CPL-01:行動應用 App 應設計並實作適當身分認證機制，並依使用者身分授權，以防止敏感資料被非授權人員存取。

安全程式碼範例(Android：Java)

```
•  //定義使用 ACCOUNT_MANAGER 與 INTERNET 權限
   <manifest ... >
       <uses-permission android:name="android.permission.ACCOUNT_MANAGER" />
       <uses-permission android:name="android.permission.INTERNET" />
       ...
   </manifest>
   //實作 CallBack
   AccountManager am = AccountManager.get(this);
   Bundle options = new Bundle();
   am.getAuthToken(
       myAccount_,                     // Account retrieved using getAccountsByType()
       "Manage your tasks",            // Auth scope
       options,                        // Authenticator-specific options
       this,                           // Your activity
       new OnTokenAcquired(),          // Callback called when a token is successfully acquired
       new Handler(new OnError()));    // Callback called if an error occurs

•  private class OnTokenAcquired implements AccountManagerCallback<Bundle> {
       @Override
       public void run(AccountManagerFuture<Bundle> result) {
           ...
           Intent launch = (Intent) result.getResult().get(AccountManager.KEY_INTENT);
           if (launch != null) {
               startActivityForResult(launch, 0);
               return;
           }
       }
   }
   //獲取 token
   private class OnTokenAcquired implements AccountManagerCallback<Bundle> {
       @Override
       public void run(AccountManagerFuture<Bundle> result) {
           // Get the result of the operation from the AccountManagerFuture.
           Bundle bundle = result.getResult();

           // The token is a named value in the bundle. The name of the value
           // is stored in the constant AccountManager.KEY_AUTHTOKEN.
           token = bundle.getString(AccountManager.KEY_AUTHTOKEN);
           ...
       }
   }
   //呼叫 OAuth2 服務
   URL url = new URL("https://www.googleapis.com/tasks/v1/users/@me/lists?key=" +
   your_api_key);
   URLConnection conn = (HttpURLConnection) url.openConnection();
   conn.addRequestProperty("client_id", your client id);
   conn.addRequestProperty("client_secret", your client secret);
   conn.setRequestProperty("Authorization", "OAuth " + token);
```

安全程式碼範例(iOS：Swift)-參考 CPF-13 安全範例：使用 OAuth2。

通常會使用第三方帳號認證，茲提供 OAuth2 範例，以取得第三方的 Token，當作認證 ID

OAuth 2 應用範例

1.註冊並宣告服務

```
let googleConfig = GoogleConfig(
clientId: "YOUR_GOOGLE_CLIENT_ID",                            // [1] Define a Google
configuration
scopes:["https://www.googleapis.com/auth/drive"])             // [2] Specify scope

let gdModule = AccountManager.addGoogleAccount(googleConfig)  // [3] Add it to
AccountManager
self.http.authzModule = gdModule                              // [4] Inject the
AuthzModule
                                                              // into the HTTP layer
object

let multipartData = MultiPartData(data: self.snapshot(),      // [5] Define multi-
part
          name: "image",
          filename: "incognito_photo",
          mimeType: "image/jpg")
let multipartArray =  ["file": multipartData]

self.http.POST("https://www.googleapis.com/upload/drive/v2/files",   // [6] Upload image
               parameters: multipartArray,
       completionHandler: {(response, error) in
  if (error != nil) {
    self.presentAlert("Error", message: error!.localizedDescription)
  } else {
    self.presentAlert("Success", message: "Successfully uploaded!")
  }
})
```

2.註冊 App 應用連結

```
<key>CFBundleURLTypes</key>
<array>
    <dict>
        <key>CFBundleURLSchemes</key>
        <array>
            <string>com.raywenderlich.Incognito</string>
        </array>
    </dict>
</array>
```

3.接收 App 應用連結

```
func application(application: UIApplication,
  openURL url: NSURL,
  sourceApplication: String?,
  annotation: AnyObject?) -> Bool {
    let notification = NSNotification(name: AGAppLaunchedWithURLNotification,
      object:nil,
      userInfo:[UIApplicationLaunchOptionsURLKey:url])
    NSNotificationCenter.defaultCenter().postNotification(notification)
    return true
}
```

CPQ-01:行動應用 App 應實作驗證使用者輸入字串資料型別及長度之正確性，避免惡意輸入導致應用程式毀損、緩衝區溢位、各種注入攻擊發生。

安全程式碼範例(iOS:Objective-C)

驗證輸入長度、格式Email

如

```
- (BOOL) validEmail:(NSString*) emailString {
    if([emailString length]==0){
        return NO;
    }
    NSString *regExPattern = @"[A-Z0-9a-z._%+-]+@[A-Za-z0-9.-]+\\.[A-Za-z]{2,4}";
    NSRegularExpression *regEx = [[NSRegularExpression alloc]
initWithPattern:regExPattern options:NSRegularExpressionCaseInsensitive error:nil];
    NSUInteger regExMatches = [regEx numberOfMatchesInString:emailString options:0
range:NSMakeRange(0, [emailString length])];
    if (regExMatches == 0) {
        return NO;
    } else {
        return YES;
    }
}
```

安全程式碼範例(網頁主機：Server .NET C#)

- 驗證輸入長度、格式
  .NET 使用 RegularExpressionValidator ，使用正規表示式來驗證使用者輸入的格式。
  例如檢查英數字輸入 6 至 10 位。
  ValidationExpression="[a-zA-Z]{6,10}"