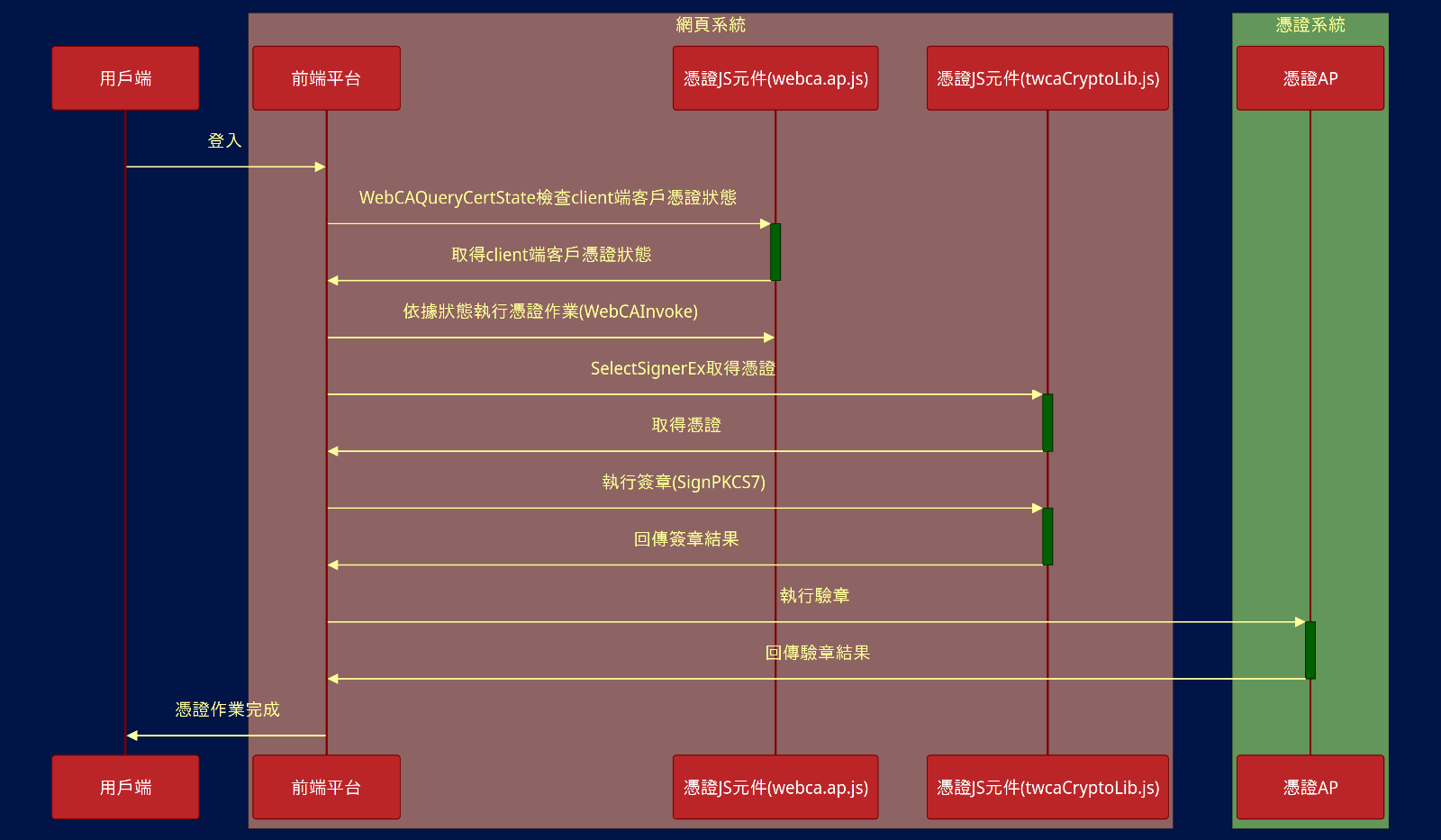
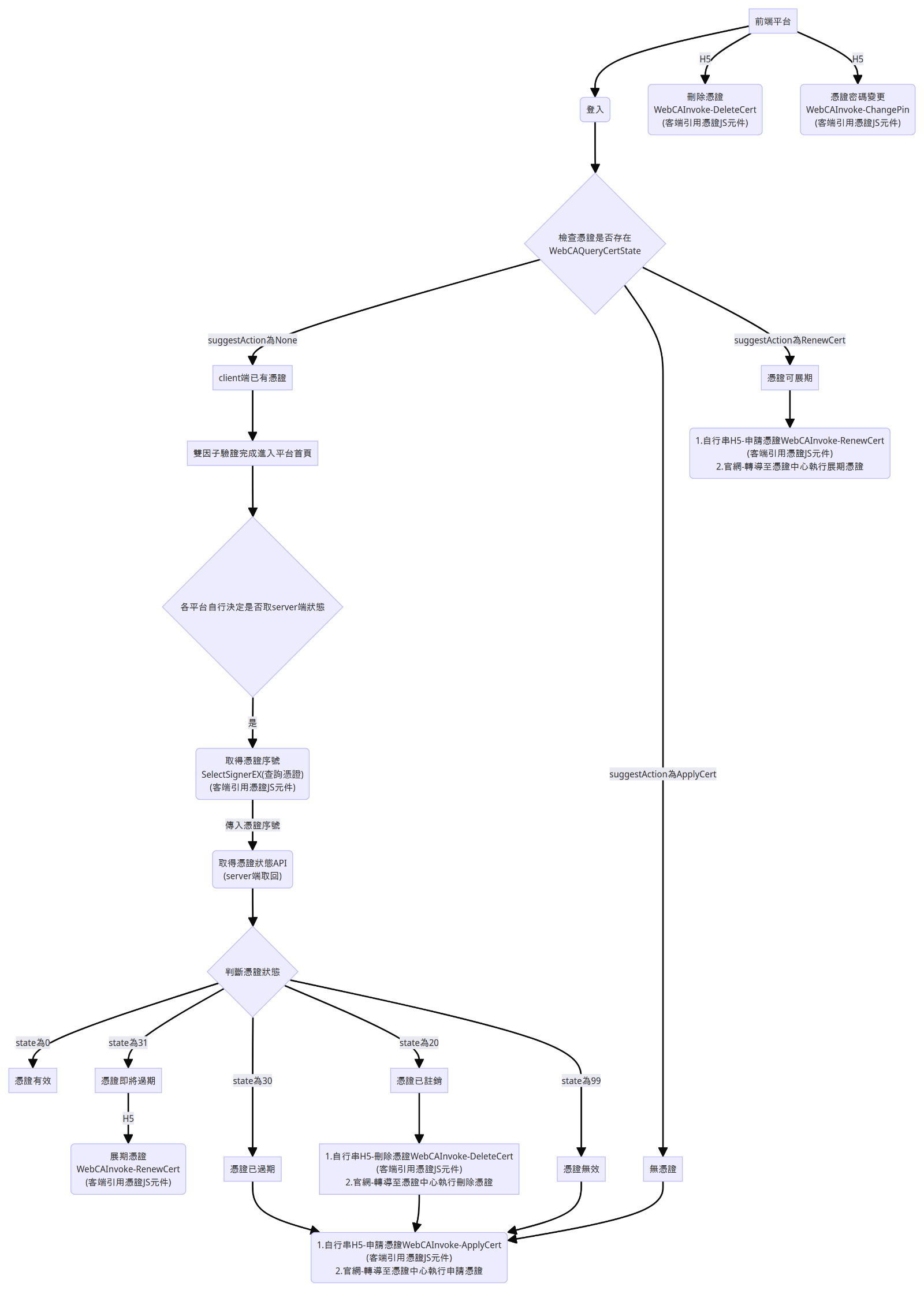
# H5憑證文件

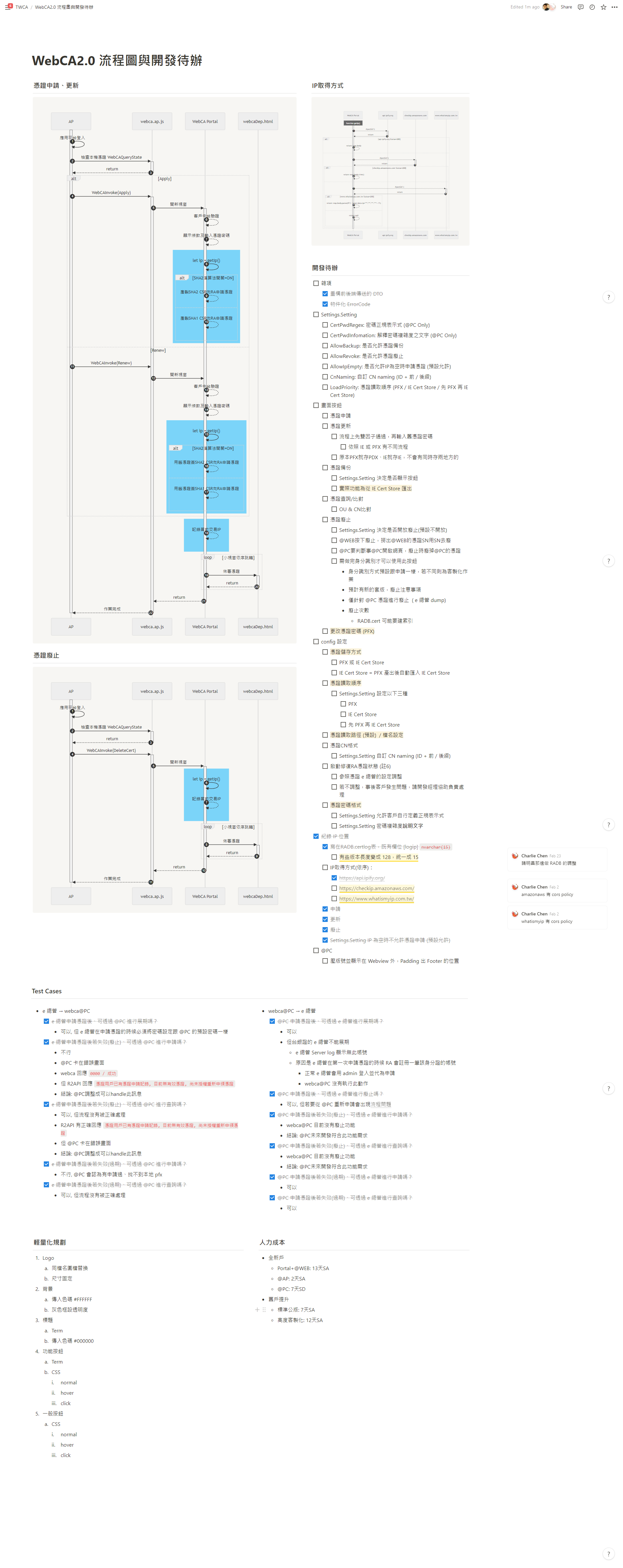
## 1.1憑證作業循序圖



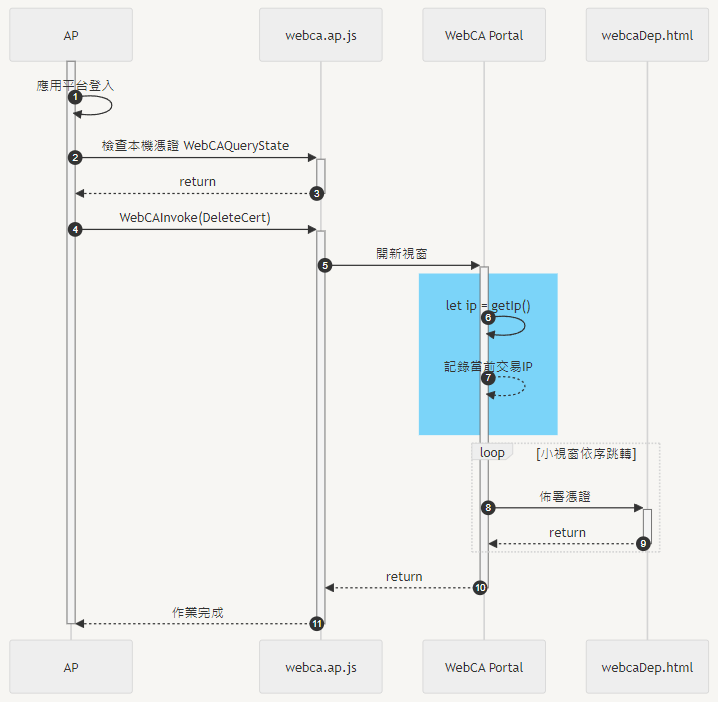
## 1.2前端平台判斷憑證流程圖



## 1.3憑證申請、更新



## 1.4憑證廢止



## 2.1 設定相關事項

1. 在網頁程式內，需引用「twcaCyptoLib.js」、「webca.ap.js」
2. 「webcaDepV2.html」需放在各個佈署的domain下，並確認每一個佈署網址都可以連線到，且路徑需提供給玉證進行設定，需設定跨來源資源共用(CORS)，以確保憑佈署功能可正常運作(參考7.1)。
3. 設定Hashkey的對應方式 **(請詢問憑證負責人)**

|  |  |
| --- | --- |
| HashKeyNo | HashKey |
|  |  |

1. 設定BusinessNo : 70781513 (玉山證券)

## 3.1 程式規格

### 3.1.1 WebCAQueryCertState

|  |  |  |
| --- | --- | --- |
| 說明 | | 查詢本機憑證狀態，並提供後續操作的建議 |
| 輸入 | **定義** | function WebCAQueryCertState (  id: string,  filter: string,  callback: function/string  ) |
| **說明** | id 使用者帳號  filter 憑證過濾條件，請參考3.1.1.1 輸入欄位說明  callback 完成後回應之Function Name |
| 輸出 | **定義** | function callback(code, msg, token, data)  data = {  userID: string,  isCertExist: boolean,  isReadyWebCA: boolean  isCertMultiType: boolean/undefined,  suggestAction: string  } |
| **說明** | userID 使用者帳號  isCertExist 使用者憑證是否存在  isReadyWebCA 是否該進行WebCA作業  isCertMultiType 是否為一瀏覽器一憑證  suggestAction 建議進行的WebCA操作，請見備註 |
| 錯誤碼 | | 5001 簽章元件不存在 |
| 備註 | | 1. 建議操作有以下三種類型：   ApplyCert 憑證新申請  RenewCert 憑證展期  None 不須進行WebCA操作   1. 若isCertExist 為false則isCertMultiType為undefined |

#### 3.1.1.1 輸入欄位說明

* 輸入欄位：Filter ( 請勿寫死在程式碼中 )
* 欄位規格：
* Filter 欄位：

//S\_CN=參數A,S\_OU=70781513-RA-ESUNSECEC01,S\_OU=RA-ESUNSECEC01,S\_O=參數B,S\_O=參數C,S\_C=TW//

* 參數說明：

參數A：身分證

參數B：

* 正式：Certification Service Provider
* 測試：Certificate Service Provider - Evaluation Only

參數C：

* 正式：TaiCA Secure CA
* 測試：TaiCA Secure CA - Evaluation Only
* 範例程式：

-正式環境

const CERT\_SO\_1: string = "Certificate Service Provider ";

const CERT\_SO\_2: string = "TaiCA Secure CA ";

-測試環境

const CERT\_SO\_1: string = "Certification Service Provider- Evaluation Only ";

const CERT\_SO\_2: string = "TaiCA Secure CA - Evaluation Only ";

-id 為身份證字號

export const genCAFilter = (id) => {

return `//S\_CN=${id},S\_OU=70781513-RA-ESUNSECEC01,S\_OU=RA-ESUNSECEC01,S\_O=${CERT\_SO\_1},S\_O=${CERT\_SO\_2},S\_C=TW//`

}

### 3.1.2 WebCAInvoke

|  |  |  |
| --- | --- | --- |
| 說明 | | 呼叫WebCA進行憑證操作 |
| 輸入 | **定義** | function WebCAInvoke (  businessNo: string,  apiVersion: string,  hashKeyNo: string,  verifyNo: string,  returnParams: string,  memberNo: string,  raFunc: string,  identifyNo: string,  windowURL: string,  webcaURL: string,  callback: function/string  ) |
| **說明** | businessNo 應用AP代號  apiVersion API版本，固定為1.0  hashKeyNo HashKey組別  verifyNo 驗證編號（需唯一，請使用yyyyMMddHHmmssffff）  **returnParams** 保留欄位(填入AuthToken)  memberNo 顧客身分證號  raFunc WebCA操作功能  identifyNo 驗證碼  windowURL 固定帶空值  webcaURL WebCA的網址  callback 完成後回應之Function Name |
| 輸出 | **定義** | function callback(code, msg, token, data)   1. ApplyCert、RenewCert、ChangePIN   data = {  sn: string,  cn: strgin,  ntb: string,  nta: string,  issdn: string,  subdn: string,  certfinger: string,  certb64: string  }   1. DeleteCert   data = {  userID: string  } |
| **說明** | sn 憑證序號  cn 憑證的主旨用戶名稱（common name）  ntb 憑證生效日yyyyMMddHHmmss（GMT+8）  nta 憑證效期yyyyMMddHHmmss（GMT+8）  issdn 憑證簽發者，各項以逗點區隔  subdn 憑證主旨，各項以逗點區隔  certfinger 憑證摘要值（SHA1 Hex）  certb64 X.509 Base64編碼之憑證  userID 使用者帳號 |
| 錯誤碼 | | 請參考錯誤碼列表 |
| 備註 | | 1. raFunc必須為以下其中之一   ApplyCert  RenewCert  ChangePIN  DeleteCert   1. 若執行失敗，data回傳空字串，不可轉為JSON物件 2. 因現行主流瀏覽器會封鎖自動彈出式視窗，造成WebCA憑證申請畫面被封鎖，請注意整合本項API時，必須有主動人為介入的動作，例如放在按鈕觸發function內：  |  |  | | --- | --- | | 1  2  3  4  5  6  7  8  9  10  11  12 | //瀏覽器會封鎖  WebCAInvoke(…);  //瀏覽器會封鎖  $(window).load(function() {  WebCAInvoke(…);  }  //可避免瀏覽器封鎖  button.onclick = () => {  WebCAInvoke(…);  }; | |

#### 3.1.2.1輸入欄位說明

* 輸入欄位：webcaURL
* 欄位規格：

測試環境 → <https://caweb.testesunsec.com.tw/WebCAV2/WebCA>

正式環境 → <https://caweb.esunsec.com.tw/WebCAV2/WebCA>

#### 3.1.2.2 若錯誤碼為7000：至少一個網址佈署失敗

此時會多回傳的資料多一組undeployableList資料

|  |  |
| --- | --- |
| 輸出 | data[undeployableList]: json array  [  {  location: string,  errMsg: string  },  {  location: string,  errMsg: string  }, …  ] |
| 說明 | undeployableList 無法佈署到的網域列表，為jsonArray  location 無法佈署到的網址  errMsg 無法佈署的原因 |

#### 3.1.2.3參數定義與認證碼計算方式，請注意驗證碼計算順序

資料格式請參閱

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 參數名稱 | 中文名稱 | 格式 | 可空 | 備註 |
| businessNo | 應用AP代號 | A(15) | 不可 | 依據WebCA資料庫設定 |
| apiVersion | API版本 | 9(2,1) | 不可 | 固定值"1.0" |
| hashKeyNo | HashKey組別 | 9(2) | 不可 | 依據WebCA資料庫設定 |
| verifyNo | 驗證編號 | A(50) | 不可 | 需唯一 請使用yyyyMMddHHmmssffff |
| returnParams | 應用AP自訂字串 | A(250) | 可 | 保留欄位  (填入AuthToken ) |
| memberNo | 顧客身分證號 | A(15) | 不可 | 1.唯一識別值  2.不須補滿 |
| raFunc | WebCA功能 | A(15) | 不可 | 必須為以下其中一項   1. ApplyCert 2. RenewCert 3. ChangePIN 4. DeleteCert |
| identifyNo | 驗證碼\* | A(256) | 不可 | SHA256 (  businessNo +  apiVersion +  hashKeyNo +  verifyNo +  returnParams +  memberNo +  raFunc +  hashKey  ) |

\* 驗證碼為使用UTF-16LE編碼後再計算SHA-256之小寫HEX字串。

### 3.1.3 錯誤碼列表

|  |  |  |
| --- | --- | --- |
| 錯誤碼 | 說明 | 處理方式 |
| 0000 | 成功 |  |
| GENERAL ERRORS | | |
| 1000 | 參數錯誤 | 請檢查輸入參數 |
| 1001 | 驗證編號重複 | 更新VerifyNo |
| 1002 | hashKeyNo錯誤 | 檢查前端輸入的HashKeyNo是否存在Settings.HashKey中 |
| 1003 | 驗證碼錯誤 | IdentifyNo拼裝錯誤，請檢查組成IdentifyNo的參數是否正確 |
| 1004 | JSON參數讀取失敗 | 對外請求回應的JSON格式解析錯誤，請檢查網路環境及請求位置是否正確 |
| 1006 | RAFunc參數錯誤 | 傳入的 RAFunc參數，不符合 "ApplyCert","ForceApplyCert", "RenewCert","ChangePIN", "DeleteCert" |
| 1010 | JSON參數設定失敗 | 類似1004，詳細原因請查看Log |
| 1011 | 產生驗證碼失敗 | 計算IdentifyNo失敗，檢查對應參數是否正確 |
| 1012 | 登入記錄寫入資料庫失敗 | 請檢查資料庫連線 |
| 1013 | 應用AP不合法 | 請檢查輸入的BusinessNo是否存在Settings.Business |
| 1015 | 更新資料庫失敗 | 於資料庫更新時，用BusinessNo， VerifyNo 查無對應的RAFuncLog |
| APPLICATION ERRORS | | |
| 2000 | 無登入記錄 | Log.APLoginLog 或Log.APVerifyLog 或Log.RAFunctionLog中無登入紀錄 |
| 2010 | 等待AP登入 | 尚未登入 |
| 2020 | AP登入失敗 | 登入失敗 |
| 2021 | 帳號密碼不可為空 | 帳號或密碼不可為空 |
| 2030 | AP登入失敗超過三次 | 登入失敗超過三次 |
| 2040 | 等待生日驗證 | 尚未進行生日驗證 |
| 2059 | 生日驗證資訊寫入資料庫失敗 | 請檢查APVerifyLog ，及資料庫狀態是否異常 |
| 2500 | HashKey 驗證成功 | hashKeyNo存在Settings.HashKey中 |
| USER EXCEPTIONS | | |
| 5070 | 使用者按下取消 | 使用者按下取消 |
| 5071 | 憑證密碼錯誤 | 憑證密碼錯誤 |
| 5075 | 密碼錯誤超過三次 | 憑證密碼錯誤超過三次 |
| 5079 | 使用者關閉視窗 | 使用者關閉視窗 |
| CERTIFICATE ERRORS | | |
| 5061 | (Windows Webview) 簽章失敗 | WebView憑證處理內部錯誤 |
| 5065 | (Windows Webview) 產生憑證請求檔失敗 | WebView憑證處理內部錯誤 |
| 5111 | (Windows Webview) 匯入憑證失敗 | WebView憑證處理內部錯誤 |
| 5112 | (Windows Webview) 讀取憑證失敗 | WebView憑證處理內部錯誤 |
| 6001 | 憑證作業失敗 | 查看webca.log和PSO4RA.log |
| 6002 | 取得用戶CN失敗 | 查看webca.log和PSO4RA.log |
| 6010 | 憑證已存在不可申請 | 查看webca.log和PSO4RA.log |
| 6020 | 憑證廢止失敗 | 查看webca.log和PSO4RA.log |
| EXPLORER ERRORS | | |
| 7001 | 瀏覽器儲存憑證失敗 | 瀏覽器憑證處理內部錯誤 ，請用戶清空localStorage再重新嘗試 |
| 7002 | 瀏覽器讀取憑證失敗 | 瀏覽器憑證處理內部錯誤 ，請用戶清空localStorage再重新嘗試 |
| 7003 | 瀏覽器刪除憑證失敗 | 瀏覽器憑證處理內部錯誤 ，請用戶清空localStorage再重新嘗試 |
| 7020 | 瀏覽器產生憑證請求檔失敗 | 瀏覽器憑證處理內部錯誤 ，請用戶清空localStorage再重新嘗試 |
| 7025 | 瀏覽器安裝憑證失敗 | 瀏覽器憑證處理內部錯誤 ，請用戶清空localStorage再重新嘗試 |
| 7050 | 瀏覽器簽章失敗 | 瀏覽器憑證處理內部錯誤 ，請用戶清空localStorage再重新嘗試 |
| 7500 | 瀏覽器封鎖彈出視窗 | AP端收到此錯誤碼應有輔助教學畫面協助使用者允許瀏覽器彈出視窗 |
| OTHERS | | |
| 2305 | 第三方個資驗證失敗 | AuthToken驗證失敗 |
| 9101 | 第三方個資驗證失敗 | 生日驗證失敗 |
| 9999 | 其他例外 |  |

## 4.1台網憑證安控元件(twcaCryptoLib.js)

### 4.1.1檢查及取得憑證資訊

透過此語法可檢查瀏覽器localstorage中有無憑證存在，若無憑證存在，則提示用戶進行憑證相關申請作業。

* SelectSignerEx(檢查及取得憑證)

|  |  |  |
| --- | --- | --- |
| 說明 | | 載入簽章憑證以準備簽章，或確認憑證是否存在 |
| 輸入 | **定義** | function SelectSignerEx (  signerCertFilter: string,  pfxPath: null,  pfxName: null,  userPIN: string / null,  compareDate: string,  dwFlags: int,  dwKeyUsage: 0,  trustToken: null,  callback: function/string  ) |
| **參數說明** | |
| 參數 | 值 |
| string  signerCertFilter | 測試環境帶以下參數(身分證字號依實際狀況為主)  //S\_CN=身分證,S\_OU=70781513-RA-ESUNSECEC01,S\_OU= RA-ESUNSECEC01,S\_O=Certificate Service Provider - Evaluation Only,S\_O=TaiCA Secure CA - Evaluation Only,S\_C=TW//  正式環境帶以下參數(身分證字號依實際狀況為主)  //S\_CN=身分證,S\_OU=70781513-RA-ESUNSECEC01,S\_OU=RA-ESUNSECEC01,S\_O=Certification Service Provider,S\_O=TaiCA Secure CA,S\_C=TW// |
| string  pfxPath | 保留欄位，請使用null |
| string pfxName | 保留欄位，請使用null |
| string  userPIN | 憑證密碼(帶null，則會跳出視窗讓用戶自行輸入密碼) |
| string compareDate | 空字串 |
| number dwFlags | 檢查憑證是否存在 → 0x04 | 0x08 | 0x00001000  取得憑證 → 0x04 | 0x08 | 0x00000000 |
| number dwKeyUsage | 0 |
| string trustToken | 保留欄位 (可為空字串或null不影響結果) |
| string callback | 完成後回應之Function Name |
| 輸出 | **定義** | function callback (code, msg, token, data)  data = {  sn: string,  cn: strgin,  ntb: string,  nta: string,  issdn: string,  subdn: string,  certfinger: string,  certb64: string  } |
| **回傳資料說明** | |
| 參數 | 值 |
| code | 執行結果代碼  0      成功  5070 使用者按下取消按鈕  5005 參數錯誤  5010 無憑證  5071 憑證密碼錯誤  5112 讀取憑證失敗  5001 一般錯誤 |
| msg | 執行結果訊息 |
| token | 保留 |
| data | Json格式回傳，回傳範例：  {"sn":"602198D3","cn":"A123456789","ntb":"20160602104023","nta":"20160616235959","issdn":"CN=TaiCA Secure CA -Evaluation Only, OU=Certification Service Provider-Evaluation Only, O=TAIWAN-CA.COM Inc., C=TW","subdn":"","certfinger":"","certb64":"MIIE1zCCA…"}  sn: 憑證序號  cn: 憑證的主旨用戶名稱（common name）,  ntb: 憑證生效日yyyyMMddhhmmss（GMT+8）,  nta: 憑證效期yyyyMMddhhmmss（GMT+8）,  issdn: 憑證簽發者，各項以逗點區隔,  subdn: 憑證主旨，各項以逗點區隔,  certfinger: 憑證摘要值（SHA1 Hex）  certb64: X.509 Base64編碼之憑證 |
| 備註 | | 1. 若執行失敗，data回傳空字串，不可轉為JSON物件。 2. compareDate優先權高於dwFlags的憑證效期選項，若compareDate為空字串憑證效期選項才有作用。 3. 若有多張憑證符合signerCertFilter條件，則依照dwFlag的憑證效期選項，選出最適合的一張。 |

### 4.1.2 簽章

* SignPKCS7(簽章)

→透過此語法，若localstorage中有憑證存在，則可依此語法產生P7簽章值

|  |  |  |
| --- | --- | --- |
| 說明 | | PKCS#7格式簽章 |
| 輸入 | **定義** | function SignPkcs7 (  strContent: string,  dwFlags: string,  callback: function/string  ) |
| **參數說明** | |
| 參數 | 值 |
| strContent | 簽章明文，元件會以UTF-16LE編碼後簽章 |
| dwFlags | 簽章選項，請帶0 |
| callback | 完成後回應之Function Name |
| 輸出 | **定義** | function callback (code, msg, token, data)  data = {  signature: string  sn: string,  cn: strgin,  ntb: string,  nta: string,  issdn: string,  subdn: string,  certfinger: string,  certb64: string  } |
| **回傳資料說明** | |
| 參數 | 值 |
| code | 0 成功 5005 參數錯誤  5067 執行簽章值，尚未選擇簽章憑證  5061 簽章失敗  5001 一般錯誤 |
| msg | 執行結果訊息 |
| token | 保留 |
| data | Json格式回傳，回傳範例：  {"sn":"602198D3","cn":"A123456789","ntb":"20160602104023","nta":"20160616235959","issdn":"CN=TaiCA Secure CA -Evaluation Only, OU=Certification Service Provider-Evaluation Only, O=TAIWAN-CA.COM Inc., C=TW","subdn":"","certfinger":"","certb64":"MIIE1zCCA…","signature":"MIIG3AYJKoZIhvc…"}  　Signature　  　　p7簽章值，以Base64編碼  　sn                 　　憑證序號  　cn                 　　憑證的主旨用戶名稱（common name）  　ntb                       憑證生效日yyyyMMddhhmmss（GMT+8）  　nta                        憑證效期yyyyMMddhhmmss（GMT+8）  　issdn            　　憑證簽發者，各項以逗點區隔  　subdn           　　憑證主旨，各項以逗點區隔  　certfinger     　　憑證摘要值（SHA1 Hex）  　certb64         　　X.509 Base64編碼之憑證 |

### 4.1.3 驗章API

VerifyPkcs7withAPName.jsp(驗章)

* 正式URL(交易類)：[http://](http://172.17.90.122:8086/ssCgi/VerifyPkcs7withAPName.jsp)172.20.91.5[:8086/ssCgi/VerifyPkcs7withAPName.jsp](http://172.17.90.122:8086/ssCgi/VerifyPkcs7withAPName.jsp)
* 正式URL(非交易類)：[http://](http://172.17.90.122:8086/ssCgi/VerifyPkcs7withAPName.jsp)172.20.91.46[:8086/ssCgi/VerifyPkcs7withAPName.jsp](http://172.17.90.122:8086/ssCgi/VerifyPkcs7withAPName.jsp)
* 測試URL：[http://](http://172.17.90.122:8086/ssCgi/VerifyPkcs7withAPName.jsp)172.20.189.48[:8086/ssCgi/VerifyPkcs7withAPName.jsp](http://172.17.90.122:8086/ssCgi/VerifyPkcs7withAPName.jsp)
* 傳送方式：POST

|  |  |  |
| --- | --- | --- |
| Request | | |
| 參數 | 值 | 備註 |
| txnCode | CONSULTING | 依定義值傳送 |
| Plaintext | 簽章明文 | 明文規則請參考4.1 |
| Signature | P7簽章值 | 簽章時產生 |
| SignatureType | 7 | 固定值 |
| 範例 | http://172.20.189.48:8086/ssCgi/VerifyPkcs7withAPName.jsp?txnCode=APName&Plaintext=Qww&Signature=MIIG6gYJKoZIhvcNAQcCoIIG2zCCBtcCAQExCzAJBgUrDgM...XssB6Q%3D%3D&SignatureType=7 | |

|  |  |
| --- | --- |
| Response | |
| 格式 | ErrorCode||ErrorMsg(TimeStamp)||NotBefore||NotAfter||DN||Serial||Plantext||Cert |
| 範例 | 成功：  0||20130508164122517(成功)||20130507174124000||20130604235959000|| C=TW,O=TaiCA Secure CA - Evaluation Only,O=Certificate Service Provider - Evaluation Only,OU=RA-KHSecurities,OU=MIS,CN=E249811562-00,E=@@|| 60056135||123|| -----BEGIN CERTIFICATE----- MIIE0TCCA7mgAwIBAgIEYAVhNTANBgkqhkiG9w0BAQUFADCBjjELMAkGA1UEBhMC…..  失敗：  8062||簽章值驗證錯誤!(失敗)||null||null||null||null||null||null |
| ErrorCode非0即為驗章失敗，Response若含跳脫字元則需自行處理 | |

## 5.1注意事項

### 5.1.1取得代登入Token(authToken)

|  |  |  |
| --- | --- | --- |
| Http Method | | POST |
| URL | | {APIdomain}/api/SSO/GetAuthToken |
| 測試 | https://ecip.testesunsec.com.tw | |
| 正式 | https://ecip.esunsec.com.tw | |
| **Objective** | | |
| **此API不可以開放讓外部直接串接！**  **串接此代登入功能，必須確保系統已透過SSO-003登入API或FIDO搭配雙因子驗證過顧客身分並允許授權登入，在授權有效期內才可以呼叫此API，取得代登入Token(JWT格式)。**  **另外authToken的有效期只有2分鐘，請需要使用authToken的時候再去觸發，避免顧客掛網超過2分鐘後逾期。** | | |

Request

|  |  |  |  |
| --- | --- | --- | --- |
| **參數說明** | | | |
| **Content-Type** | **x-www-form-urlencoded** | | |
| **參數名稱** | **欄位說明** | **必/選填** | **備註** |
| AID | 證券帳號 | required | 分公司(4碼)+帳號(7碼)，共11碼 |
| systemID | 系統代號 | required | 請參考ECIP規格文件 |
| clientIP | 使用者IP | required |  |

Response

|  |  |  |  |
| --- | --- | --- | --- |
| **參數說明(json)** | | | |
| **參數名稱** | **欄位說明** | **型別** | **備註** |
| returnCode | API狀態代碼 | string | 0000/-1000/-2002 |
| returnMsg | API回應訊息 | string | -1000 系統別不存在 -2000 傳入參數有誤 |
| authToken | 代登入Token | string | Token有效期間為2分鐘 |
| **Example** | | | |
| {  "returnCode": "0000",  "returnMsg": "",  "extraMsg": "",  "serverName": "SSOA000T01",  "serverTime": "20220303-104740.706",  "data": {  "rowCount": 1,  "row": [  {  "authToken":  eyJhbGciOiJIUzUxMiIsInR5cCI6IkpXVCJ9.eyJBSUQiOiI4ODQwMTIzNDU2NyIsIm5iZiI6MTY5MjE4MTczNCwiZXhwIjoxNjkyMTgxODU0LCJpYXQiOjE2OTIxODE3MzQsImlzcyI6IkVDSVAiLCJhdWQiOiJLIn0.FTCqnd2YKgnK5u9D4EYHBrNRA4XKwINAigfHlZM9Ohw0MsEDqzlQZFvF2L0Th99rgBkRkVmYv0ap-llOrEcrxA  }  ]  } | | | |

### 5.1.2取得identifyNo方式

●加密方式: SHA256

●組成內容:

|  |  |
| --- | --- |
| 名稱 | 範例 |
| businessNo(公司帳號) | 70781513 |
| apiVersion(API版本) | 固定為1.0 |
| hashKeyNo | HashKey組別 |
| verifyNo(驗證編號) | 需唯一，請使用yyyyMMddhhmmssffff |
| returnParams | 填入 authToken |
| memberNo | 身分證或統一編號 |
| raFunc |  |
| hashKey | 參考2.1 設定相關事項的第四點 |

●使用範例程式:

string stSHA256 = businessNo + apiVersion + hashKeyNo + verifyNo + returnParams + memberNo + raFunc + hashKey;

SHA256 sha256 = new SHA256CryptoServiceProvider();

var identifyNo = BitConverter.ToString(sha256.ComputeHash(Encoding.Unicode.GetBytes(stSHA256))).Replace("-", "").ToLower();

### 5.1.3 注意彈窗問題

因串接時會遇到彈窗被瀏覽器阻擋的問題，故須在前端加上下方範例程式，來預防此事件發生。

參考：<https://blog.darkthread.net/blog/pop-win-block-casestudy/>

範例：

export function openWinTWCA() {  
   const w = 1024;  
   const h = 860;  
   const dualScreenLeft = window.screenLeft != undefined ? window.screenLeft : window.screenX;  
   const dualScreenTop = window.screenTop != undefined ? window.screenTop : window.screenY;  
   const width = window.innerWidth;  
   const height = window.innerHeight;  
   const left = width / 2 - w / 2 + dualScreenLeft;  
   const top = height / 2 - h / 2 + dualScreenTop;  
   const htmlContent = '<html><body><h3>讀取中…</h3></body></html>';  
 const twcaNewWin = window.open('', 'twcaNewWin', `scrollbars=yes,width=${w},height=${h},left=${left},top=${top}`);  
   twcaNewWin.document.write(htmlContent);  
   return twcaNewWin;  
}

## 6.1 其他事項

### 6.1.1詞彙說明

|  |  |
| --- | --- |
| 名詞 | 說明 |
| WebCA | 瀏覽器憑證作業與管理平台，提供憑證申請、展期、變更密碼與刪除瀏覽器憑證等功能。 |
| 應用平台 | 與WebCA界接之應用平台 |
| HashKey組別 | WebCA提供多組交易金鑰，可藉以檢核來源是否合法及資料是否為被竄改 |
| HashKey | 對應HashKey組別的交易金鑰 |
| Proxy Window | 呼叫WebCA需要使用的Proxy網頁視窗webcaFrm.html，必須與應用平台在同一個domain中，且可以由外部連線。 |

### 6.1.2資料格式說明

|  |  |  |
| --- | --- | --- |
| 資料格式 | 說明 | 範例 |
| 9(5) | 數字，長度最長為5位 | 12345  123 |
| 9(5,2) | 數字含小數點，小數點左邊最多5位整數，小數點右邊最多2位小數 | 12345.67  123.45  12.3  12 |
| A(5) | 英數字，長度最長為5位 | ABC  ABCDE  ABC1  12AB |
| X(10) | 文數字、符號、文字(如中文)字串，長度最長10碼 | 鞋子兩雙  USD  台北市1234567 |
| yyyyMMdd  HHmmssfff | 日期時間格式  yyyy 西元年 MM 月  dd 日 HH 時  mm 分 ss 秒  fff 毫秒  西元年月日，24小時制，月日時分秒不滿兩碼左補0 | 20130605082235999 |
| JSON | JavaScript Object Notation | {  "firstName": "John",  "lastName": "Smith",  "age": 25  } |

## 7.1 跨來源資源共用(CORS)設定參考

憑證佈署機制由於透過瀏覽器進行跨網域資源共用，故對應整合站台需設定CORS，本文件提供IIS與Apache之設定做為參考，實際設定方式以整合站台為主。

### 7.1.1 IIS站台設定參考

|  |
| --- |
| Access-Control-Allow-Headers : charset  Access-Control-Allow-Origin : <https://caweb.testesunsec.com.tw>(測試)；<https://caweb.esunsec.com.tw>(正式)  Access-Control-Allow-Methods : GET, POST  Access-Control-Request-Headers : \* |

### 7.1.2 Apache站台設定參考

|  |
| --- |
| LoadModule headers\_module modules/mod\_headers.so  LoadModule rewrite\_module modules/mod\_rewrite.so  <IfModule mod\_headers.c>  Header always set Access-Control-Allow-Origin “<https://caweb.testesunsec.com.tw>”(測試)；”<https://caweb.esunsec.com.tw>”(正式)  Header always set Access-Control-Allow-Headers "Authorization"  Header always set Access-Control-Allow-Methods "GET"  Header always set Access-Control-Expose-Headers "Location"  Header always set Access-Control-Max-Age "600"  TraceEnable Off  RewriteEngine On  RewriteCond %{REQUEST\_METHOD} OPTIONS  RewriteRule ^(.\*)$ $1 [R=200,L] |

※參考來源:https://stackoverflow.com/questions/42558221/how-to-cors-enable-apache-web-server-including-preflight-and-custom-headers

附件I、裝置別Device 與 瀏覽器Browser判別的JavaScript

|  |
| --- |
| functio*n funGetDevice() {*  *const ua = navigator.userAgent.toLowerCase();*  *let sDevice = '';*      if (ua.indexOf('windows phone') > -1) {          sDevice = 'wp';      } else if (ua.indexOf('android') > -1) {          if (ua.indexOf('mobile') > -1) {              sDevice = 'gp';          } else {              sDevice = 'aa';          }      } else if (ua.indexOf('iphone') > -1) {          sDevice = 'ip';      } else if (ua.indexOf('ipad') > -1) {          sDevice = 'ia';      } else if (ua.indexOf('mac') > -1) {          sDevice = 'mc';      } else if (ua.indexOf('windows') > -1) {          sDevice = 'pc';      } else {          sDevice = 'ot';      }      return sDevice;  }  function funGetBrowser() {      const ua = navigator.userAgent.toLowerCase();      let sBrowser = '';      if (ua.indexOf('edg') > -1) {          sBrowser = 'edge';      } else if (ua.indexOf('firefox') > -1) {          sBrowser = 'firefox';      } else if (ua.indexOf('msie') > -1) {          sBrowser = 'ie';      } else if (ua.indexOf('rv:11.0') > -1 && ua.indexOf('like gecko') > -1) {          if (ua.indexOf('windows nt 6.3;') > -1) {              sBrowser = 'ie';          } else {              if (ua.indexOf('windows nt ') > -1) {                  sBrowser = 'ie';              } else {                  sBrowser = 'other';              }          }      } else if (ua.indexOf('iemobile') > -1) {          sBrowser = 'ie';      } else if (ua.indexOf('chrome') > -1) {          sBrowser = 'chrome';      } else if (ua.indexOf('safari') > -1) {          if (ua.indexOf('gsa') > -1 || ua.indexOf('crios') > -1) {              sBrowser = 'chrome';          } else {              sBrowser = 'safari';          }      } else if (ua.indexOf('dophin') > -1) {          sBrowser = 'dophin';      } else if (ua.indexOf('opera') > -1) {          sBrowser = 'opera';      } else {          sBrowser = 'other';      }      return sBrowser;  } |