

CorpID Sandbox Programme Developer Guide

Version: 1.0

December 2025

© The Government of the Hong Kong Special Administrative Region
of the People's Republic of China

The contents of this document remain the property of and may not be reproduced in whole or in part without the express permission of the Government of the Hong Kong Special Administrative Region of the People's Republic of China

Amendment History			
Version No.	Release Date	Section Affected	Summary of Changes

CONTENTS

1. OVERVIEW	4
2. CORE CONCEPTS	6
2.1. INTRODUCTION OF API FUNCTIONS.....	6
2.2. API DATA ENCRYPTION AND DECRYPTION	7
2.3. CORPID USER PROFILES.....	11
2.4. AUTHORISATION SCOPES.....	13
3. PREREQUISITES	14
3.1. REQUIREMENTS OF E-SERVICE.....	14
3.2. SETTING IN CORPID SANDBOX DEVELOPER PORTAL.....	14
3.3. SETUP OF “IAM SMART” E-SERVICE ACCOUNT.....	16
3.4. “IAM SMART” TESTING MOBILE APP.....	17
4. IMPLEMENTATION PROCEDURE	18
4.1. CORPORATE IDENTITY VERIFICATION WORKFLOW.....	19
4.2. DIGITAL SIGNING (PDF) WORKFLOW.....	21
4.3. FORM PRE-FILLING WORKFLOW	23
4.4. FORM PRE-FILLING WITHOUT SERVICE LOGIN WORKFLOW	25
APPENDIX	27
A. IAM SMART SIMULATED API FOR CORPID SANDBOX	27

1. OVERVIEW

To accelerate the digital economy development in Hong Kong through strengthening digital infrastructure and fostering digital transformation, the Digital Policy Office (DPO) is developing the Digital Corporate Identity (CorpID) Platform, thereby facilitating and bringing ease to corporations in their use of e-Government services. The DPO will launch the CorpID Platform in end 2026 and gradually extending the range of services.

CorpID can serve as a common digital key for identity authentication, enabling corporations to use the e-government services and conduct online business transactions at ease. Successful applicant will be issued a CorpID. The responsible person of the corporation can set the authorised representatives and their authorities to conduct electronic commercial activities on behalf of the corporation.

The DPO, collaborates with the Cyberport, has launched the CorpID Sandbox Programme in December 2025 for corporations and government bureaux/departments interested in adopting CorpID to conduct Proof-of-Concept (PoC) testing and develop their applications in order to design application scenarios and solutions that can better meet the market demands. Typical use cases include remote account opening where corporate identities and authorisation of authorised representatives can be verified, login to various online service platforms in a simple, secure and convenient manner, verifiable digital signing of application documents, funding agreements or contracts, and automatic form pre-filling without having to repeatedly provide corporate information, thereby reducing business processing time and human error.

The CorpID Sandbox Programme provides a thematic website and a CorpID Sandbox Developer Portal in which technical references such as this Developer Guide, CorpID Sandbox Application Programming Interface (API) Specifications, CorpID sample applications, information on pre-requisites and relevant configurations are available for developers to understand the technical requirements, and preview the CorpID Simulated APIs. These CorpID Simulated APIs can be tested using mock-up data and predefined responses in the CorpID Sandbox Developer Portal without coding, facilitating smooth and efficient adoption of CorpID and deployment of real applications.

Besides, a dedicated and isolated testing environment (Sandbox Platform) is also developed, offering a suite of API functionalities and simulated workflows that mirror the actual operations for core CorpID functions (namely Corporate Identity Verification, Digital Signing, Form Pre-filling and Document Wallet). While the CorpID Sandbox Developer Portal contains information of all CorpID Simulated APIs, the Sandbox Platform together with a CorpID mock-up mini-program integrated with “iAM Smart” ITE environment and “iAM Smart” testing mobile app, provides executable CorpID Simulated APIs in selected scenarios for illustration purpose. These scenarios allow a desktop web application to access four selected CorpID functionalities, including corporate identity verification (i.e. integrated authentication with “iAM Smart”), form pre-filling after integrated authentication, digital signing after integrated authentication, as well as anonymous form pre-filling. Developers may develop its PoC application to understand the true benefits of CorpID, and ascertain the technical requirements and resource implications in adopting CorpID. In addition, the Sandbox Platform also provides testing on simulated API endpoints (supporting partial flows) other than these scenarios for verification by developers.

Please note that APIs in the CorpID Sandbox Developer Portal and the Sandbox Platform are tuned for illustration purpose. These APIs are being enhanced continuously following the actual implementation of the production CorpID Platform. These portal and platforms including the simulated APIs and sample applications are intended solely for development and validation purposes and should not be regarded as completely indicative of live production behaviour.

2. CORE CONCEPTS

2.1. INTRODUCTION OF API FUNCTIONS

The core functions as mentioned in Section 1 are built in form of RESTful APIs, which could be accessed by registered e-services. CorpID makes reference to OAuth 2.0 for authentication and authorisation amongst CorpID users, e-services and the CorpID Sandbox Platform. The e-services adopting CorpID are required to provide RESTful callback APIs to receive API responses from the CorpID Sandbox Platform in order to complete the entire workflow. The core CorpID functions include:

- Corporate Identity Verification

Corporate Identity Verification makes use of the Authentication API provided by the CorpID to verify the identities of the users in a simple and secure way. The API can be used in various scenarios, such as user login, membership system, booking system, etc.

- Digital Signing

e-service can make use of the Digital Signing API to enable a corporate user to perform digital signing with legal backing after the user has authenticated with e-service using CorpID. It can be used in many cases, such as digital signing an online application form and digital signing a contract and agreement.

- Form Pre-Filling

e-service can make use of the Profiles API to retrieve data of “corpProfileFields” and “eCorpFields”. The API can be used in various scenarios, such as account opening, account linkup, form pre-filling, etc.

- Document Wallet

Through the Document Wallet API, e-services can securely access and verify digital certificates and documents submitted by the corporations and issued by government B/Ds as supporting documents. These digital certificates are issued using the globally recognized Decentralized Identity (DID) and Verifiable Credential (VC) standards developed by the World Wide Web Consortium to ensure authenticity.

2.2. API DATA ENCRYPTION AND DECRYPTION

HTTPS will be used to secure communication channels between an e-service and the CorpID Sandbox Platform, establishing a robust foundation for data protection. To further protect the data in transit, an additional application layer of data encryption will be applied to all API POST requests (except the one that e-service requests for getting a symmetric encryption key). Callbacks from the CorpID Sandbox Platform will also be encrypted using the same method described herein, ensuring data transferred by HTTPS remains secure and protected from eavesdropping or manipulation by malicious service providers in exceptional situations.

The process of API data encryption is summarised as follows:

- (a) e-service requests a content encryption key from the CorpID Sandbox Platform.
- (b) The CorpID Sandbox Platform generates a symmetric Content Encryption Key (CEK).
- (c) The CorpID Sandbox Platform securely stores the generated key, encrypts and returns it to the e-service who initiated the request with the e-service's public key (KEK).
- (d) For subsequent interactions, all business data exchanged between the e-service and the CorpID Sandbox Platform will be encrypted using this CEK.
- (e) Both the CorpID Sandbox Platform and e-service will use the same CEK to encrypt and decrypt the API data.
- (f) When the CEK expires, the e-service must renew the CEK by repeating the above process.

2.2.1. Encryption and Decryption Workflow

The diagram below illustrates the detailed workflow for how an e-service can request a CEK, perform encryption and decryption using the CEK, and obtain a new CEK if the current one expires.

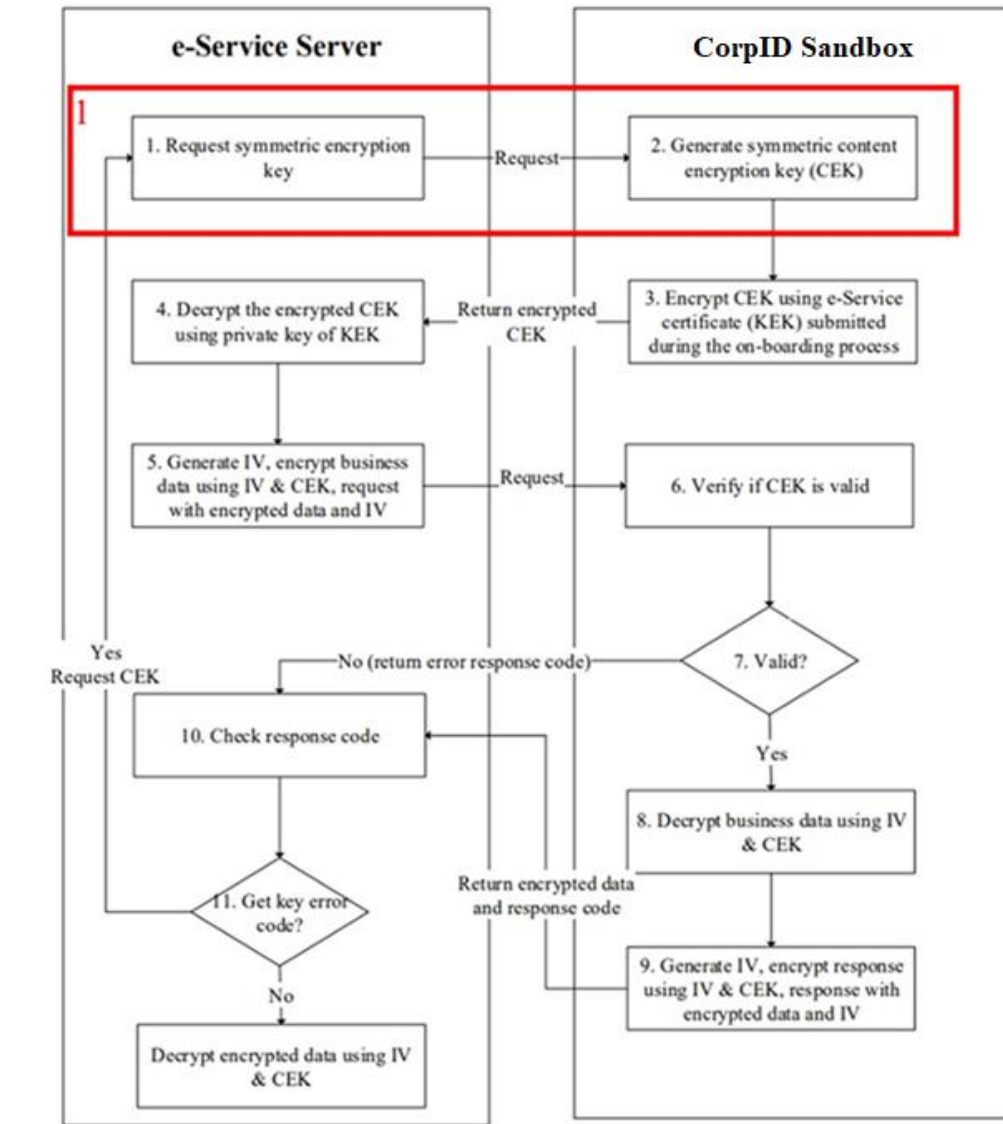


Figure – 1 Application of Content Encryption Key and Procedures of Encryption and Decryption (Perform by e-service)”

2.2.2. High Level Workflow Description of Callback Encryption and Decryption

Before the CorpID Sandbox initiates a callback, it will check if the CEK of the corresponding e-service is still valid. If the CEK is expired or does not exist, the CorpID Sandbox Platform will re-generate a new content encryption key for that e-service.

The CorpID Sandbox Platform uses the CEK to encrypt the callback body. Then such CEK will be encrypted by KEK of the e-service. The encrypted callback data and the encrypted CEK will be returned to the e-service through the callback.

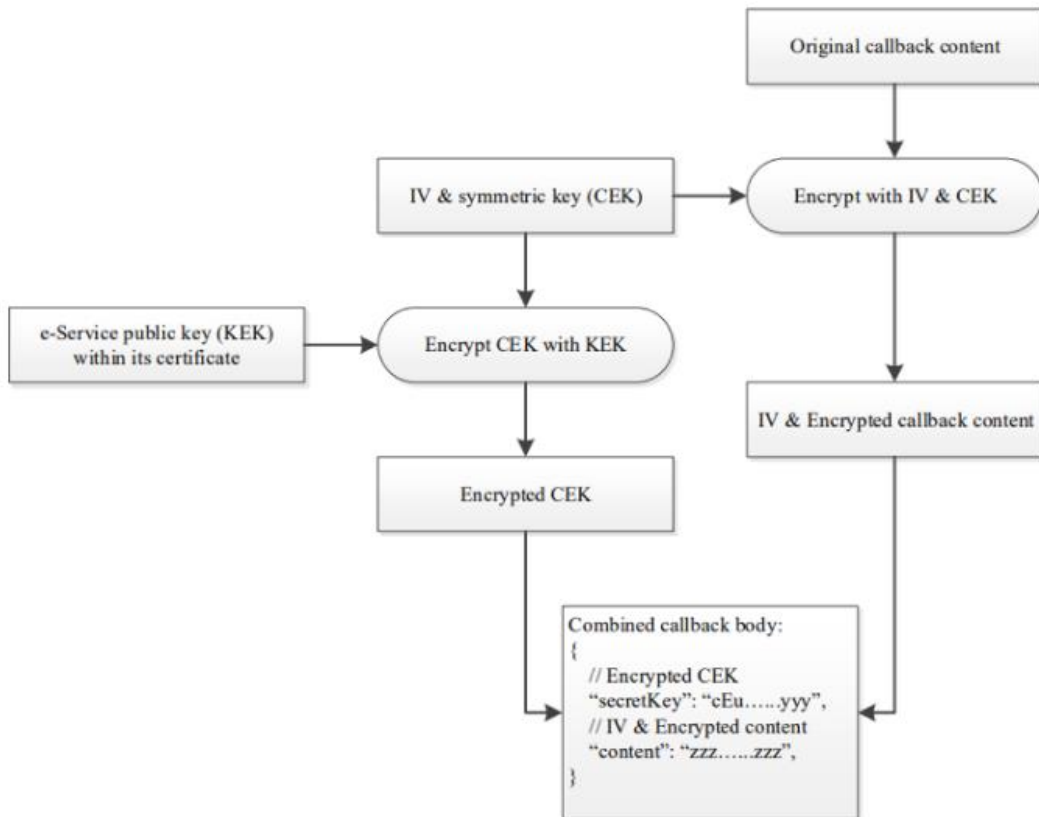


Figure – 2 Callback Encryption Process (Perform by CorpID / iAM Smart System)

After an e-service receives the callback, it should use the private key of the KEK to decrypt the encrypted CEK. Then the decrypted CEK can be used to decrypt the callback data.

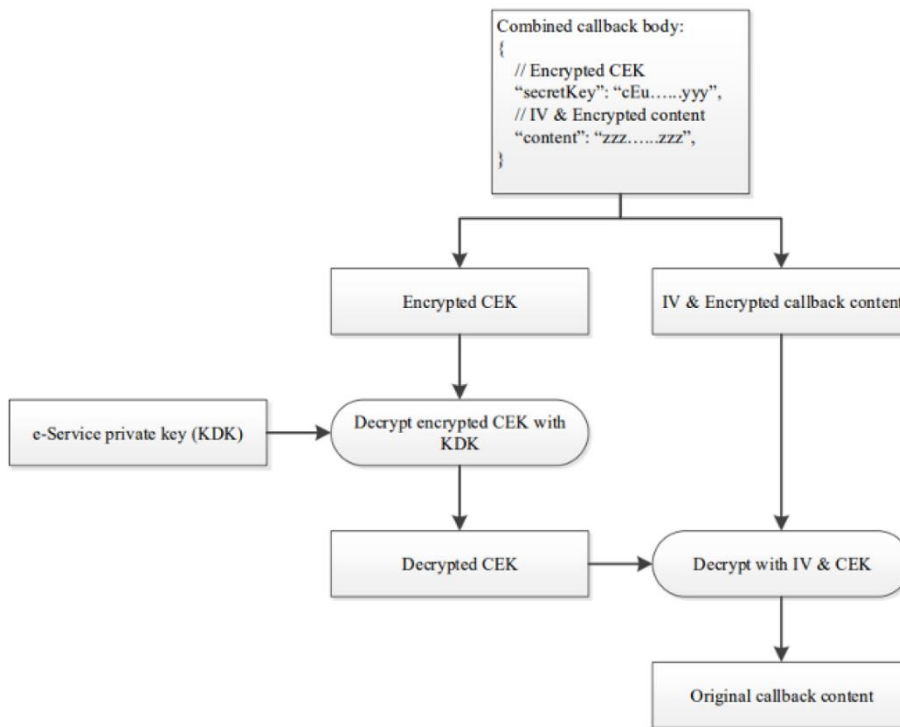


Figure – 3 Callback Decryption Process (Perform by the e-service)

For details, please refer to the CorpID Sandbox API Specifications.

2.3. CORPID USER PROFILES

A CorpID User Profiles represents both the individual user of CorpID and the company he/she represents. It contains the following user and corporate information.

Components	Information
“corpUserProfileFields”	The “corpUserProfileFields” consist of the CorpID user account information.
“corpProfileFields”	The “corpProfileFields” consist of the account information including the Unique Business Identifier (UBI) (previously known as Business Registration Number (BRN)), English Company Name, Chinese Company Name (must have 1 available) and Company Address. “corpProfileFields” are solely used for the purpose of corporate identity verification, such as remote account opening, login, etc.
“eCorpFields”	In addition to the account information, CorpID user can voluntarily maintain a “e-Corp profile” in which additional information of the corporate can be stored and retrieved for the purpose of online form pre-filling.

The details of each components of the CorpID User Profiles are as below:

“corpUserProfileFields”

Data Field	Description
id_cty_issue	ISO 3166 3-letter country code
id_type	1. Identity Card 2. Passport 3. Other Identity Type

“corpProfileFields”

Data Field	Description
corpID	The unique ID from the CorpID Sandbox Platform
Brn	Business Registration Number (or called Unique Business Identifier)
corpNameEN	Corporation Name (EN)
corpNameTC	Corporation Name (TC)
corpNameSC	Corporation Name (SC)
corpAddr	Corporation Business Address

“eCorpFields”

Data Field	Description
corpID	The Unique ID from CorpID Sandbox Platform
brn	Business Registration Number (or called Unique Business Identifier)
corpNameEN	Corporation Name (EN)
corpNameTC	Corporation Name (TC)
corpNameSC	Corporation Name (SC)
corpAddr	Corporation Business Address
corpTypeEN	Corporation Type (EN)
corpTypeTC	Corporation Type (TC)
corpTypeSC	Corporation Type (SC)
corpTel	Corporation Telephone No.
corpStatusEN	Corporation Active Status (EN)
corpStatusTC	Corporation Active Status (TC)
corpStatusSC	Corporation Active Status (SC)
placeOfIncorp	Place of Incorporation
dateOfReg	Date of Registration

“profileFields” and “eMEFields” of the associated “iAM Smart” user (if any) will also be returned through the integrated profile APIs. For details, please refer to “iAM Smart” API Specifications document separately.

2.4. AUTHORISATION SCOPES

The CorpID Sandbox Platform utilises the OAuth 2.0 authentication framework to enable an e-service to gain access to the CorpID Simulated APIs, with CorpID user's authorisation facilitated through the "iAM Smart" testing mobile app and CorpID mock-up mini-program.

Determine Authorisation Scope

To access the CorpID Simulated APIs, the e-service should first identify the appropriate authorisation scope(s) required for the specific CorpID Simulated APIs to be invoked for its business needs. The table below provides the mapping of authorisation scopes:

Authorisation Scope	Value
Authentication	corpidapi_auth
Switch corporation under same user	corpidapi_corplist
Form Pre-filling with Service Login (aka Profiles)	corpidapi_profiles
Form Pre-filling without Service Login (Anonymous Form Pre-filling)	corpidapi_formfill
Digital Signing with Service Login	corpidapi_sign
Digital Signing without Service Login (Anonymous Digital Signing)	
Re-authentication with Service Login	corpidapi_fr
Bulk Digital Signing with Service Login	corpidapi_bulksign
Bulk Digital Signing without Service Login (Anonymous Bulk Digital Signing)	
Document Wallet	corpidapi_docwallet

e-service should combine all authorisation scope(s) required into a single request to get the authorisation from the CorpID user through the CorpID Sandbox Platform when the CorpID user requests login to an e-service using "iAM Smart" testing mobile app and CorpID mock-up mini-program. For example, authorisation scope "corpidapi_auth", "corpidapi_formfill" and "corpidapi_sign" are used for CorpID authentication, form pre-filling and digital signing respectively.

3. PREREQUISITES

e-service should review and fulfill the prerequisites outlined in this section before testing the integration with the CorpID Sandbox Platform.

3.1. REQUIREMENTS OF E-SERVICE

e-service developers are required to prepare an Internet-facing server endpoint with HTTPS enabled to handle callbacks from the CorpID Sandbox Platform.

3.2. SETTING IN CORPID SANDBOX DEVELOPER PORTAL

The e-service should complete the necessary setup in the CorpID Sandbox Developer Portal.

3.2.1. Check the Client ID and Client's Secret

Upon the approval on registration of CorpID Sandbox account, each e-service will be assigned with a unique identifier (client ID) and a client secret key which will be used to identify a particular e-service for authentication and authorised CorpID Sandbox APIs. Developers can obtain the client ID and client secret via the Credential page under the Account Centre of the CorpID Sandbox Developer Portal.

The screenshot shows the CorpID Sandbox Developer Portal interface. At the top, there is a navigation bar with the CorpID Sandbox logo, 'Overview', 'Guides and Tutorials', 'Developer Resources', and 'Account' (with a dropdown arrow). The 'Account' dropdown menu is open, showing options: 'Account Centre', 'Callback Whitelist', and 'Change Password'. The 'Account Centre' option is highlighted with a red box and a red circle labeled '1'. Below the navigation bar, the 'Account Centre' page is displayed. It shows the user profile for 'CP_ORG_USER04' with fields for Company Name (Cyberport HK), Company Phone Number, Company Email Address (cporguser04@cyberport.hk), and Role (IT Contact). Below the profile, there are two input fields: 'Client ID' and 'Client Secret', both containing masked text. These fields are highlighted with a red box and a red circle labeled '2'. Below the input fields, there is a button labeled 'Download KEK certificate (Encipherment Cert):' and a green button labeled 'Download KEK'.

3.2.2. Setup Callback URL whitelist

To enhance data security and prevent callbacks from being sent to unauthorised URLs, e-services are required to add their callback URL into the Callback Whitelist, as some APIs in the CorpID Sandbox Platform will invoke callbacks to the e-service. For details, please refer to the CorpID Simulated API Specifications.



「數碼企業身份」沙盒
CorpID Sandbox

[Overview](#)

[Guides and Tutorials](#)

[Developer Resources](#)

 Account ▾

| [Logout](#) | [ENG](#) | [繁](#) | [簡](#) | [Text Size](#)

Account

> Account Centre

> **Callback Whitelist**

> Change Password

> Logout

Callback Whitelist

Callback Whitelist Settings

http://15.134.147.28:3000/callback/pdfSigningCallback

https://api.eservice.gov.hk/app/bdss/result

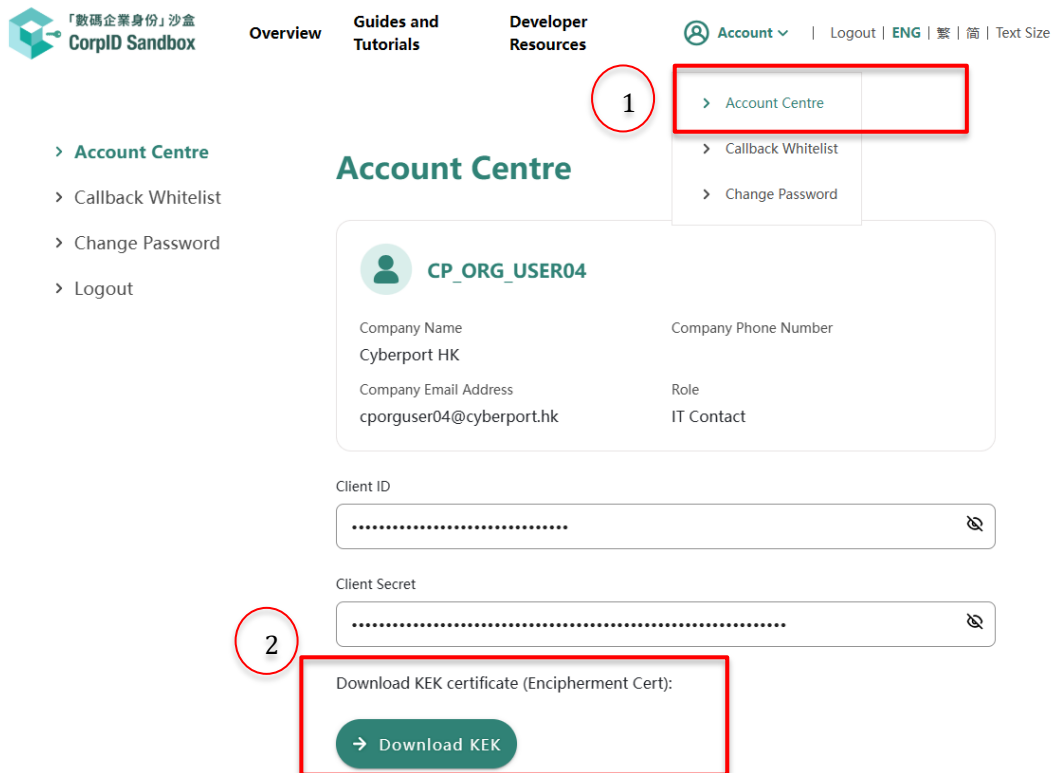
https://example.com/callback

https://15.134.147.28:3000/callback/GeneralCallback

https://api.eservice.gov.hk/app/bdss/callback

3.2.3. Setup KEK certificate (Encipherment Cert)

The API requests and callbacks with the CorpID Sandbox Platform occur over the Internet, data security is enforced by encrypting the request body using a CEK and a KEK. To streamline the testing and saving the need for e-service providers to obtain individual encryption keys from a Hong Kong Recognised Certification Authority, the CorpID Sandbox Platform will provide a Hongkong Post Trial Certificate, enabling faster access for e-services to test the CorpID Sandbox APIs.



The PIN for using the Hongkong Post Trial Certificate is listed below:

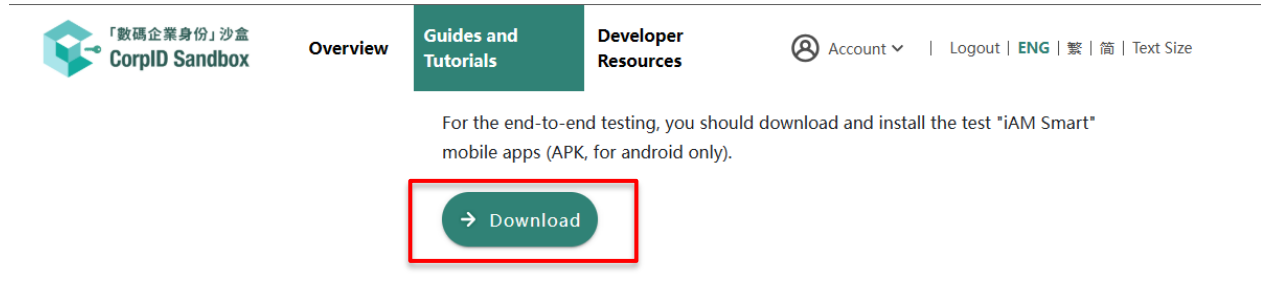
File	PIN
account-centre-kek.p12	8568185550716550

3.3. SETUP OF “IAM SMART” E-SERVICE ACCOUNT

To integrate the Corporate Identity Verification, Digital Signing, and Form Pre-Filling APIs, an “iAM Smart” e-service account in the “iAM Smart” ITE environment is required in addition to the CorpID Sandbox account. If the applicant did not select “Reuse Existing iAM Smart Testing Account” during the registration, the “iAM Smart” ITE platform will send an email to the designated contacts of the e-services, notifying the e-service about the setup necessary in the “iAM Smart” ITE environment. Please refer to the “iAM Smart” ESP user guide for setting up the “iAM Smart” ITE account.

3.4. “IAM SMART” TESTING MOBILE APP

To integrate the Corporate Identity Verification, Digital Signing and Form Pre-Filling APIs, the “iAM Smart” testing mobile app (in form of APK for android mobile devices) can be downloaded from “Guides and Tutorials” page under the CorpID Sandbox Developer Portal for end-to-end testing on the selected scenarios.



4. IMPLEMENTATION PROCEDURE

The CorpID Sandbox Platform provides a comprehensive testing environment designed specifically for selected scenarios for web-based applications. The workflow of mobile applications is similar to web-based applications. For details, please refer to the CorpID Sandbox API Specifications.

It offers four dedicated end-to-end workflows that allow developers to conduct Proof-of-Concept (PoC) testing before the CorpID system launch. These workflows combine CorpID Simulated APIs, a CorpID mock-up mini-program, “iAM Smart” Simulated APIs, and the “iAM Smart” mock-up mobile app to simulate the complete user journey with e-service’s web application.

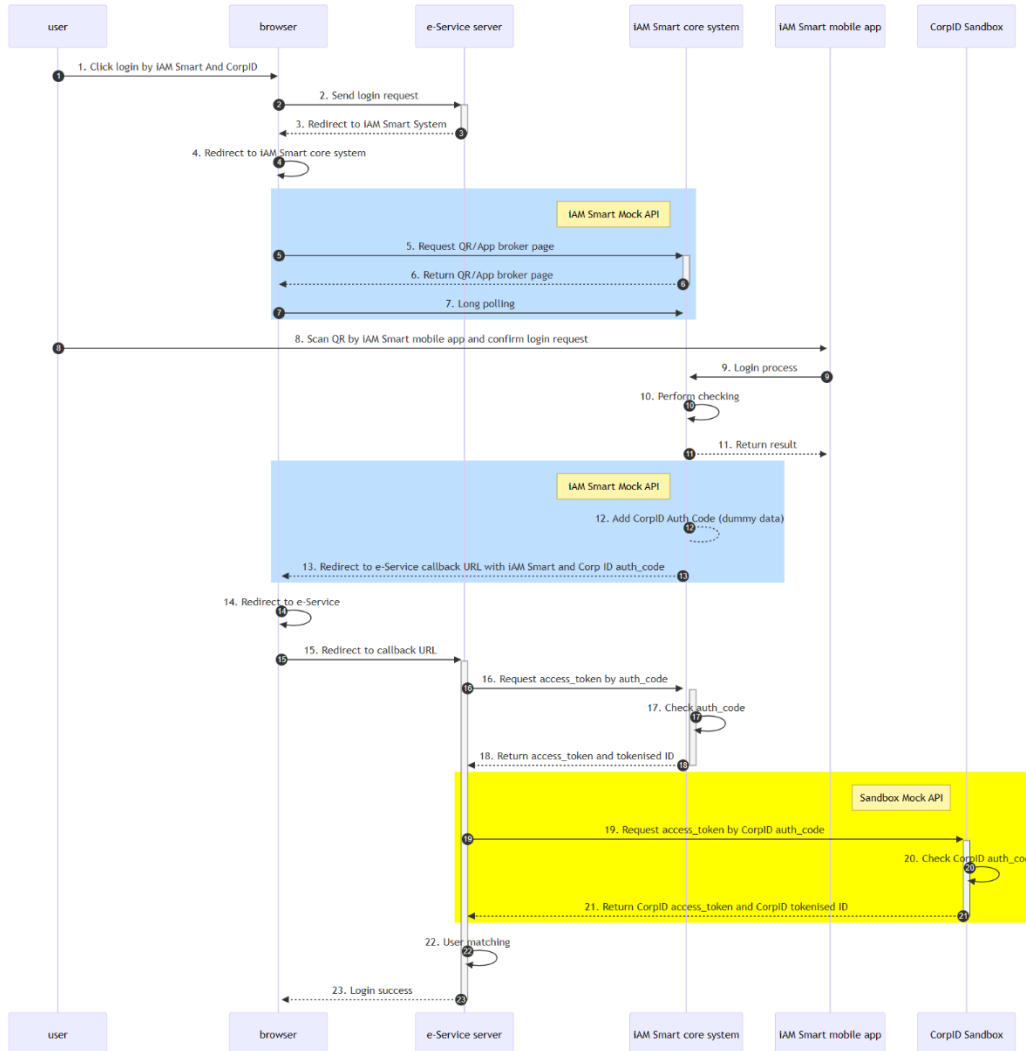
With these four workflows, developers can test the core CorpID functions including CEK/KEK encryption, Corporate Identity Verification, form pre-filling (with and without service login), and electronic signing in the CorpID Sandbox Platform.

Remarks: The steps outline in the following sections are subject to change. For the most up-to-date information and technical details of each of these APIs, please refer to the latest API specifications published in the CorpID Sandbox Developer Portal.

4.1. CORPORATE IDENTITY VERIFICATION WORKFLOW

The sequence diagram below shows how an e-service in form of a web application leverages the CorpID Sandbox Platform and the “iAM Smart” to perform user authentication via the “iAM Smart” testing mobile app.

CorpID Sandbox Authentication (Online Service Website in Different Device)



For a complete authentication process, the e-service should follow these steps:

Step 1. The user clicks “Login by iAM Smart and CorpID” button on the e-service;

Step 2. The e-service initiates a login request to e-service Server with the browser’s user agent value (for use as value for request parameter “source”);

Step 3-4. The e-service Server prepares the request parameters such as “clientID”, “redirectURI”, “source”, “scope”, etc. and constructs the GET request to invoke the “iAM Smart” API using browser redirection;

(For details, please refer to Section 1.1.2 of Appendix A.)

Step 6-7. The browser opens the broker page and displays the QR Code to user;

Step 8-9. “iAM Smart” user logs in “iAM Smart” testing mobile app and scan the QR Code and confirms Online Service's login request;

Step 10-11. “iAM Smart” System verifies the validity of QR code and other necessary information, and responds the login result to “iAM Smart” testing mobile app (e.g., login success and inform “iAM Smart” user to proceed in Online Service, invalid / expired QR code, etc.);

Step 12. “iAM Smart” includes the CorpID authCode in the callback request

Step 13-15. “iAM Smart” Simulated API automatically injects a mock-up CorpID authentication code;

(For details, please refer to Section 1.1.3 of Appendix A.)

Step 16-18: When e-service Server gets the “iAM Smart” authCode, it should invoke the “iAM Smart” API to obtain the accessToken and the Tokenised ID (i.e. openID) of the “iAM Smart” user. API data encryption is required;

(For details, please refer to Section 1.1.4 of Appendix A.)

Step 19-21: When e-service Server gets the CorpID mock-up authCode, it should invoke the CorpID API to obtain the accessToken and the tokenised CorpID user ID (i.e. openID) of the CorpID user. API data encryption is required;

(For details, please refer to Section 3.1 of the CorpID Simulated API Specifications.)

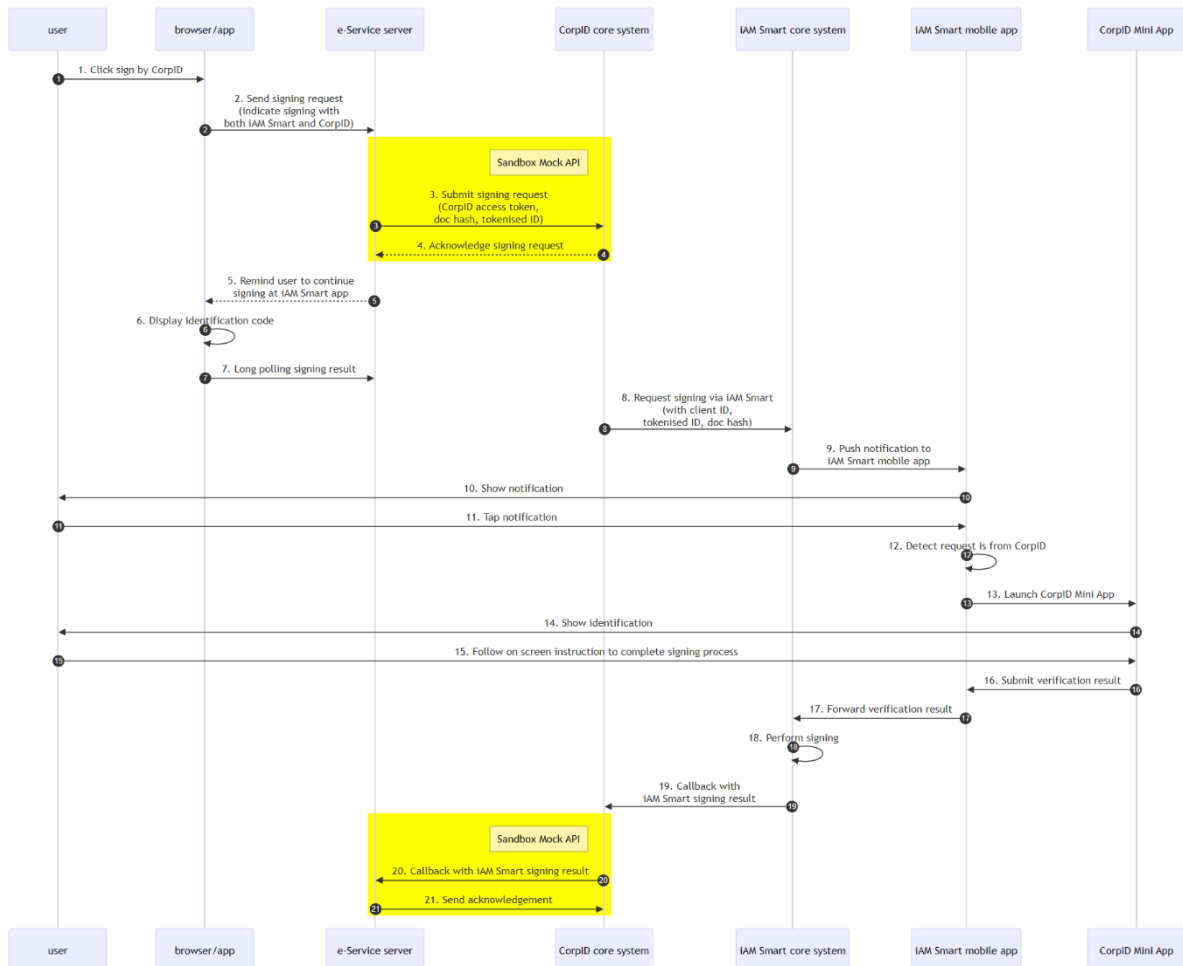
Step 22: e-service Server then performs user matching using the CorpID Tokenised ID with its user repository and determines the result of this login request (e.g. if the CorpID Tokenised ID doesn't exist in e-service system, redirect the user to register);

Step 23. After successful matching, e-service Server completes the login process and returns a login-success page/response to the browser.

4.2. DIGITAL SIGNING (PDF) WORKFLOW

An e-service can initiate a Digital Signing request after service login with the CorpID Sandbox authentication APIs. The sequence diagram below shows how an authenticated user authorises and signs the document hash when an e-service website/App and the "iAM Smart" testing mobile app and CorpID mock-up mini-program.

CorpID Sandbox Digital Signing (Online Service Website in Different Device)



For a complete Digital Signing with Service Login, the e-service should go through the following steps:

Step 1. User clicks the “Sign with CorpID” button (or similar) on the e-service web application.

Step 2. The browser sends a signing request to the e-service server, explicitly indicating that the document must be signed using both CorpID and “iAM Smart”.

Step 3–4. The e-service server submits the first signing request to the CorpID Sandbox Platform, including the CorpID access token, document hash, and tokenised CorpID user ID. The CorpID Sandbox Platform immediately acknowledges receipt.

(For details, please refer to Section 3.11 of the CorpID Simulated API Specifications.)

Step 5. The e-service server responds to the browser with a message instructing the user to continue the signing process in the "iAM Smart" mobile app.

Step 6. The browser displays an identification code (or waiting screen) to the user.

Step 7. The browser starts long-polling the e-service server to obtain the eventual signing result.

Step 8. The CorpID core system requests the "iAM Smart" core system to perform signing on behalf of the same user, providing the "iAM Smart" client ID, tokenised CorpID user ID, and document hash.

Step 9–10. The "iAM Smart" core system pushes a notification to the user's mobile device for display.

Step 11. The user taps the notification message.

Step 12. The "iAM Smart" mobile app detects that the signing request originated from CorpID.

Step 13. The "iAM Smart" testing mobile app launches the CorpID mock-up mini-program inside the "iAM Smart" environment.

Step 14. The CorpID mock-up mini-program displays the consent page to the user.

Step 15. The user follows the on-screen instructions in the CorpID mock-up mini-program (e.g. review document details, confirm signing intent) to complete the "iAM Smart" signing process.

Step 16–17. The CorpID mock-up mini-program submits the verification result back to the "iAM Smart" core system via the "iAM Smart" mobile app.

Step 18. The "iAM Smart" core system performs the actual digital signing using the user's "iAM Smart" signing credential.

Step 19. The "iAM Smart" core system calls back the CorpID core system with the "iAM Smart" signing result (signature, timestamp, etc.).

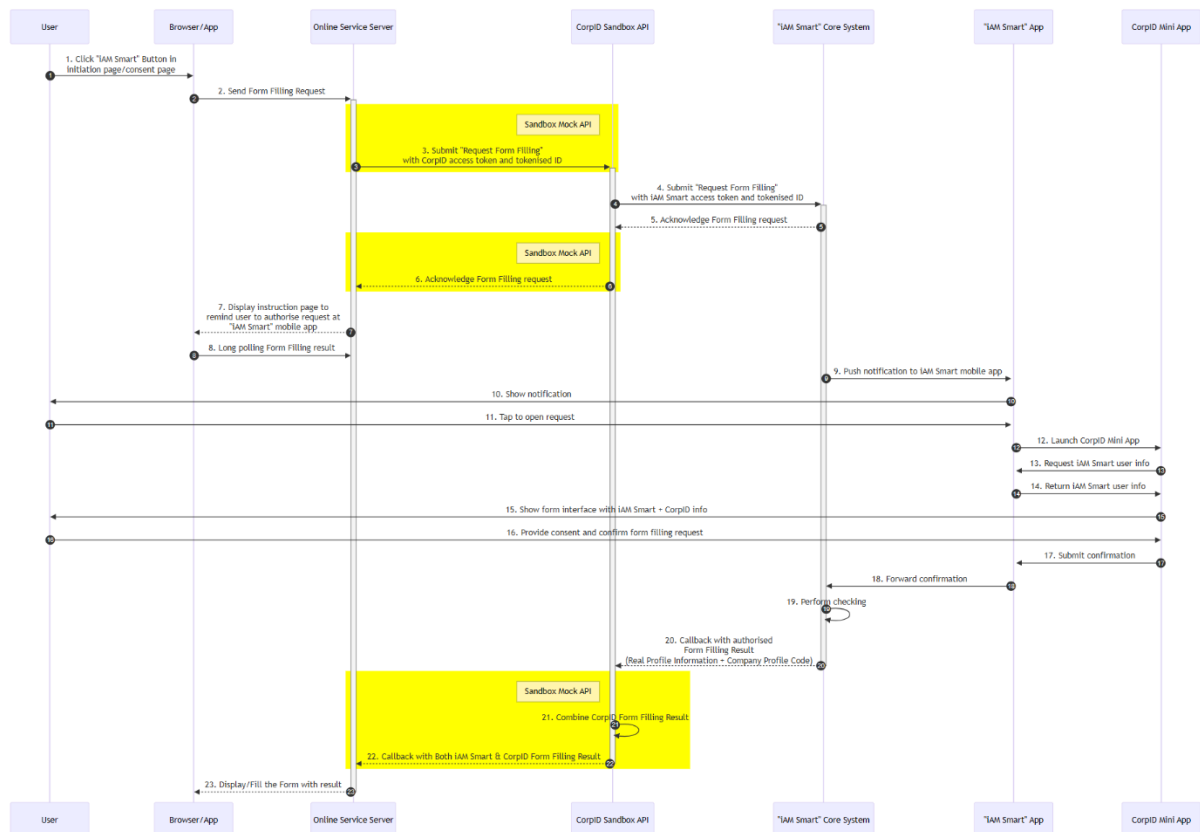
Step 20–21. The CorpID Sandbox Platform immediately forwards the "iAM Smart" signing result to the e-service server via callback. The e-service server acknowledges receipt.

4.3. FORM PRE-FILLING WORKFLOW

e-service can initiate a Form Pre-Filling request after service login with the CorpID Sandbox authentication APIs. e-service will receive the authorised data after the user consents to the authorisation page.

The typical use cases cover account link-up, application form system, remote account onboarding, identity verification and address verification.

CorpID Sandbox Form Filling (Online Service Website in Different Device)



For a complete Form Pre-Filling Workflow, e-service should go through the following steps:

Step 1. User clicks the “iAM Smart” button on the e-service initiation/consent page (this indicates the user wishes to pre-fill the form using both personal and corporate data via “iAM Smart” + CorpID).

Step 2. The browser sends a Form Pre-Filling request to the Online Service Server.

Step 3. The Online Service Server submits a “Request Form Pre-Filling” to the CorpID Sandbox API, including the CorpID access token and the tokenised CorpID user ID.

(For details, please refer to Section 3.2 of the CorpID Simulated API Specifications.)

Step 4. The CorpID Sandbox Platform forwards the “Request Form Pre-Filling” to the “iAM Smart” Core System.

Step 5. The “iAM Smart” Core System acknowledges the request back to the CorpID Sandbox Platform.

Step 6. The CorpID Sandbox Platform immediately acknowledges the original request back to the Online Service Server.

(Refer to CorpID API Spec Section 3.2)

Step 7. The Online Service Server returns an instruction/waiting page to the browser, reminding the user to complete authorisation in the CorpID mock-up mini-program.

Step 8. The browser/app starts long-polling the Online Service Server for the final form-filling result.

Step 9–10. The “iAM Smart” Core System pushes a notification to the user’s CorpID mock-up mini-program, which displays the notification.

Step 11. The user taps the notification message to open the authorisation request.

Step 12. The “iAM Smart” App detects that the request originated from CorpID Sandbox Platform and launches the CorpID mock-up mini-program.

Step 13–14. The CorpID mock-up mini-program requests the user’s “iAM Smart” personal information; the “iAM Smart” App returns the requested personal profile data.

Step 15–16. The CorpID mock-up mini-program displays the form-filling interface pre-populated with both “iAM Smart” personal data and CorpID corporate data, and asks the user to review and confirm the form-filling request .

Step 17–21. Once the user confirms the form-filling request, CorpID mock-up mini-program would process the internal checking and forward the request to CorpID Sandbox Platform.

Step 22. The CorpID Sandbox Platform combines its own corporate data with the received “iAM Smart” result and immediately calls back the Online Service Server with the complete combined Form Pre-Filling Result (both personal and corporate data).

(For details, please refer to Section 3.4 of the CorpID Simulated API Specifications.)

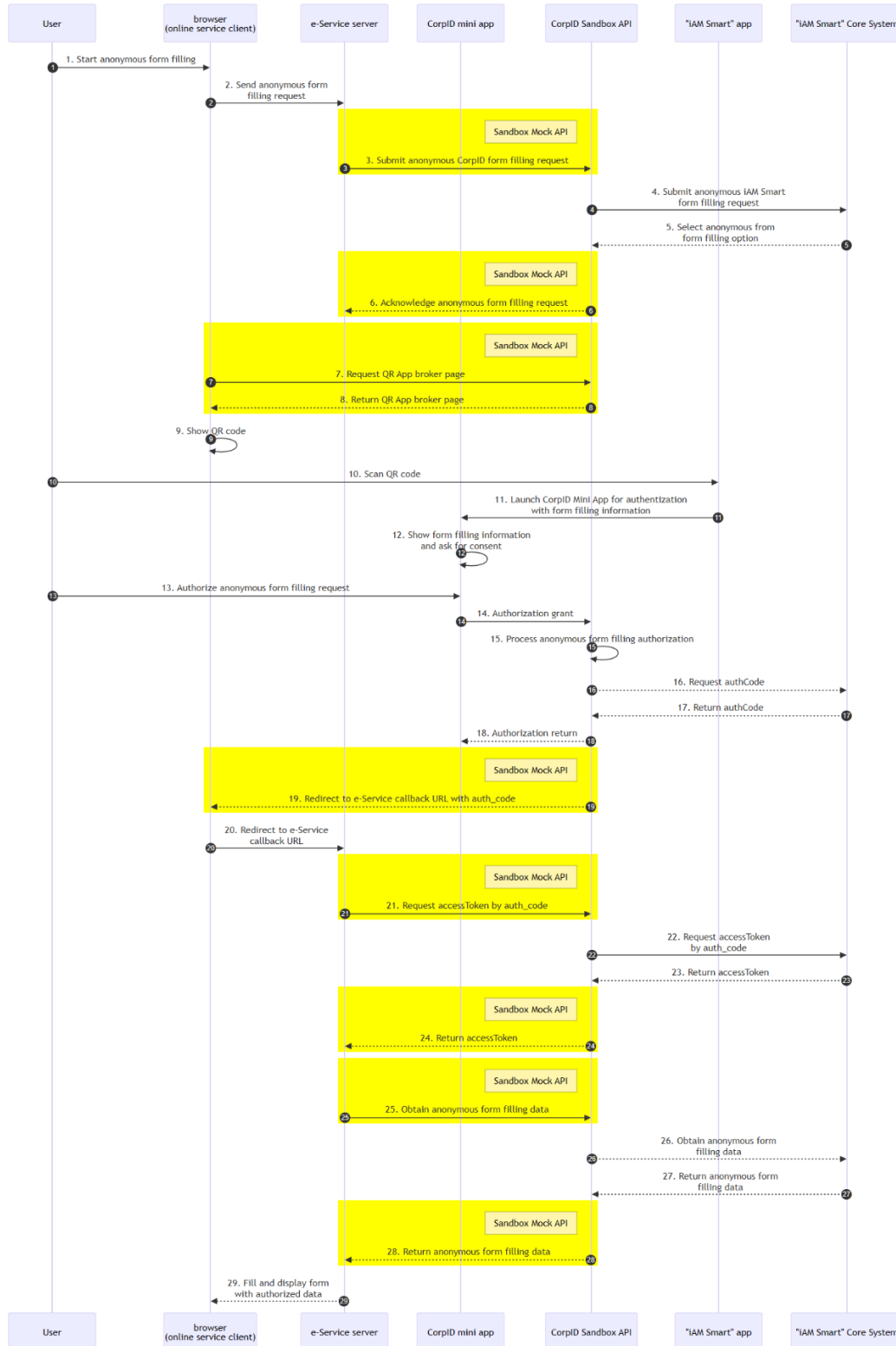
Step 23. The Online Service Server receives the full result, automatically fills the electronic form on behalf of the user, and displays the completed form (or proceeds to the next step) in the browser/app.

4.4. FORM PRE-FILLING WITHOUT SERVICE LOGIN WORKFLOW

e-service can initiate a Form Pre-Filling request without requiring service login. e-service will receive the authorise code after the user consents to the authorisation page and get the authorised data.

The typical use cases cover registration form system, identity verification and address verification.

CorpID Sandbox Anonymous Form Filling (Online Service Website in Different Device)



For a complete Anonymous Form Pre-Filling Workflow, e-service should go through the following steps:

Step 1. User begins filling an online form and chooses the anonymous “iAM Smart” + CorpID form-filling option (no prior login session).

Step 2. The browser sends an anonymous form-filling request to the e-service server.

Step 3. The e-service server submits an anonymous CorpID form-filling request to the CorpID Sandbox API.

(For details, please refer to Section 3.5 of the CorpID Simulated API Specifications.)

Step 4-5. The CorpID Sandbox Platform forwards an anonymous “iAM Smart” form-filling request to the “iAM Smart” Core System.

Step 6. The CorpID Sandbox Platform immediately acknowledges the original anonymous request back to the e-service server.

(Refer to CorpID API Spec Section 3.5)

Step 7–9. The browser open the broker page and display the QR code to user.

(Refer to CorpID API Spec Section 4.1)

Step 10-12. “iAM Smart” user logs in “iAM Smart” testing mobile app to scan the QR Code and the CorpID mock-up mini-program prompts and displays the exact data that will be released for anonymous form pre-filling and asks for user consent.

Step 13. The user reviews the data and grants consent for anonymous form pre-filling.

Step 14-18. The CorpID mock-up mini-program and CorpID perform internal checking and display successful message to user.

Step 19. The CorpID Sandbox Platform redirects the browser to the e-service callback URL, appending the auth_code.

(For details, please refer to Section 3.6 of the CorpID Simulated API Specifications.)

Step 20. The browser follows the redirect and lands on the e-service server’s callback endpoint.

Step 21–24. The e-service server exchanges the received auth_code for an access token with the CorpID Sandbox Platform. The CorpID Sandbox Platform returns the access token to the e-service server after checking the auth_code.

(For details, please refer to Section 3.1 of the CorpID Simulated API Specifications.)

Step 25-28. Using the access token, the e-service server requests the anonymous form-filling data from the CorpID Sandbox Platform. The CorpID Sandbox Platform returns the anonymous form-filling data to the e-service server after checking the access token.

(For details, please refer to Section 3.7 of the CorpID Simulated API Specifications.)

Step 29. The e-service server automatically fills the form with the authorised anonymous data and displays the completed form to the user in the browser.

APPENDIX

A. IAM SMART SIMULATED API FOR CORPID SANDBOX

1.1.1 Host information of the “iAM Smart” system in Sandbox

The following describes the API information of the “iAM Smart”. <iAM_Smart_domain> will be used to concatenate the website URL or API URL for API call. e-service should replace <iAM_Smart_domain> with the URL listed in the table below.

Domain	Sandbox Environment
iAM_Smart_domain	apigw-isit.staging-eid.gov.hk

1.1.2 Request QR Page

● API Description

Name	Description
Service Full Name	Request QR Page
URI (as in RESTFUL API)	https:// <iAM_Smart_domain>/api/v1/auth/mockCorpId/getQR
Request Type	GET
Service Version	1.0.0
Description of Service	Online service calls this API to get the QR/App broker page or QR page. After the user authorises login or anonymous request on the “iAM Smart” testing mobile app, the page will be redirected to the redirectURI with authCode and state parameters. If the user denies it, only the state parameter will be redirected.

● Request Parameters

Parameter	Type	Presence	Description
clientID	String	Required	iAM Smart Online service client identifier. The clientID will be assigned to online service at the initial registration. Please refer to the iAM Smart self-service portal.
responseType	String	Required	The value MUST be set to code.
Source	String	Required	Request initiator. The supported source values can be found in Appendix A of the “iAM Smart” API specification. Except (App_Link and App_Scheme).
redirectURI	String	Required	The callback redirect URI. The value should be URL encoded and registered in the “iAM Smart” self-service portal.
scope	String	Required	The value should be URL encoded. Specify the scope of “iAM Smart” functions: - eidapi_auth - eidapi_profiles - eidapi_formFilling - eidapi_sign - eidapi_fr - eidapi_bulksign - eidapi_sua The value of the scope parameter is expressed as a list of space-delimited, case-sensitive strings.

lang	String	Optional	Language to display: en-US, zh-HK, or zh-CN. If this parameter is not specified, zh-HK will be shown.
state	String	Optional	If the state parameter is presented in the request message, the same state value will be returned to online service during the callback. It is used to prevent the CSRF attack. The value of state is defined by online service and it should be a secure random string of any length less than or equal to 36 (UUID string length). Only ASCII letters, numbers, underscore and hyphens are accepted.
brokerPage	Boolean	Optional	If brokerPage is set to true, Universal Link (iOS) / App Link (Android) will be leveraged to open the “iAM Smart” testing mobile app. This feature is useful for online service supporting mobile web versions while triggering the “iAM Smart” testing mobile app or showing a QR page automatically. (i.e. Show QR page without detecting whether the “iAM Smart” testing mobile app is installed). The default value is false.
corpMock	Boolean	Optional	The value MUST be set to true in CorpID Sandbox Platform

● Example Request

```
// Line breaks are for legibility only.
// Please refer to Section 2.4 of the "iAM Smart" API Specification for generating shared common parameters/
header format.
GET
https://<iAM_Smart_domain>/api/v1/auth/getQR
?clientId=Online Service1
&responseType=code
&source= PC_Browser
&redirectURI=https%3A%2F%2Frp_domain%2Frp_context%2Fcall_back_endpoint
&scope=eidapi_auth%20eidapi_formFilling%20eidapi_profiles
&lang=en-US
&state=eb9b7b8eddd5
&corpMock=true
```

● Response Parameters

N/A

1.1.3 Callback with authCode to Online Service Server

● API Description

Name	Description
Service Full Name	Callback with authCode to Online Service Server
URI (as in RESTFUL API)	https://<rp_domain>/<rp_context>/<call_back_endpoint>
Request Type	GET
Service Version	V1.0.0
Description of Service	This callback is used to pass authCode to online service Server. The URI must be registered in the self-service portal.

- **API Specific Timeout**

Title	Timeout value	Description
Callback	18 minutes	Online service could treat the request as failed if it doesn't get callback within 18 minutes

- **Callback Parameters**

Parameter	Type	Presence	Description
code	String	Required (Conditional)	The authorisation code is generated by the “iAM Smart” System. The authorisation code will be expired 60 seconds after issuance. Online service MUST NOT use the authorisation code more than once. An error message will be returned if an authorisation code is expired or re-used.
corpMockCode	String	Required (Conditional)	The authorisation code for the CorpID Sandbox Authentication workflow. It will be expired 60 seconds after issuance. Online service MUST NOT use the authorisation code more than once. An error message will be returned if an authorisation code is expired or re-used.
state	String	Optional	If the state parameter has been presented in the request, the same state value will be returned during the callback. It is used to prevent the CSRF attack. The value of the state is defined by online service and it should be a secure random value.
error_code	String	Required (Conditional)	Return error code when exception.

- **Example Callback**

Allow

```
// Line breaks are for legibility only.
GET
https://<call_back_endpoint>
?code=0ad186353c424c64897fcc00445c9ba1
&corpMockCode=c12a6bccfa044230b2735547e905dd20
&state=eddd527b6
```

Deny

```
// Line breaks are for legibility only.
```

```
GET
https://<call back endpoint>
?error_code=D20001
&state=eddd527b6
```

1.1.4 Request accessToken & Tokenised ID

● API Description

Name	Description
Service Full Name	Request access token and tokenised ID with authCode
URI (as in RESTFUL API)	https://<iAM_Smart_domain>/api/v1/auth/getToken
Request Type	POST
Service Version	1.0.0
Description of Service	Online service uses this API to retrieve the access token and Tokenised ID (openID). An authorisation code is necessary during the process. The accessToken and openID will be used to call corresponding “iAM Smart” services subsequently.

● Request Parameters

Parameter	Type	Presence	Description
code	String	Required	The authorisation code is received from the authorisation server. It can only be used once.
grantType	String	Required	the value MUST be set to authorization_code.

● Example Request

```
// Line breaks are for legibility only.
// Please refer to Section 2.4 of the "iAM Smart" API Specification for generating shared common parameters/
header format.
POST
https://<iAM_Smart_domain>/api/v1/auth/getToken
// Request Headers
clientID: "edae2e2529ff46228af1e4d18c8405d1"
signatureMethod: "HmacSHA256"
signature: "5X42Y1B7MEd8Mm%2BonwjiQz9VCZkkrntADskXsYntavU%3D"
timestamp: 1557048906183
nonce: "e893647dc4204eb9b7b8eddd527b687c"
// Unencrypted Request Body (authCode from Direct Login v2)
{
  "code": "xxxxa42e76bf4cb0846a68e6d83d6096",
  "grantType": "authorization_code"
}

// Unencrypted Request Body(authCode from other)
{
  "code": "xxxxa42e76bf4cb0846a68e6d83d6096",
  "grantType": "authorization_code"
}
```

● Response Parameters

Parameter	Type	Presence	Description
accessToken	String	Required	accessToken value can be used multiple of times before expiry.
tokenType	String	Required	Token type, support "Bearer" only.

issueAt	Long	Required	The accessToken issue time is expressed in the number of milliseconds since January 1, 1970 00:00:00 GMT.				
expiresIn	Long	Required	The lifetime in milliseconds of the token. The value may vary for different Online Services.				
openID	String	Required	Tokenised ID, uniquely generated for each user of each online service website or mobile app.				
lastModifiedDate	Long	Required	The datetime of the user complete registration at “iAM Smart” System. The value will be updated when either of the following is valid:- (1) If any one of the following verified data are changed. <table><tr><td>English name</td></tr><tr><td>Chinese name (* not applicable if it was marked as unverified during registration)</td></tr><tr><td>Gender</td></tr><tr><td>Date of birth</td></tr></table> (2) User re-register “iAM Smart” after “iAM Smart” de-registration. The modification time will be expressed in the number of milliseconds since January 1, 1970 00:00:00 GMT.	English name	Chinese name (* not applicable if it was marked as unverified during registration)	Gender	Date of birth
English name							
Chinese name (* not applicable if it was marked as unverified during registration)							
Gender							
Date of birth							
userType	String	Required	default or sign default: “iAM Smart” user sign: “iAM Smart+” user (digital signing capability)				
scope	String	Required	The scope of the token. Please refer to the corresponding section specified in each API function.				

● Example Success Response

```
// The descriptions of txID, code, and message are in Section 2.4 of the "iAM Smart" API Specification
// The decrypted body
{
  "txID": "<T=938ffb193b4b4370b6c2584372c6a588>",
  "code": "D00000",
  "message": "SUCCESS",
  "content": {
    "accessToken": "0ad186353c424c64897fcc00445c9ba1",
    "tokenType": "Bearer",
```

```
"issueAt": 1557053922938,  
"expiresIn": 14400000,  
"openID": "liR14%2BvX%2F5hSum5uf4ERczu0KcDnIJA5BM7FoM1ag9c%3D",  
"lastModifiedDate": 1560849218006,  
"userType": "sign",  
"scope": "eidapi_auth eidapi_formFilling"  
}  
}
```

- **Example Error Response**

```
// The descriptions of txID, code, and message are in Section 2.4  
// The decrypted body  
{  
  "txID": "<T=938ffb193b4b4370b6c2584372c6a588>",  
  "code": "D40004",  
  "message": "authCode not exist or expired",  
}
```

End of Document