GTIDEM JS Library

Contents

Revision History

A revision occurs with each release of the product, or as needed. A revised version can contain minor or major changes. Table 1 lists the versions of this manual.

Table 1. Revision History

Revision	Description	Date
1.0	First release。Define functions and status code.	**08/20/2021**
1.1	1. 增加 GetTokenInfo 回傳參數。 2. 增加 Pin 鎖定的錯誤碼定義。 3. 增加GTIDEM_ReadCertByIndexWithoutPIN和 GTIDEM_ReadCertByLabelWithoutPIN的參數。 4. 所有函式增加序號回傳。 5. 增加錯誤碼定義。	09/08/2021
1.2	1. 增加指定使用者名稱函式。 2. 增加錯誤碼定義。	09/16/2021
1.3	1. 增加 GTIDEM_InitToken 和 GTIDEM_UnlockPIN 2. 增加製作初始化資料的說明	09/22/2021

1. 概述

描述 GTIDEM JS 的行為，定義名詞和解釋狀態碼的意義

2.名詞解釋

GTIDEM JS	Javascript library, 在瀏覽器上使用 webauthn API 與載具溝通。
Label	儲存在載具中，可作為搜尋特定憑證或是金鑰的條件。
Key ID	由使用者指定，在建立金鑰對或匯入憑證使用。之後當作Label 使用。
Key Handle	使用者未指定 Key ID 時，由載具產生此值。之後可當作Label 使用。
CSR	Certificate Signing Request

2. 函式

2.1函式概述

名稱	描述
GTIDEM_SignDataByIndex	使用特定位址金鑰對資料簽名
GTIDEM_SignDataByLabel	使用特定標籤金鑰對資料簽名
GTIDEM_ReadCertByIndexWithoutPIN	不需要驗證使用者密碼就可讀取特定位址憑證。
GTIDEM_ReadCertByLabelWithoutPIN	不需要驗證使用者密碼就可讀取特定標籤憑證。
GTIDEM_GenRSA2048	產生RSA 2048 金鑰對並返回公鑰
GTIDEM_GenRSA2048CSR	產生RSA 2048 金鑰對並返回CSR 格式
GTIDEM_ImportCertificate	匯入憑證
GTIDEM_DeleteCertByLabel	刪除指定標籤的憑證和金鑰對
GTIDEM_ClearToken	清除卡片中的所有憑證和金鑰對
GTIDEM_ChangeUserPIN	修改使用者密碼
GTIDEM_GetTokenInfo	返回載具資訊
GTIDEM_SetName	指定使用者名稱

2.1. 函式說明

2.1.1. GTIDEM_SignDataByIndex

GTIDEM_SignDataByIndex(index, bSerialNumber ,alg_number, bPlain)

使用特定位置的金鑰對傳入的資料做簽名。

參數		
Number	index	指定位址的金鑰對
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列
Number	alg_number	簽名演算法, ALG_RSA2048SHA256 或者 ALG_RSA2048SHA256_PreHash
Uin8Array	bPlain	被簽名的資料

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.signature	簽名
Uin8Array	GTIdemJs.sn	載具序號

2.1.2. GTIDEM_SignDataByLabel

GTIDEM_SignDataByLabel(bLabel, bSerialNumber ,alg_number, bPlain)

使用指定標籤的金鑰對傳入的資料做簽名。

參數		
Uin8Array	bLabel	指定標籤的金鑰對

參數		
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列
Number	alg_number	簽名演算法, ALG_RSA2048SHA256 或者 ALG_RSA2048SHA256_PreHash
Uin8Array	bPlain	被簽名的資料

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.signature	簽名
Uin8Array	GTIdemJs.sn	載具序號

2.1.3. GTIDEM_ReadCertByIndexWithoutPIN

```
GTIDEM_ReadCertByIndexWithoutPIN(index, bSerialNumber)
```

不需要使用者密碼，就讀取特定位址的憑證。

參數		
Number	index	指定位址的憑證
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.certificate	憑證內容
Number	GTIdemJs.credentialNum	憑證數量
Uin8Array	GTIdemJs.sn	載具序號

2.1.4. GTIDEM_ReadCertByLabelWithoutPIN

GTIDEM_ReadCertByLabelWithoutPIN(bLabel, bSerialNumber)

不需要使用者密碼，就讀取特定標籤的憑證。

參數		
Uin8Array	bLabel	指定標籤的憑證
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.certificate	憑證內容
Number	GTIdemJs.credentialNum	憑證數量
Uin8Array	GTIdemJs.sn	載具序號

2.1.5. GTIDEM_GenRSA2048

GTIDEM_GenRSA2048(bSerialNumber, bKeyID)

產生 RSA 2048 金鑰對，並回傳公鑰的 RAW

參數		
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列
Uin8Array	bKeyID	用來綁定金鑰對，可填入 undefined 或是空陣列則載具會產生預設的 KeyHandle回傳。

回傳		
Number	GTIdemJs.statusCode	狀態碼

回傳		
ArrayBuffer	GTIdemJs.keyhandle	若 bKeyID 有效，則此欄位為 undefined
ArrayBuffer	GTIdemJs.rsakeypair	公鑰的 RAW
Uin8Array	GTIdemJs.sn	載具序號

2.1.6. GTIDEM_GenRSA2048CSR

GTIDEM_GenRSA2048CSR(bSerialNumber, bKeyID)

產生 RSA 2048 金鑰對並回傳 CSR 資料

參數		
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列
Uin8Array	bKeyID	用來綁定金鑰對，可填入 undefined 或是空陣列則載具會產生預設的 KeyHandle回傳。

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.keyhandle	若 bKeyID 有效，則此欄位為 undefined
ArrayBuffer	GTIdemJs.csr	CSR
Uin8Array	GTIdemJs.sn	載具序號

2.1.7. GTIDEM_ImportCertificate

GTIDEM_ImportCertificate(bSerialNumber, bKeyHandle, bKeyID, HexCert, bPlain)

指定 KeyHandle 匯入憑證。若在 GTIDEM_GenRSA2048 或是 GTIDEM_GenRSA2048CSR 使用 KeyID綁定金鑰，則此處的 KeyHandle 要使用已指定的 KeyID。

參數		
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列
Uin8Array	bKeyHandle	指定欄位
Uin8Array	bKeyID	用來綁定金鑰對，可填入 undefined 或是空陣列則載具會產生預設的 KeyHandle回傳。
Uin8Array	HexCert	欲匯入的憑證內容
Uin8Array	bPlain	使用指定的金鑰簽名並用 ALG_RSA2048SHA256_PreHash 演算法對填入的資料簽名，所以資料長度必須為32 bytes，可做為確認憑證和金鑰對的匹配。若不需此功能，則可填入 undefined 或是空陣列。

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.signature	簽名
Uin8Array	GTIdemJs.sn	載具序號

2.1.8. GTIDEM_DeleteCertByLabel

```
GTIDEM_DeleteCertByLabel(bLabel, bSerialNumber)
```

刪除特定標籤的金鑰和憑證。

參數		
Uin8Array	bLabel	指定標籤
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列

回傳		
Number	GTIdemJs.statusCode	狀態碼

回傳		
Uin8Array	GTIdemJs.sn	載具序號

2.1.9. GTIDEM_ClearToken

```
GTIDEM_ClearToken( bSerialNumber)
```

清除載具中的所有憑證和金鑰。

參數		
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列

回傳		
Number	GTIdemJs.statusCode	狀態碼
Uin8Array	GTIdemJs.sn	載具序號

2.1.10. GTIDEM_ChangeUserPIN

```
GTIDEM_ChangeUserPIN(bOldPIN, bNewPIN, bSerialNumber)
```

修改使用者密碼。

參數		
Uin8Array	bOldPIN	舊密碼
Uin8Array	bNewPIN	新密碼
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列
Number	GTIdemJs.pinRetry	剩餘密碼次數

回傳		
Number	GTIdemJs.statusCode	狀態碼
Uin8Array	GTIdemJs.sn	載具序號

2.1.11. GTIDEM_GetTokenInfo

```
GTIDEM_GetTokenInfo(bSerialNumber)
```

回傳載具資訊。

參數		
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.fw	韌體版本
ArrayBuffer	GTIdemJs.sw	軟體版本
Number	GTIdemJs.pinRetry	剩餘密碼次數
Number	GTIdemJs.credentialNum	憑證數量
ArrayBuffer	GTIdemJs.sn	載具序號
ArrayBuffer	GTIdemJs.rn	隨機亂數
ArrayBuffer	GTIdemJs.ecpoint	橢圓曲線公鑰

2.1.12. GTIDEM_SetName

```
GTIDEM_SetName(sName)
```

指定使用者名稱，目前已知在 windows 上的少部分API 才有作用。

參數		
text	sName	指定使用者名稱，若無指定則會顯示預設名稱。

2.1.13. GTIDEM_InitToken

```
GTIDEM_InitToken(bSerialNumber, bEncrypted_InitData, bHmacValue_InitData)
```

重新設定 SO PIN, User PIN 和各項參數。

參數		
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列
Uin8Array	bEncrypted_InitData	被加密的初始化資料
Uin8Array	bHmacValue_InitData	對明文的初始化資料產生 HMAC

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.sn	載具序號

2.1.14. GTIDEM_UnlockPIN

```
GTIDEM_UnlockPIN(bSerialNumber, bEncrypted_InitData, bHmacValue_InitData)
```

重新設定新的 User PIN 和最大密碼嘗試次數。

參數		
Uin8Array	bSerialNumber	指定序號序號。若不指定載具序號，則可填入 undefined 或是空陣列
Uin8Array	bEncrypted_InitData	被加密的初始化資料

參數		
Uin8Array	bHmacValue_InitData	對明文的初始化資料產生 HMAC

回傳		
Number	GTIdemJs.statusCode	狀態碼
ArrayBuffer	GTIdemJs.sn	載具序號

3. 狀態碼

狀態碼	名稱	描述
0x00	CTAP1_ERR_SUCCESS	成功
0x14	CTAP2_ERR_MISSING_PARAMETER	缺少必要參數
0x19	CTAP2_ERR_CREDENTIAL_EXCLUDED	憑證或是金鑰已經存在
0x20	CTAP2_ERR_CERTIFICATE_NOT_VALID	憑證或是金鑰不能使用
0x26	CTAP2_ERR_UNSUPPORTED_ALGORITM	不支援的演算法
0x28	CTAP2_ERR_KEY_STORE_FULL	憑證數量已滿
0x2E	CTAP2_ERR_NO_CERTIFICATE	要求的憑證或是金鑰不存在
0x30	CTAP2_ERR_NOT_ALLOWED	不允許此操作
0x31	CTAP2_ERR_PIN_INVALID	驗證使用者密碼錯誤
0x32	CTAP2_ERR_PIN_BLOCKED	密碼鎖定
0x35	CTAP2_ERR_PIN_NOT_SET	使用者密碼未設定或是需要修改使用者密碼
0x36	CTAP2_ERR_PIN_REQUIRED	需要驗證使用者密碼
0x38	CTAP2_ERR_TOKEN_EXPIRED	載具到期
0xF2	CTAP2_VENDOR_ERROR_TOKEN	載具序號錯誤
0xF3	CTAP2_VENDOR_ERROR_LENGTH	載具序號長度錯誤
0xF4	CTAP2_ERR_VENDOR_ERROR_NO_USER	沒有使用者觸碰
0xF5	CTAP2_ERR_VENDOR_ERROR_CREDENTIAL_EXIST	憑證已匯入
0xF6	CTAP2_ERR_VENDOR_ERROR_INVALID_DATA	此資料無效

狀態碼	名稱	描述
0xF7	CTAP2_ERR_VENDOR_ERROR_NOT_ALLOWED_RPID	不允許在此網站使用
0xF8	CTAP2_ERR_VENDOR_ERROR_PIN_RXPIRED	密碼到期，需要變更密碼
0xE001	WEB_ERR_UserCancelorTimeout	使用者取消操作或是操作逾時
0xE002	WEB_ERR_OperationAbort	操作被拒絕
0xE003	WEB_ERR_Timeout	操作逾時
0xE004	WEB_ERR_Unknow	發生不預期的錯誤
0xE005	WEB_ERR_InvalidState	此指令被拒絕，已有指令正在執行中。

4. 其他項目

4.1. Key Handle 的產生

當呼叫 GTIDEM_GenRSA2048 或是 GTIDEM_GenRSA2048CSR 而沒有指定 Key ID時，載具會使用金鑰的 modulus 進行 SHA1 運算，取得其雜湊值當作 Key Handle 並回傳。

4.2. 初始化資料

4.2.1. 編碼

各項參數使用 CBOR MAP 編碼，下方表格為各項初始化參數

參數標籤	資料型態	描述
soPIN	Byte Array	新的 SO PIN
userPIN	Byte Array	新的 User PIN
allowedRPID	Byte Array	被允許的 Rpid 雜湊列表，Rpid 通常是網頁的 domain.

範例一：載具初始化 **SO PIN = 12345678** **User PIN = 12345678**	

Javascript code	{
	soPIN: [0x31,0x32,0x33,0x34,0x35,0x36,0x37,0x38] ,
	userPIN: [0x31,0x32,0x33,0x34,0x35,0x36,0x37,0x38] ,
	allowedRPID:[….]
	}

CBOR 編碼	{"soPIN": h'3132333435363738', "userPIN": h'3132333435363738', "allowedRPID": h'BEDA17EF4D6D57F203E55BFBC0E7E3739DB0E2FF300D4D09BD82491E27CB2EBC616146FA64F732679BE23B280315AE8E'} A3 # map(3) 65 # text(5) 736F50494E # "soPIN" 48 # bytes(8) 3132333435363738 # "12345678" 67 # text(7) 7573657250494E # "userPIN" 48 # bytes(8) 3132333435363738 # "12345678" 6B # text(11) 616C6C6F77656452504944 # "allowedRPID" 58 30 # bytes(48) BEDA17EF4D6D57F203E55BFBC0E7E3739DB0E2FF300D4D09BD82491E27CB2EBC616146FA64F732679BE23B280315AE8E # "\xBE\xDA\x17\xEFMmW\xF2\x03\xE5[\xFB\xC0\xE7\xE3s\x9D\xB0\xE2\xFF0\rM\t\xBD\x82I\x1E'\xCB.\xBCaaF\xFAd\xF72g\x9B\xE2;(\x03\x15\xAE\x8E"

範例二：解鎖PIN **User PIN = 12345678**	

Javascript code	{
	userPIN: [0x31,0x32,0x33,0x34,0x35,0x36,0x37,0x38]
	}

範例二：解鎖PIN **User PIN = 12345678**	
CBOR 編碼	{"userPIN": h’3132333435363738'} A1 # map(1) 67 # text(7) 7573657250494E # "userPIN" 48 # bytes(8) 3132333435363738 # “12345678" }

4.2.2. 加密初始化資料和產生 HMAC

將新的管理者密碼 NewSOPIN 和 新的使用者密碼 NewUserPIN 以及初始化參數用CBOR 編碼為初始化資料 InitData。

按照下方步驟加密初始化資料和其 HMAC，

- 呼叫 GTIDEM_GetTokenInfo 取得隨機碼 RN(32 bytes) 和載具序號 SN (9 bytes)
- 計算 IV = SHA256(SN) 取左邊 16 bytes
- 產生會議金鑰 SessionKey = SHA256(SO_PIN||RN||0x01)
- 產生HMAC 金鑰 MacKey = SHA256(SO_PIN||RN||0x02)
- 加密處初始化資料 ENC_InitData = AES-CBC(SessionKey, InitData) with IV with PKCS#7padding
- 產生HMAC 訊息確認碼 HMAC_InitData = HMAC-SHA256(MacKey,InitData)

4.2.3. 允許的 RPID 列表製作

RPID 應該為 domain name 不包含 port，製作列表前應該過濾沒有特別指定字元的 domain(wildcard domain) 和帶有port 的字串。

將每個合格的 RPID移除特定字元後，做 SHA256 計算後取前 8 bytes。

以下是範例：

欲加入白名單的RPIDs

- gotrustid.com
- gotrustid.com.tw
- *.ctbc.com
- ctbc123.com
- gotrustidem-dev.github.io

過濾後的RPIDs

- gotrustid.com
- gotrustid.com.tw
- ctbc.com
- ctbc123.com
- gotrustidem-dev.github.io

轉換為 SHA256 取前 8 bytes

- beda17ef4d6d57f2
- 03e55bfbc0e7e373
- bd82491e27cb2ebc
- 616146fa64f73267
- 9db0e2ff300d4d09

將轉換後的白名單PRIDs 排列為列表：

allowedRPID:[beda17ef4d6d57f203e55bfbc0e7e373bd82491e27cb2ebc616146fa64f73267
9db0e2ff300d4d09]