遊戲紀錄與平台方達成一致的結果

### Reuquest

- Method: POST
- URL: `{{api_server_url}}/SingleWalletRestoreRecord`
- Headers : Content-Type:application/json
- Body:

```
{
  "gameRecordId": "12ff-345asf-fas32ff"
}
```

欄位	屬性	說明
gameRecordId	string	注單 id

## Response

■ Body:

```
{
  "statusCode": "2000001",
  "message": "SUCCESS"
}
```

欄位	屬性	說明
statusCode	string	"2000001": 成功、其他: 失敗原因。
message	string	API 訊息

## 3. JP 說明

本遊戲供應商的 JP 是分別不同包網(平台)內的所有老虎機共用，不同包網(平台)的老虎機都是各別分開的，目前遊戲的 JP 佔遊戲整體的 RTP 大約為 0.3%，原始遊戲的 RTP 如果是 97%，那開啟 JP 之後會變成 97.3% 提升。

JP 的獎項是由提高 RTP 產出，並非無條件提供的獎金，所以月結帳方式保持玩家輸贏的結帳方案。

# 附件

## Response & Error code 對應表

Code	錯誤訊息
2000001	成功，有回傳資料
2040001	成功，不需要回傳資料(no content)
2040404	成功，無資料
4030001	錯誤，代理權限驗證錯誤
4000002	錯誤，參數簽名驗證錯誤
4030003	錯誤，代理幣別驗證錯誤
4000001	錯誤，參數相關驗證錯誤
4040001	錯誤，參數資料找不到對應內容
4040099	錯誤，會員資料重複
4040121	錯誤，無代理資料
4040001	錯誤，無相關資料
4030104	錯誤，金額不足
4030105	錯誤，存款超過限制
4030106	錯誤，提款餘額不足
4030102	錯誤，重複登陸
5000001	錯誤，系統執行錯誤
5000002	錯誤，系統維護中

## Error message 對應表

Message	錯誤訊息
RES_BAD_PARAMS	參數錯誤
System access limit is 100000000000 per transaction	超過一次存取限制金額
EcSite Data Not Found	沒有這個營運商
Player Data Not Found	沒有這個玩家
PLAYER_IN_GAME	玩家正在玩遊戲
INSUFFICIENT_FUNDS1	玩家餘額不足(餘額不足)
INSUFFICIENT_FUNDS2	玩家餘額不足(玩家還在遊戲中，不允許轉帳)
The server is busy, please try again later	伺服器忙碌中，請稍後再試
RES_SYSTEM_EXCEPTION	未知錯誤

# 語系說明

## 語系對應表

代碼	說明
en-us	英文
zh-tw	繁體中文
zh-cn	簡體中文
ko-kr	韓文
ja-jp	日語
vi-vn	越南文
th-th	泰文



# Category 說明

Category對應表

代碼	說明
0	老虎機
2	捕魚機
9	加密貨幣
99	其他

# History 說明

Slot 圖標對應表

Code	Symbol
50	WILD
100	FWILD
20	SCATTER
1	H1
2	H2
3	H3
4	H4
5	H5
6	H6
7	H7
8	H8
9	H9
11	N1
12	N2
13	N3
14	N4
15	N5
16	N6
17	N7
18	N8
19	N9

## Slot 消除類 (深海,埃及,瑪雅)

陣列裡面為該次的遊戲次數,如果有免費遊戲,該陣列裡面會有對多筆紀錄

*history* 裡的物件結構

參數名稱	參數型態	參數說明
AllReels	array	盤面結果
RandomNumList	array	轉輪表位置index
LongWildIndex	integer	滿軸位置 如果是5代表本次旋轉沒有滿軸
TotalWinBonus	float	總派彩金額
EliminateList	array	消除結果清單

*EliminateList* 裡物件的結構

參數名稱	參數型態	參數說明
WinPos	array	消除位置對應表
IsEliminate	boolean	是否為消除
NextReels	array	下次旋轉盤面結果
WinBonus	float	該次旋轉派彩金額
WinLineList	array	消除圖標結果清單

*WinLineList* 裡物件的結構

參數名稱	參數型態	參數說明
SpecialMultiple	integer	倍率
X	integer	幾條線
WinNum	integer	圖標
WinBonus	float	該次消除派彩金額
LineNum	integer	幾連線

history response example

```
[
  {
    "AllReels": [
      [
        14,
        11,
        13
      ],
      [
        14,
        3,
        13
      ],
      [
        3,
        13,
        4
      ],
      [
        4,
        12,
        20
      ],
      [
```











## Slot 連線類 (電音,封神)

resultList如果有免費遊戲,該陣列裡面會有對多筆紀錄

## history 裡的物件結構

參數名稱	參數型態	參數說明
totalBonus	float	總派彩金額
totalBet	float	投注額
resultList	array	遊戲結果清單, 有free game 會有多筆
allReels	array	對應resultList,紀錄遊戲盤面

## resultList 裡物件的結構

參數名稱	參數型態	參數說明
anyNumPos	string	百搭位置 Ex:"00000"
randomNumList	boolean	轉輪表位置index
totalWinBonus	array	該次旋轉派彩金額
WinLineList	array	消除圖標結果清單

## WinLineList 裡物件的結構

參數名稱	參數型態	參數說明
lineIndex	integer	連線位置
winNum	integer	圖標
winBonus	float	派彩
lineNum	integer	幾連線
specialMultiple	integer	倍率

### history response example

```
{
  "totalBonus": 0,
  "totalBet": 1.8,
  "resultList": [
    {
      "anyNumPos": "00000",
      "randomNumList": [
        1,
        1,
        1,
        1,
        1
      ],
      "totalWinBonus": 0,
      "winLineList": [
        {
          "lineIndex": 5,
          "winNum": 3,
          "winBonus": 1.8,
          "lineNum": 3,
          "specialMultiple": 1
        }
      ]
    }
  ]
}
```



富貴捕魚 魚隻結果對應表

結果	魚種
101	飛魚
102	小丑魚
103	關刀
104	藍倒吊
105	比目魚
106	獅子魚
107	龍蝦
108	河豚
109	海馬
110	海龜
111	燈籠魚
112	企鵝
201	魔鬼魚
202	鰻魚
203	美人魚
204	河豚
205	蟾蜍
206	巨型蟾蜍
207	虎鯨
301	八爪魚
302	玄武龜
303	青龍
401	八爪魚Boss

402	玄武龜Boss
403	青龍Boss
501	瘋狂鑽頭
502	粒子激光
503	超級炸彈
504	神秘黑洞
505	雷神之力
506	幸運轉盤
666	Jackpot

富貴捕魚 子彈類對應表

結果	子彈
0	一般子弹
401	八爪鱼Boss大招
402	玄武龟Boss大招
403	青龙Boss大招
501	疯狂钻头
502	粒子激光
503	超级炸弹
504	神秘黑洞
505	雷神之力
506	幸运转盘

富貴捕魚

history 的物件結構(為子彈資料組成的陣列)

子彈資料結構,每一筆遊戲記錄最多只會紀錄100發子彈資訊

參數名稱	參數型態	參數說明
ante	integer	子彈金額
skillType	integer	技能類型(參考子彈類型對應表)
deadFishes	array	死魚(這發子彈打死了多少魚種)

deadFishes 資料結構

參數名稱	參數型態	參數說明
fishId	string	魚隻流水編號
fishType	integer	魚隻類型(參考魚隻結果對應表)
position	array	魚隻死掉位置
luckyMultiples	array	幸運轉輪的倍率
luckyReelIdxs	array	幸運轉輪的index
odds	integer	魚隻賠率
pay	integer	賠付金額
skillId	string	技能流水編號
skillType	integer	技能類型(參考子彈類型對應表)

history response example

```
[
  {
    "ante": 1000,
    "skillType": 0,
    "deadFishes": [
      {
        "fishId": "f107-c99sh7d1npb7c2c4o9bg",
        "fishType": 107,
        "position": [
```



·  
·  
]

# BetInfo說明

## 視訊真人龍虎鬥

BetInfo 值	說明	韓文對應
1	Odd (單數)	홀
2	Even (雙數)	짝
3	Under (隱藏卡片範圍 1 到 5)	히든카드 언더
4	Over (隱藏卡片範圍 6 到 10)	히든카드 오버
5	Black (隱藏卡片顏色為黑色)	검정
6	Red (隱藏卡片顏色為紅色)	빨강
7	Spade (隱藏卡片花色為黑桃)	스페이드
8	Diamond (隱藏卡片花色為方塊)	다이아몬드
9	Heart (隱藏卡片花色為紅心)	하트
10	Club (隱藏卡片花色為梅花)	클로버

## 視訊真人百家樂

BetInfo 值	說明	韓文對應
1	Player (閒家)	플레이어
2	Tie (平局)	타이
3	Banker (莊家)	뱅커



體育足球

BetInfo 值	說明	韓文對應
1	Home	
2	Away	
3	X	
4	Over	
5	Under	

體育籃球

BetInfo 值	說明	韓文對應
1	Home	
2	Away	

體育賽馬

BetInfo 值	說明	韓文對應
1	馬 1號	
2	馬 2號	
3	馬 3號	
4	馬 4號	
5	馬 5號	
6	馬 6號	
7	馬 7號	
8	馬 8號	
9	馬 9號	
10	馬 10號	
11	馬 11號	
12	馬 12號	

體育賽狗

BetInfo 值	說明	韓文對應
1	狗 1號	
2	狗 2號	
3	狗 3號	
4	狗 4號	
5	狗 5號	
6	狗 6號	
7	狗 7號	
8	狗 8號	
9	狗 9號	
10	狗 10號	
11	狗 11號	
12	狗 12號	

金豬

BetInfo 值	說明	韓文對應
1	Odd (單數)	홀
2	Even (雙數)	짝
3	Left (左)	좌
4	Right (右)	우
5	Row3 (3 條)	3줄
6	Row4 (4 條)	4줄
7	Row3Left (左 3 條)	좌 3줄
8	Row4Left (左 4 條)	좌 4줄
9	Row3Right (右 3 條)	우 3줄
10	Row4Right (右 4 條)	우 4줄

獅子兔子

BetInfo 值	說明	韓文對應
1	Lion (獅子)	라이언
2	Mooji (兔子)	무지

龍虎鬥(Tiger Dragon)

BetInfo 值	說明
1	龍
2	虎
3	合

上和下(Up & Down)

BetInfo 值	說明
1	Start 上
2	Start 下
3	End 上
4	End 下
5	Result 上上
6	Result 上下
7	Result 下上
8	Result 下下

天梯(Paricle Ladder)

BetInfo 值	說明
1	左
2	右
3	Row3
4	Row4
5	奇數
6	偶數

射龍門(Pently Kick)

BetInfo 值	說明
1	進球
2	沒進球
3	進球左
4	進球中
5	進球右
6	防守左
7	防守中
8	防守右

賽馬(Horse Racing)

BetInfo 值	說明
1	1號
2	2號
3	3號
4	4號

體育遊戲資料結構說明：

下注產生的時候送給貴司的單一錢包請求的物件內有一個欄位 `history` 根據遊戲有不同的結構

體育遊戲 ID

體育名稱	gameld
足球	890
籃球	891
賽馬	892
賽狗	893

## 足球 890 主結構：Sport1Bets

欄位名稱	類型	JSON 鍵	說明	範例值
Key	string	_key	唯一鍵值，用於標識此筆紀錄	"uniqueKey789"
RoundEndTimestamp	uint64	roundEndTimestamp	所有串關賽事的結束時間 (Unix 時間)	1627891234
BetTypes	[]string	betTypes	下注類型的陣列	["注單1", "注單2"]
CompletedBetTypes	[]string	completedBetTypes	已完成結帳的 betTypes	[""]
RoundIds	[]string	roundIds	所有串關的賽事 ID	["R456", "R789"]
PlayerId	string	playerId	玩家 ID	"player789"
Nickname	string	nickname	玩家暱稱	"靜香"
Bets	[]*Sport1Bet	bets	下注單的陣列	見 Sport1Bet
TotalBetAmount	float64	totalBetAmount	總下注金額(金額都要除10000)	1000000
TotalWinAmount	float64	totalWinAmount	總獲得金額(金額都要除10000)	1800000
CreateAt	int64	createAt	紀錄建立時間 (Unix 13碼)	1627891234567
ExpectedWinAmount	float64	ExpectedWinAmount	預計贏的金額 (小遊戲專用)(金額都要除10000)	2400000
GamePid	string	gamePid	遊戲子編號 (專用於 888 遊戲)	"G888PID"
Gameld	string	gameId	遊戲編號	"890"
WinResult	int	WinResult	贏的結果 (0: 尚未開獎, 1: 贏, 2: 不贏, 3: 和局)	0
Xid	string	Xid	UUID，用於唯一標識	"uuid789"
StartTimestamp	uint64	StartTimestamp	下注時間 (Unix 時間)	1627890000
EndTimestamp	uint64	EndTimestamp	結束時間 (Unix 時間)	1627895000
BetOdds	float64	BetOdds	下注賠率 (小遊戲專用)	2.4



子結構：Sport1Bet

欄位名稱	類型	JSON 鍵	說明	範例值
Xid	string	xid	UUID，用於唯一標識	"uniqueKey789"
BetType	string	betType	下注類型	"單注1"
LeagueName	string	leagueName	聯盟名稱	"聯盟A"
RoundId	string	roundId	賽事 ID	"R456"
DateTime	int64	dateTime	下注時間 (Unix 13碼)	1627891234567
W1BetAmount	float64	w1BetAmount	第一隊下注金額(金額都要除10000)	500000
W1Odds	float64	w1odds	第一隊下注當時的倍率	1.2
W2BetAmount	float64	w2BetAmount	第二隊下注金額(金額都要除10000)	0
W2Odds	float64	w2odds	第二隊下注當時的倍率	2.0
XBetAmount	float64	xBetAmount	和局的下注金額(金額都要除10000)	0
XOdds	float64	xodds	和局下注當時的倍率	0.0
OverBetAmount	float64	overBetAmount	over 下注金額(金額都要除10000)	0
OverOdds	float64	overOdds	over 下注當時的倍率	0.0
UnderBetAmount	float64	underBetAmount	under 下注金額(金額都要除10000)	0
UnderOdds	float64	underOdds	under 下注當時的倍率	0.0
PlayerId	string	playerId	玩家 ID	"player789"
Nickname	string	nickname	玩家暱稱	"靜香"
BetInfo	string	BetInfo	下注的詳細資訊	"1"
BetOdds	float64	BetOdds	下注賠率	2.4
GameResult	string	GameResult	遊戲最終結果 (小遊戲專用)	"勝利"

# JSON 範例

```
{
  "_key": "uniqueKey789",
  "roundEndTimestamp": 1627891234,
  "betTypes": ["單注1", "單注2"],
  "completedBetTypes": [""],
  "roundIds": ["456", "789"],
  "playerId": "player789",
  "nickname": "靜香",
  "bets": [
    {
      "Xid": "uniqueKey789",
      "betType": "單注1",
      "leagueName": "聯盟A",
      "roundId": "456",
      "dateTime": 1627891234567,
      "w1BetAmount": 500000,
      "w1Odds": 2.4,
      "w2BetAmount": 0,
      "w2Odds": 2.0,
      "xBetAmount": 0.0,
      "xOdds": 4.0,
      "overBetAmount": 0.0,
      "overOdds": 5.0,
      "underBetAmount": 0.0,
      "underOdds": 0.0,
      "playerId": "player789",
      "nickname": "靜香",
      "BetInfo": "1",
      "BetOdds": 2.4,
      "GameResult": "-1"
    },
    {
      "Xid": "uniqueKey789",
      "betType": "單注2",
```



# 籃球 891 主結構：

欄位名稱	類型	JSON 鍵	說明	範例值
Key	string	_key	唯一鍵值，用於標識此筆紀錄	"uniqueKey456"
RoundEndTimestamp	uint64	roundEndTimestamp	所有串關賽事的結束時間 (Unix 時間)	1627891234
BetTypes	[]string	betTypes	下注類型的陣列	["注單1", "注單2"]
CompletedBetTypes	[]string	completedBetTypes	已完成結帳的 betTypes	[""]
RoundIds	[]string	roundIds	所有串關賽事的 ID	["123", "789"]
PlayerId	string	playerId	玩家 ID	"player123"
Nickname	string	nickname	玩家暱稱	"大雄"
Bets	[]*Sport2Bet	bets	下注單的陣列	見 Sport2Bet
TotalBetAmount	float64	totalBetAmount	總下注金額(金額都要除10000)	1000000
TotalWinAmount	float64	totalWinAmount	總獲得金額(金額都要除10000)	2000000
CreateAt	int64	createAt	紀錄建立時間 (Unix 13碼)	1627891234567
ExpectedWinAmount	float64	ExpectedWinAmount	預計贏的金額 (小遊戲專用)(金額都要除10000)	2400000
GamePid	string	gamePid	遊戲子編號 (專用於 888 遊戲)	"G888PID"
Gameld	string	gameId	遊戲編號	"891"
WinResult	int	WinResult	贏的結果 (0: 尚未開獎, 1: 贏, 2: 不贏, 3: 和局)	1
Xid	string	Xid	UUID，用於唯一標識	"uuid456"
StartTimestamp	uint64	StartTimestamp	下注時間 (Unix 時間)	1627890000
EndTimestamp	uint64	EndTimestamp	結束時間 (Unix 時間)	1627895000
BetOdds	float64	BetOdds	下注賠率 (小遊戲專用)	2.4

# 子結構：Sport2Bet

欄位名稱	類型	JSON 鍵	說明	範例值
Xid	string	xid	UUID，用於唯一標識	"uniqueKey456"
BetType	string	betType	下注類型	"單注1"
LeagueName	string	leagueName	聯盟名稱	"聯盟A"
RoundId	string	roundId	賽事 ID	"R789"
DateTime	int64	dateTime	下注時間 (Unix 13碼)	1627891234567
W1BetAmount	float64	w1BetAmount	第一隊下注金額(金額都要除10000)	500000
W1Odds	float64	w1Odds	第一隊下注當時的倍率	1.2
W2BetAmount	float64	w2BetAmount	第二隊下注金額(金額都要除10000)	500000
W2Odds	float64	w2Odds	第二隊下注當時的倍率	2.0
PlayerId	string	playerId	玩家 ID	"player123"
Nickname	string	nickname	玩家暱稱	"大雄"
BetInfo	string	BetInfo	下注的詳細資訊	"1"
BetOdds	float64	BetOdds	下注賠率	2.4
GameResult	string	GameResult	遊戲最終結果 (小遊戲專用)	"-1"

## JSON 範例

```
{
  "_key": "uniqueKey456",
  "roundEndTimestamp": 1627891234,
  "betTypes": ["單注1", "單注2"],
  "completedBetTypes": [],
  "roundIds": ["123", "789"],
```



```
"totalBetAmount": 1000000,  
"totalWinAmount": 0,  
"createAt": 1627891234567,  
"ExpectedWinAmount": 4800000,  
"gamePid": "G888PID",  
"gameId": "891",  
"WinResult": 0,  
"Xid": "uniqueKey456",  
"StartTimestamp": 1627890000,  
"EndTimestamp": 1627895000,  
"BetOdds": 4.8  
}
```

賽馬 892 主結構：



欄位名稱	類型	JSON 鍵	說明	範例值
Key	string	_key	唯一鍵值，用於標識此筆紀錄	"uniqueKey123"
RoundEndTimestamp	uint64	roundEndTimestamp	所有串關賽事的結束時間 (Unix 時間)	1627891234
BetTypes	[]string	betTypes	下注類型的陣列	["注單1", "注單2"]
CompletedBetTypes	[]string	completedBetTypes	已完成結帳的 betTypes	[""]
RoundIds	[]string	roundIds	所有串關賽事的 ID	["123", "456"]
PlayerId	string	playerId	玩家 ID	"player1"
Nickname	string	nickname	玩家暱稱	"小明"
Bets	[]*Sport3Bet	bets	下注單的陣列	見 Sport3Bet
TotalBetAmount	float64	totalBetAmount	總下注金額(金額都要除10000)	1000000
TotalWinAmount	float64	totalWinAmount	總獲得金額(金額都要除10000)	1500000
CreateAt	int64	createAt	紀錄建立時間 (Unix 13碼)	1627891234567
ExpectedWinAmount	float64	ExpectedWinAmount	預計贏的金額 (小遊戲專用)(金額都要除10000)	120.0
GamePid	string	gamePid	遊戲子編號 (專用於 888 遊戲)	"G888PID"
GameId	string	gameId	遊戲編號	"892"
WinResult	int	WinResult	贏的結果 (0: 尚未開獎, 1: 贏, 2: 不贏, 3: 和局)	1
Xid	string	Xid	UUID，用於唯一標識	"uuid123"
StartTimestamp	uint64	StartTimestamp	下注時間 (Unix 時間)	1627890000
EndTimestamp	uint64	EndTimestamp	結束時間 (Unix 時間)	1627895000
BetOdds	float64	BetOdds	下注賠率 (小遊戲專用)	2.5

# 子結構：Sport3Bet

欄位名稱	類型	JSON 鍵	說明	範例值
Xid	string	Xid	UUID，用於唯一標識	"uniqueKey123"
BetType	string	betType	下注類型	"單注1"
LeagueName	string	leagueName	聯盟名稱	"聯盟A"
RoundId	string	roundId	賽事 ID	"R789"
DateTime	int64	dateTime	下注時間 (Unix 13碼)	1627891234567
BetAmount	float64	betAmount	下注金額(金額都要除10000)	500000
HorseName	float64	horseName	下注馬的名字	"Champion"
Horseld	int8	horseId	下注馬的編號	1
HorseOdds	float64	horseOdds	當時下注倍率	1.8
PlayerId	string	playerId	玩家 ID	"player1"
Nickname	string	nickname	玩家暱稱	"小明"
BetInfo	string	BetInfo	下注的詳細資訊	"1"
BetOdds	float64	BetOdds	下注賠率	2.0
GameResult	string	GameResult	遊戲最終結果 (小遊戲專用)	"-1"

## JSON 範例

```
{
  "_key": "uniqueKey123",
  "roundEndTimestamp": 1627891234,
  "betTypes": ["單注1", "單注2"],
  "completedBetTypes": [],
  "roundIds": ["789", "456"],
```



```
"totalBetAmount": 800000,  
"totalWinAmount": 0,  
"createAt": 1627891234567,  
"ExpectedWinAmount": 48000000,  
"gamePid": "G888PID",  
"gameId": "892",  
"WinResult": 0,  
"Xid": "uniqueKey123",  
"StartTimestamp": 1627890000,  
"EndTimestamp": 1627895000,  
"BetOdds": 6.0  
}
```

賽狗 893 主結構：

欄位名稱	類型	JSON 鍵	說明	範例值
Key	string	_key	唯一鍵值，用於標識此筆紀錄	"abc123"
RoundEndTimestamp	uint64	roundEndTimestamp	所有串關賽事的結束時間 (Unix 時間)	1627891234
BetTypes	[]string	betTypes	下注類型的陣列	["注單1", "注單2"]
CompletedBetTypes	[]string	completedBetTypes	已完成結帳的 betTypes	[""]
RoundIds	[]string	roundIds	所有串關賽事的 ID	["123", "456"]
PlayerId	string	playerId	玩家 ID	"player1"
Nickname	string	nickname	玩家暱稱	"小明"
Bets	[]*Sport4Bet	bets	下注單的陣列	見 Sport4Bet
TotalBetAmount	float64	totalBetAmount	總下注金額(金額都要除 10000)	1000000
TotalWinAmount	float64	totalWinAmount	總獲得金額(金額都要除 10000)	1500000
CreateAt	int64	createAt	紀錄建立時間 (Unix 13碼)	1627891234567
ExpectedWinAmount	float64	ExpectedWinAmount	預計贏的金額 (小遊戲專用)	120
GamePid	string	gamePid	遊戲子編號 (專用於 888 遊戲)	"G888PID"
Gameld	string	gameId	遊戲編號	"893"
WinResult	int	WinResult	贏的結果 (0: 尚未開獎, 1: 贏, 2: 不贏, 3: 和局)	1
Xid	string	Xid	UUID，用於唯一標識	"uuid123"
StartTimestamp	uint64	StartTimestamp	下注時間 (Unix 時間)	1627890000
EndTimestamp	uint64	EndTimestamp	結束時間 (Unix 時間)	1627895000
BetOdds	float64	BetOdds	下注賠率 (小遊戲專用)	2.5

# 子結構：Sport4Bet

欄位名稱	類型	JSON 鍵	說明	範例值
Xid	string	Xid	UUID，用於唯一標識	"uuid456"
BetType	string	betType	下注類型	"單注"
LeagueName	string	leagueName	聯盟名稱	"聯盟A"
RoundId	string	roundId	賽事 ID	"R789"
DateTime	int64	dateTime	下注時間 (Unix 13碼)	1627891234567
BetAmount	float64	betAmount	下注金額(金額都要除10000)	500000
DogName	float64	dogName	下注狗的名字	"Lucky"
DogId	int8	dogId	下注狗的編號	3
DogOdds	float64	dogOdds	當時下注倍率	1.8
PlayerId	string	playerId	玩家 ID	"player1"
Nickname	string	nickname	玩家暱稱	"小明"
BetInfo	string	BetInfo	下注的詳細資訊	"3"
BetOdds	float64	BetOdds	下注賠率	2.0
GameResult	string	GameResult	遊戲最終結果 (小遊戲專用)	"-1： 尚未開獎"

## JSON 範例

```
{
  "_key": "uniqueKey123",
  "roundEndTimestamp": 1627891234,
  "betTypes": ["xxxx", "xxxx"],
  "completedBetTypes": [""],
  "roundIds": ["789", "456"],
```



```
"totalBetAmount": 800000,  
"totalWinAmount": 0,  
"createAt": 1627891234567,  
"ExpectedWinAmount": 40000000,  
"gamePid": "G888PID",  
"gameId": "893",  
"WinResult": 0,  
"Xid": "uniqueKey123",  
"StartTimestamp": 1627890000,  
"EndTimestamp": 1627895000,  
"BetOdds": 5.0  
}
```

## 遊戲 ID 890 的 BetInfo 說明

BetInfo 值	說明	韓文對應
1	Home	
2	Away	
3	X	
4	Over	
5	Under	

## 遊戲 ID 891 的 BetInfo 說明

BetInfo 值	說明	韓文對應
1	Home	
2	Away	



## 遊戲 ID 892 的 BetInfo 說明

BetInfo 值	說明	韓文對應
1	馬 1號	
2	馬 2號	
3	馬 3號	
4	馬 4號	
5	馬 5號	
6	馬 6號	
7	馬 7號	
8	馬 8號	
9	馬 9號	
10	馬 10號	
11	馬 11號	
12	馬 12號	

## 遊戲 ID 893 的 BetInfo 說明

BetInfo 值	說明	韓文對應
1	狗 1號	
2	狗 2號	
3	狗 3號	
4	狗 4號	
5	狗 5號	
6	狗 6號	
7	狗 7號	
8	狗 8號	
9	狗 9號	
10	狗 10號	
11	狗 11號	
12	狗 12號	