

BAN CƠ YẾU CHÍNH PHỦ
CỤC CHỨNG THỰC SỐ VÀ BẢO MẬT THÔNG TIN

TÀI LIỆU HƯỚNG DẪN
TÍCH HỢP GIẢI PHÁP KÝ SỐ TRÊN THIẾT BỊ DI
ĐỘNG SỬ DỤNG SIM-PKI

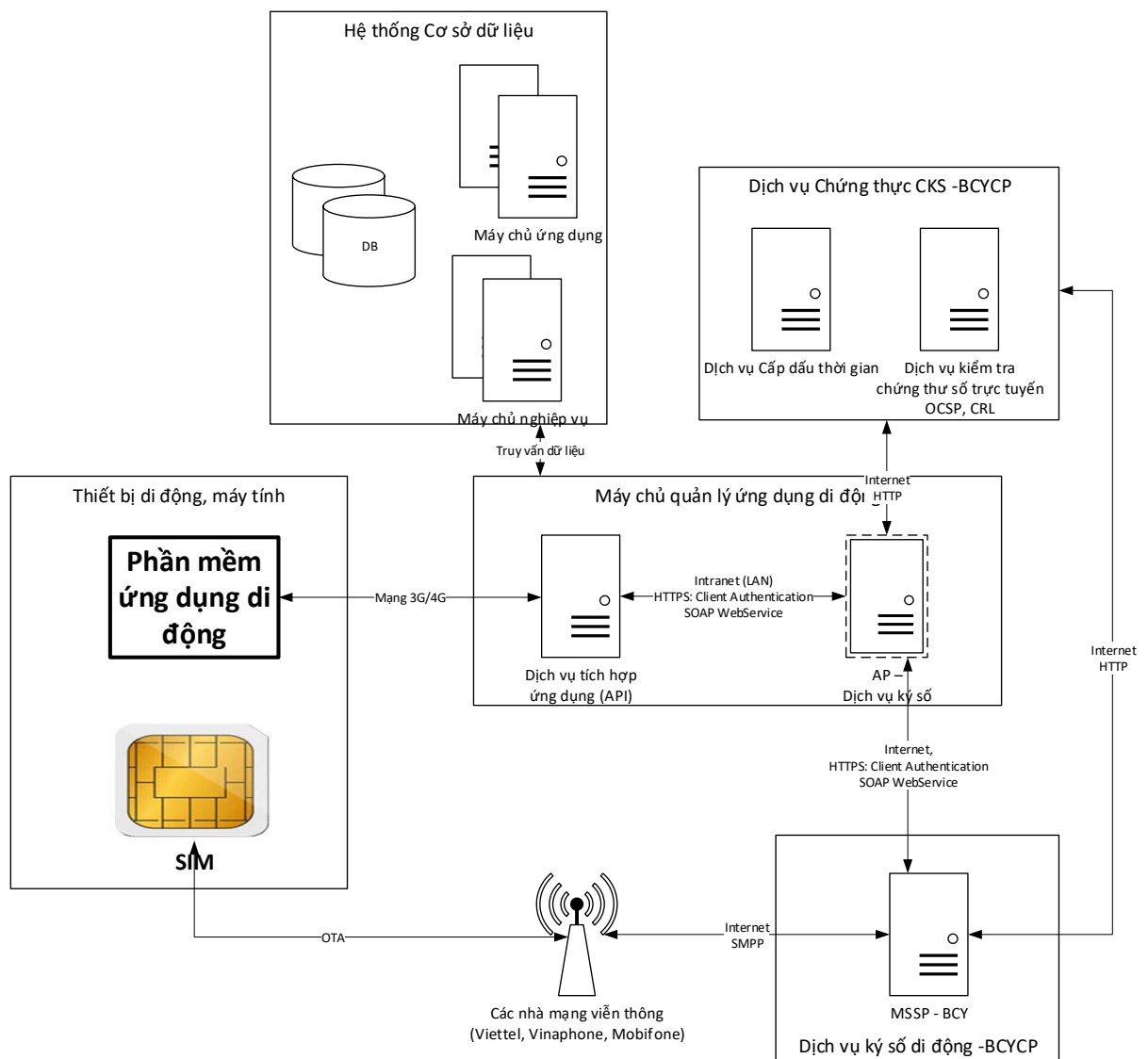
Hà Nội, 2022

HƯỚNG DẪN TÍCH HỢP CHỮ KÝ SỐ SỬ DỤNG SIM-PKI

NỘI DUNG

I. Giới thiệu giải pháp tích hợp SIM-PKI	3
1. Thông tin dịch vụ MSSP	3
2. Luồng dữ liệu ký số.....	4
3. Danh sách tệp	5
4. Điều kiện tích hợp dịch vụ MSSP	5
II. Hướng dẫn triển khai tích hợp.....	6
1. Hướng dẫn tích hợp trên môi trường dotnet	6
2. Hướng dẫn tích hợp trên môi trường Java	8
3. Danh mục mã lỗi (Exception code).....	9
III. Hướng dẫn một số trường hợp sử dụng SIM-PKI	11
1. Hướng dẫn đổi mã PIN của SIM-PKI.....	11
2. Hướng dẫn bật, tắt chuông báo của SIM-PKI.....	22
3. Hướng dẫn xem phiên bản SIM-PKI	27

I. Giới thiệu giải pháp tích hợp SIM-PKI



Hình 1. Mô hình giải pháp SIM-PKI

SDK-MobilePKI là Bộ công cụ hỗ trợ phát triển ứng dụng ký số trên thiết bị di động sử dụng SIM-PKI, cung cấp các API cơ bản như tạo chữ ký số RSA với SIM-PKI và lấy thông tin chứng thư số tương ứng với số điện thoại. Bộ công cụ cung cấp dưới dạng dịch vụ Web Service (Mobile Signature Service Provider – MSSP) trên Internet, cụ thể như sau:

1. Thông tin dịch vụ MSSP

- Địa chỉ dịch vụ:
https://mpki1.ca.gov.vn:18083/soap/services/MSS_SignaturePort
- Giao thức kết nối: TLSv1.2, xác thực hai chiều
- Giao diện kết nối: SOAP 1.1 hoặc SOAP 1.2

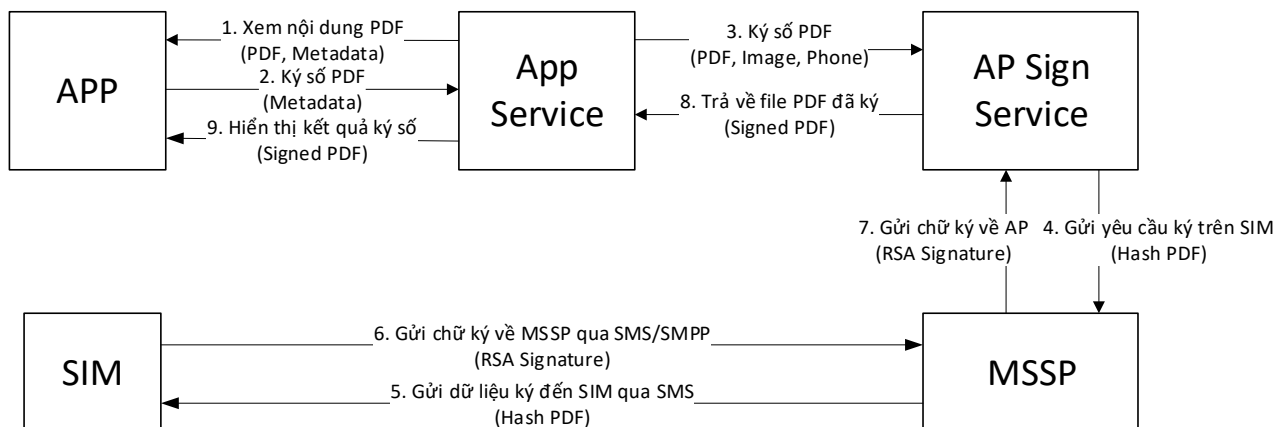
b. APIs

- Lấy thông tin chứng thư theo số điện thoại
- Ký số theo tiêu chuẩn PKCS#1 v1.5 và v2.1 (RSA)

c. Yêu cầu kết nối

- Phần mềm ứng dụng điều hành tác nghiệp của cơ quan, đơn vị phải có kết nối Internet.
- Cơ quan, đơn vị xây dựng dịch vụ AP Sign Service làm nhiệm vụ trung gian kết nối với MSSP (hoặc tích hợp trực tiếp MSSP Webservice vào phần mềm ứng dụng)
- Cơ quan, đơn vị cần cung cấp địa chỉ IP tĩnh của máy chủ triển khai dịch vụ AP, để cập nhật lên hệ thống MSSP của BCYCP cho phép kết nối và đề nghị cấp chứng thư số để kết nối dịch vụ MSSP.
- BCYCP sẽ cung cấp chứng thư số để dịch vụ AP của đơn vị có thể kết nối MSSP (đáp ứng TLSv1.2 Client Authentication).

2. Luồng dữ liệu ký số



Hình 2. Luồng dữ liệu ký số với SIM-PKI

- Bước 1: Người dùng mở giao diện App trên di động xem nội dung tệp PDF cần ký
- Bước 2: Người dùng thao tác ký số trên giao diện App, App sẽ gửi thông tin cần thiết về file PDF cần ký cho App Service.
- Bước 3: App Service lấy nội dung PDF cần ký, thông tin ảnh chữ ký của người ký, số điện thoại người ký và gọi API ký số trên AP Sign Service.
- Bước 4: AP Sign Service thực hiện băm nội dung tài liệu PDF cần ký số và gửi yêu cầu ký số đến SIM thông qua MSSP

- Bước 5: MSSP gửi yêu cầu ký số nhận được từ AP Sign Service đến SIM thông qua SMS với dữ liệu bám nội dung PDF.
- Bước 6: SIM thực hiện ký số và tra về giá trị chữ ký số RSA Signature cho MSSP thông qua SMS.
- Bước 7: MSSP trả về giá trị chữ ký số RSA Signature cho AP Sign Service để đóng gói theo định dạng PDF và lưu thành file PDF đã ký.
- Bước 8: AP Sign Service trả về file PDF đã ký cho App Service
- Bước 9: App Service thông báo kết quả, App hiển thị kết quả ký số cho người dùng.

3. Danh sách tệp

TT	Nội dung	Mô tả
1	update-link-mssp-service.txt	Link dịch vụ MSSP mới nhất
2	MSSP-Tai lieu dac ta tích hợp AE-VNP.pdf	Đặc tả API của MSSP Webservice và mã lỗi trả về
3	MSS_SignaturePort.wsdl	File cấu trúc Webservice, có thể sử dụng để các nền tảng ngôn ngữ lập trình tự động tạo Proxy class tích hợp vào ứng dụng
4	vgca.mssp.jar	Thư viện viết trên Java hỗ trợ các API thực hiện ký số với SIM-PKI và lấy chứng thư số tương ứng với số điện thoại của thuê bao. Thư viện có thể tích hợp trực tiếp triển ứng dụng Java
5	samplemssp.zip	Project demo tích hợp thư viện vgca.mssp.jar
6	mssp-sample.zip	Project demo tích hợp dịch vụ MSSP trên ngôn ngữ C# dotnet

4. Điều kiện tích hợp dịch vụ MSSP

- Các ứng dụng phát triển trên nền tảng dotnet thì cần sử dụng dotnet 4.5 trở lên (do webservice MSSP sử dụng giao thức TLS 1.2 Client Authentication)
- Các ứng dụng phát triển trên nền tảng Java thì cần JDK 1.8 trở lên (do webservice MSSP sử dụng giao thức TLS 1.2 Client Authentication)
- Cơ quan đơn vị gửi văn bản yêu cầu cấp chứng thư số cho máy chủ Application Provider (AP) đến Cục Chứng thực số và Bảo mật thông tin để có Chứng thư số Client phục vụ kết nối dịch vụ MSSP

- Liên hệ với bộ phận hỗ trợ kỹ thuật của Cục CTSBMTT và cung cấp địa chỉ IP Public của AP để được cấp tài khoản (APID/Password) kết nối hệ thống MSSP

II. Hướng dẫn triển khai tích hợp

1. Hướng dẫn tích hợp trên môi trường dotnet

- Sử dụng Visual Studio, tạo proxy class từ file MSS_SignaturePort.wsdl
- + Mở Developer Command Prompt for VS
- + Chuyển thư mục lưu file kết quả (Không bắt buộc)
- + Chạy lệnh: wsdl + url (trong đó url: địa chỉ file wsdl của MSSP WS)
- Hoặc có thể sử dụng file “MSS_SignaturePort.cs” có sẵn trong mssp_sample.zip
- Nhúng class MSSP_API.cs là code ví dụ để gọi các API của MSSP Webservice vào trong Project.
- Để thực hiện, lấy thông tin chứng thư số tương ứng với số điện thoại thì thực hiện như sau:

Bước 1: Khởi tạo đối tượng lớp MSSP_API

```
MSSP_API msspAPI = new MSSP_API(apid, appwd, apCertPath, apCertPwd);
```

Trong đó:

apid và appwd: là mã id và mật khẩu truy cập hệ thống MSSP Webservice được Cục CTSBMTT cấp.

apCertPath/apCertPwd: là đường dẫn file CTS PKCS#12 và mật khẩu của chứng thư số AP do Cục CTSBMTT cấp.

Bước 2: Gọi API getCert để nhận về nội dung chứng thư số

```
try
{
    byte[] certRaw = msspAPI.GetCert(phone);
}
catch (Exception ex)
{
    //xử lý lỗi
}
```

Bước 3: Gán nội dung chứng thư số vào đối tượng X509Certificate2 để sử dụng trong quá trình đóng gói chữ ký số theo các định dạng dữ liệu.

```
X509Certificate2 clientCert = new X509Certificate2(certRaw);
```

- Để ký số RSA hoặc ECDSA, thực hiện các bước sau:

Bước 1: Khởi tạo đối tượng lớp MSSP_API

```
MSSP_API msspAPI = new MSSP_API(apid, appwd, apCertPath, apCertPwd);
```

Trong đó:

apid và appwd: là mã id và mật khẩu truy cập hệ thống MSSP Webservice được Cục CTSBMTT cấp.

apCertPath/apCertPwd: là đường dẫn file CTS PKCS#12 và mật khẩu của chứng thư số AP do Cục CTSBMTT cấp.

Bước 2: Gọi API getCert để nhận về nội dung chứng thư số

```
try
{
    byte[] certRaw = msspAPI.GetCert(phone);
}
catch (Exception ex)
{
    //xử lý lỗi
}
```

Bước 3: Gán nội dung chứng thư số vào đối tượng X509Certificate2 để sử dụng trong quá trình đóng gói chữ ký số theo các định dạng dữ liệu.

```
X509Certificate2 clientCert = new X509Certificate2(certRaw);
```

Bước 4: Thực hiện băm dữ liệu và đóng gói giá trị băm với định dạng DigestInfo

```
try
{
    DerObjectIdentifier identify = new
DerObjectIdentifier(DigestAlgorithms.GetAllowedDigests(hashAlgorithm));
    AlgorithmIdentifier algId = new AlgorithmIdentifier(identify,
DerNull.Instance);
    IDigest digest = DigestUtilities.GetDigest(identify);
    hash = new byte[digest.GetDigestSize()];
    digest.BlockUpdate(message, 0, message.Length);
    digest.DoFinal(hash, 0);
    DigestInfo dInfo = new DigestInfo(algId, hash);
    hashedstr = dInfo.GetDerEncoded();
}
catch(Exception ex)
{
    //Xử lý lỗi
}
```

Bước 5: Gọi API ký số giá trị DigestInfo

```
try
{
    byte[] sigBuff = api.Sign(phone, messageToBeDisplayed, hashedstr);
    Log.Info("Ký số thành công.");
}
catch (Exception ex)
{
    throw new SignatureException("Ký số qua MSSP không thành công: " +
ex.Message, ex);
}
```

2. Hướng dẫn tích hợp trên môi trường Java

Cơ quan đơn vị có thể thực hiện tích hợp thư viện vgca.mssp.jar vào phần mềm ứng dụng, hoặc tích hợp trực tiếp webservice MSSP theo đường dẫn đã nêu ở mục 1, với WSDL file “MSS_SignaturePort.wsdl”, mã nguồn mẫu trong thư mục java/sample-code.

Trường hợp tích hợp thư viện “vgca.mssp.jar” thì thực hiện theo các bước sau:

- Bước 1: Thêm thư viện vgca.mssp.jar và các thư viện liên quan vào Java Project.
- Bước 2: Thiết lập các chứng thư số tin cậy vào TrustStore để phục vụ kết nối

```
String TRUSTSTORE_FILE = "etc/truststore-bcy.jks";  
String TRUSTSTORE_PASSWORD = "changeit";  
String KEYSTORE_FILE =  
"etc/3108937_VGCA_Application_Provider_certificate.p12";  
String KEYSTORE_PASSWORD = "123456";  
String KEYSTORE_TYPE = "PKCS12";
```

Trong đó, tệp TRUSTSTORE_FILE bao gồm các chứng thư số của CA. Tệp KEYSTORE_FILE là chứng thư số AP được Cục CTSBMTT cấp.

Thiết lập cấu hình kết nối SSL:

```
JvmSsl.setSSL(TRUSTSTORE_FILE,  
              TRUSTSTORE_PASSWORD,  
              KEYSTORE_FILE, KEYSTORE_PASSWORD,  
              KEYSTORE_TYPE);
```

- Bước 3: Khởi tạo đối tượng ExtSignature:

```
ExtSignature signer = new ExtSignature();
```

- Bước 4: Set các thông tin APID và Mật khẩu để truy nhập MSSP Webservice:

```
signer.setAPID(AP_ID);  
signer.setAPPWD(AP_PASSWORD);
```

- Bước 5: Lấy thông tin chứng thư số thuê bao để phục vụ các bước đóng gói chữ ký số vào các định dạng dữ liệu.

```
byte[] certBytes = signer.GetCert(msisdn);
```

Trong đó, msisdn là số điện thoại tương ứng với SIM-PKI của thuê bao.

Ghi nội dung chứng thư số vào đối tượng X509Certificate:

```
CertificateFactory factory = CertificateFactory.getInstance("X.509");
```



```
X509Certificate cert = (X509Certificate)factory.generateCertificate(new
ByteArrayInputStream(Base64.decode(certBytes)));
```

- Bước 6: Thực hiện băm dữ liệu và đóng gói thành DigestInfo:

```
Security.addProvider(new BouncyCastleProvider());

String plainString = "1234567890";

byte[] messageBuff = plainString.getBytes();

MessageDigest messageDigest = MessageDigest.getInstance("SHA-256");

byte[] hashData = messageDigest.digest(messageBuff);
```

- Bước 7: Thực hiện ký số dữ liệu:

```
byte[] signature = signer.sign(hashData, msisd, dataToBeDisplayed);
```

Trong đó, hashData là nội dung dữ liệu băm được đóng gói dạng DigestInfo; msisd là số điện thoại của thuê bao; dataToBeDisplayed là thông báo hiển thị trên điện thoại của thuê bao để xác nhận giao dịch ký số.

3. Danh mục mã lỗi (Exception code)

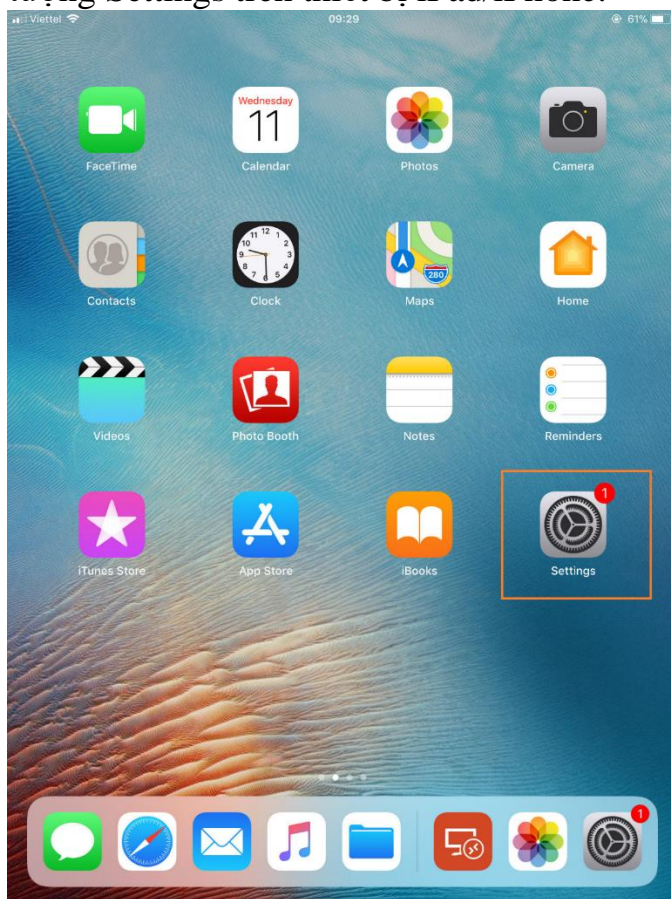
TT	Mã lỗi	Thông tin
1	101	WRONG_PARAM
2	102	MISSING_PARAM
3	103	WRONG_DATA_LENGTH
4	104	UNAUTHORIZED_ACCESS
5	105	UNKNOWN_CLIENT
6	106	HANDSHAKE_REQUIRED
7	107	INAPPROPRIATE_DATA
8	108	INCOMPATIBLE_INTERFACE
9	109	UNSUPPORTED_PROFILE
10	208	EXPIRED_TRANSACTION
11	209	OTA_ERROR
12	401	USER_CANCEL
13	402	PIN_NR_BLOCKED
14	403	CARD_BLOCKED
15	404	NO_KEY_FOUND
16	405	NO_URL_FOUND
17	406	PB_SIGNATURE_PROCESS
18	407	REGISTRATION_NOK
19	422	NO_CERT_FOUND
20	423	CRL_PB

III. Hướng dẫn một số trường hợp sử dụng SIM-PKI

1. Hướng dẫn đổi mã PIN của SIM-PKI

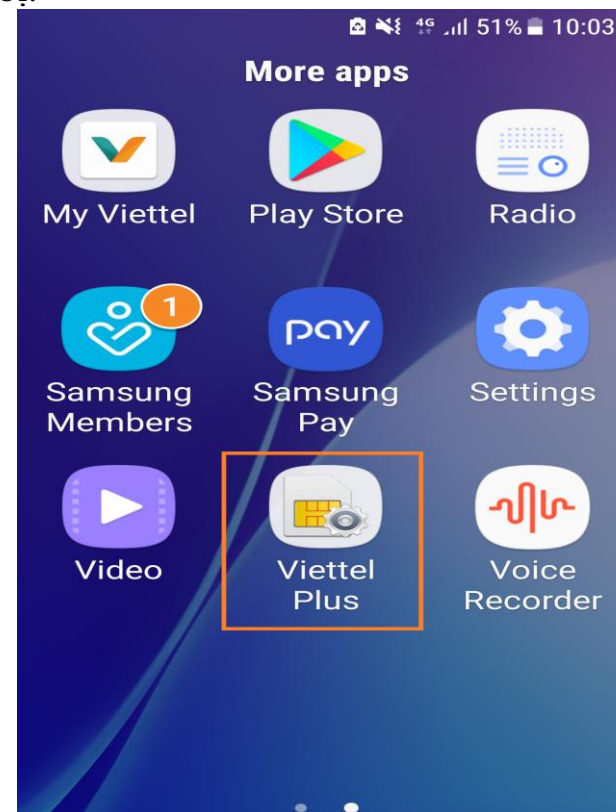
iOS

- **Bước 1:** Mở giao diện Settings trên iPad. Người dùng bấm vào biểu tượng Settings trên thiết bị iPad/iPhone:



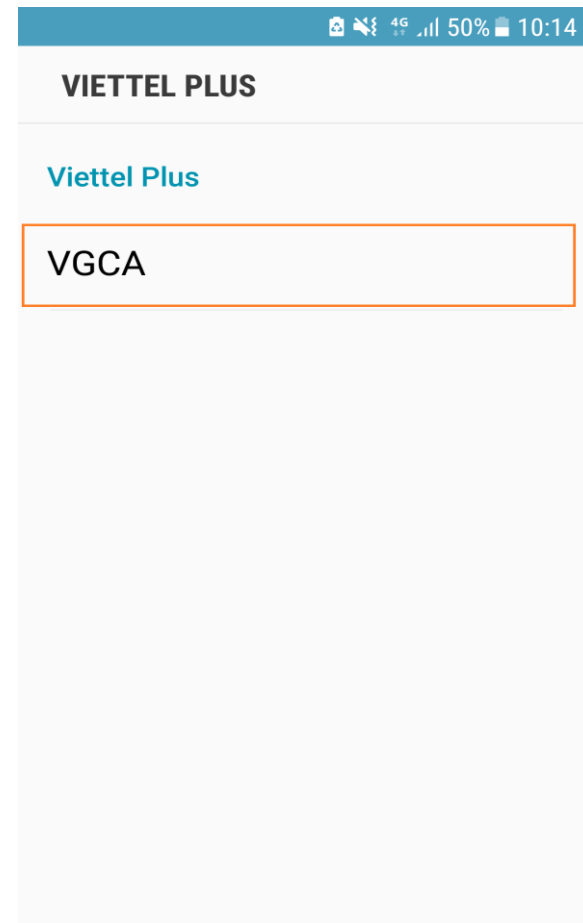
Android

- **Bước 1:** Mở giao diện SIM Settings trên Android. Người dùng bấm vào biểu tượng Viettel Plus hoặc Max SIM (Vinaphone) trên thiết bị:



- **Bước 2:** Mở menu Cellular Data Options. Trên cửa sổ Settings, người dùng bấm chọn menu Cellular Data, sau đó chọn Cellular Data Options:

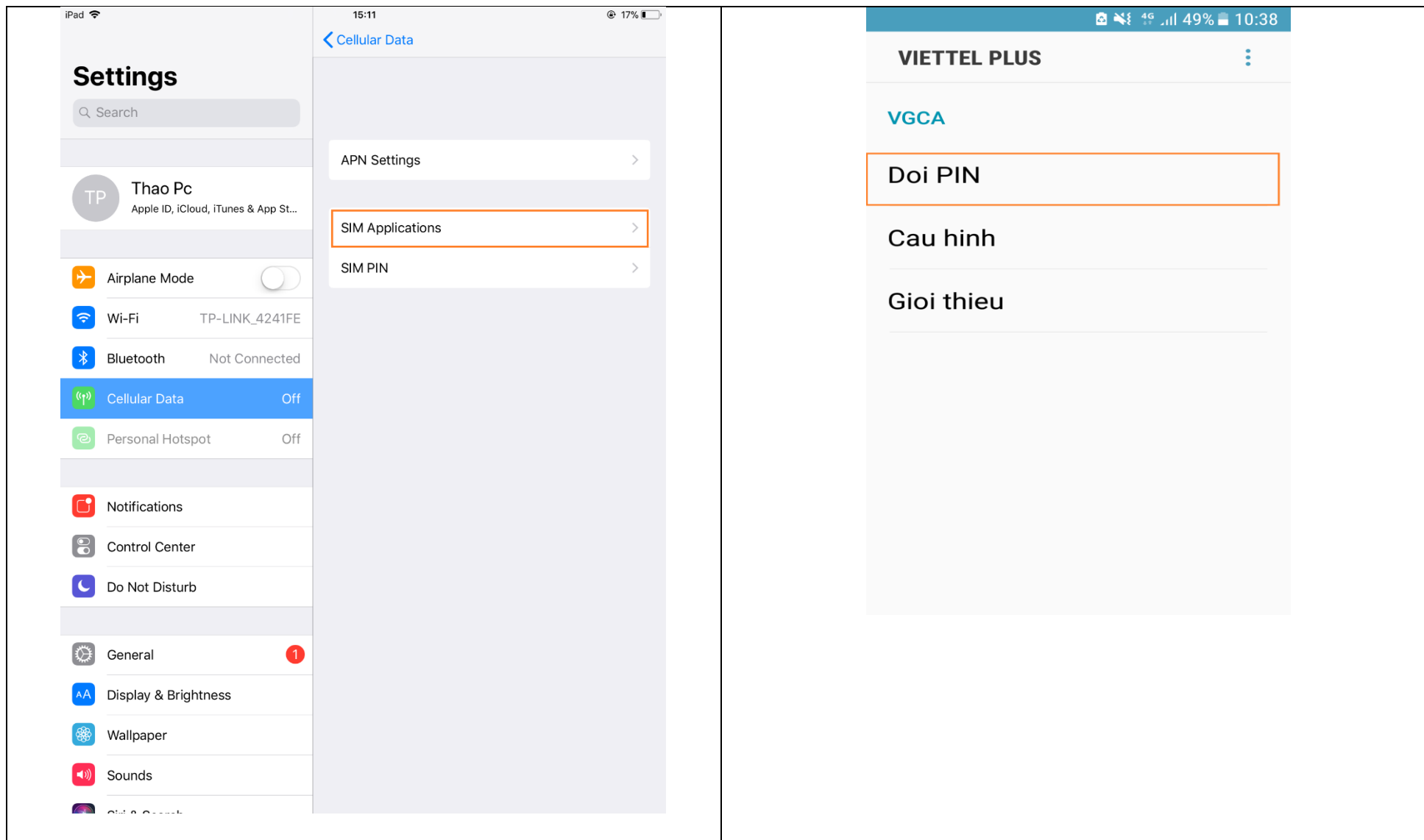
- **Bước 2:** Mở menu VGCA. Trên cửa sổ SIM Settings, người dùng bấm chọn menu VGCA.





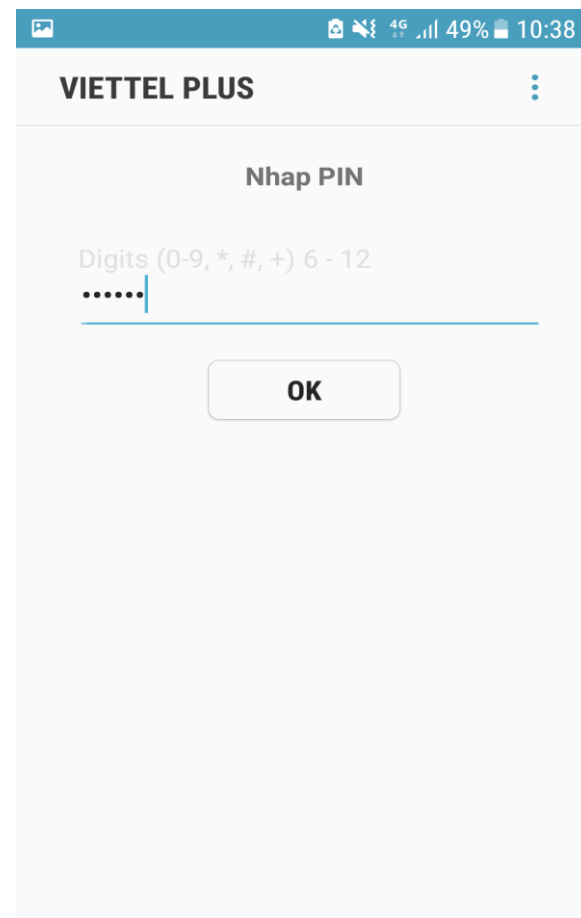
- **Bước 3:** Mở menu SIM Applications. Người dùng bấm chọn menu SIM Applications:

- **Bước 3:** Trên giao diện VGCA, người dùng chọn Dời PIN:

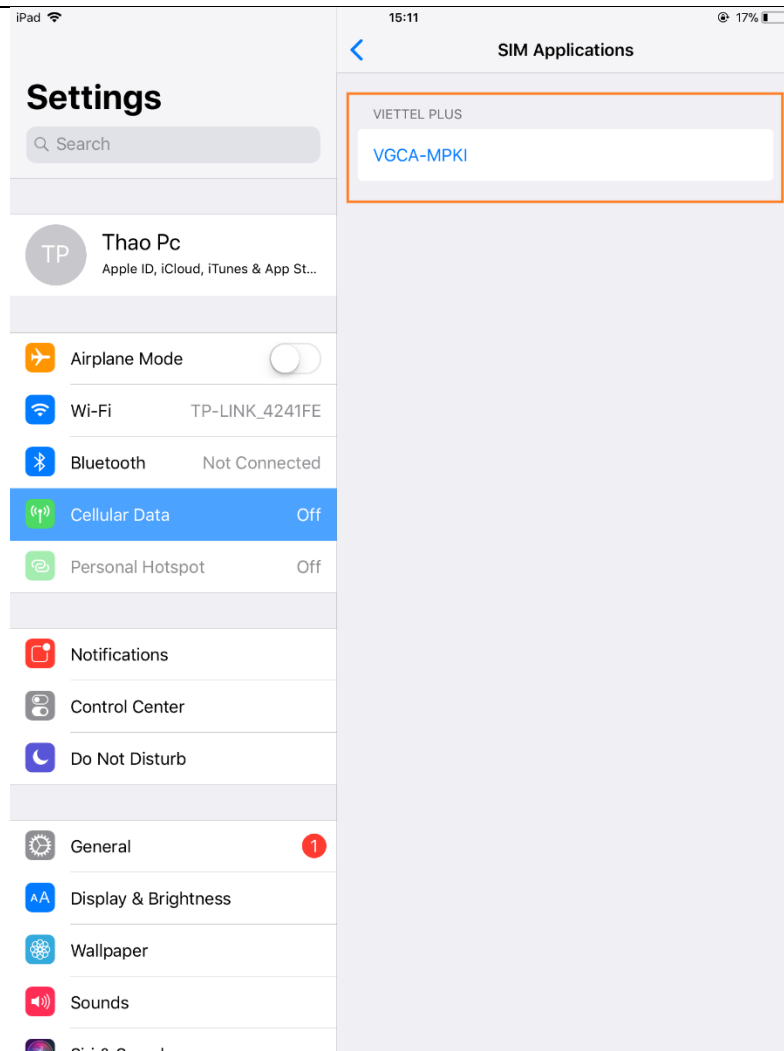


- **Bước 4:** Mở menu VGCA hoặc VGCA-MPKI. Trên cửa sổ SIM Applications, người dùng bấm chọn menu VGCA hoặc VGCA-MPKI.

- **Bước 4:** Người dùng nhập mã PIN đang sử dụng và bấm OK

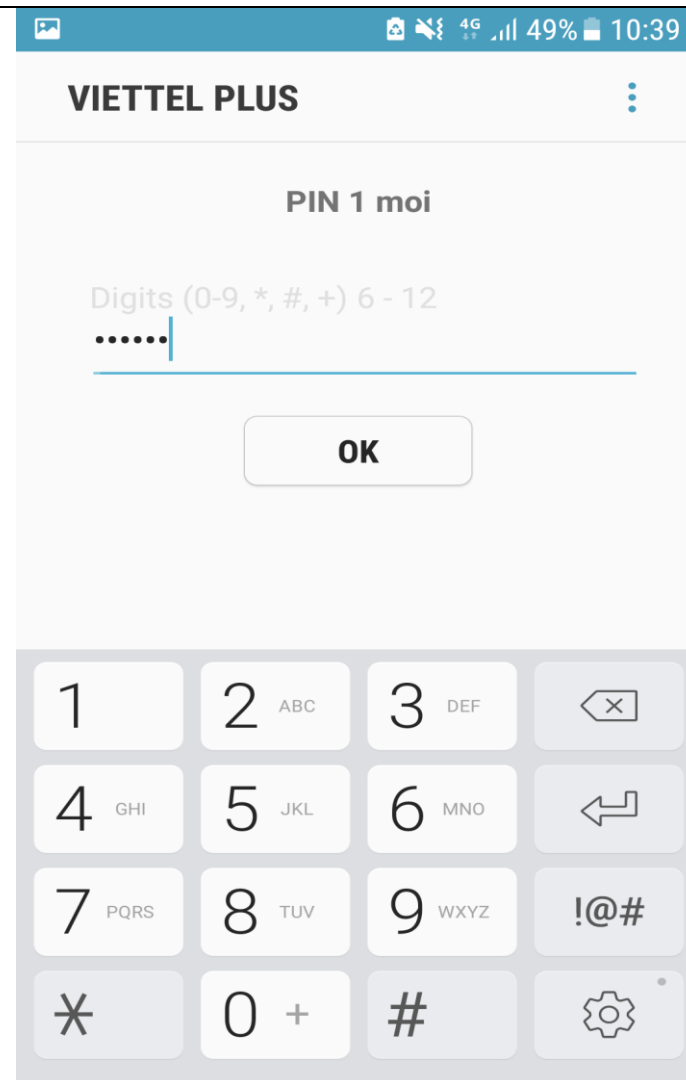
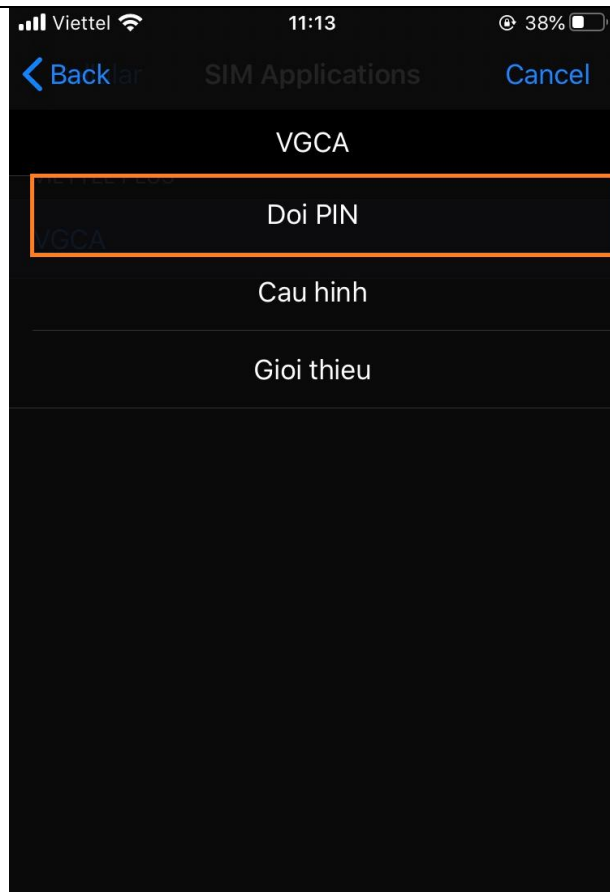


The image shows a mobile application interface for VIETTEL PLUS. At the top, there is a status bar with icons for signal, battery, and time (10:38). Below the status bar, the text "VIETTEL PLUS" is displayed in a bold, black font. To the right of this text is a vertical ellipsis menu icon. The main content area is titled "Nhập PIN" (Enter PIN) in a bold, black font. Below the title, there is a prompt "Digits (0-9, *, #, +) 6 - 12" in a smaller, gray font. Underneath the prompt is a text input field containing six dots, indicating a masked PIN. A blue cursor is positioned at the end of the input field. Below the input field is a large, rounded rectangular button with the text "OK" in a bold, black font.

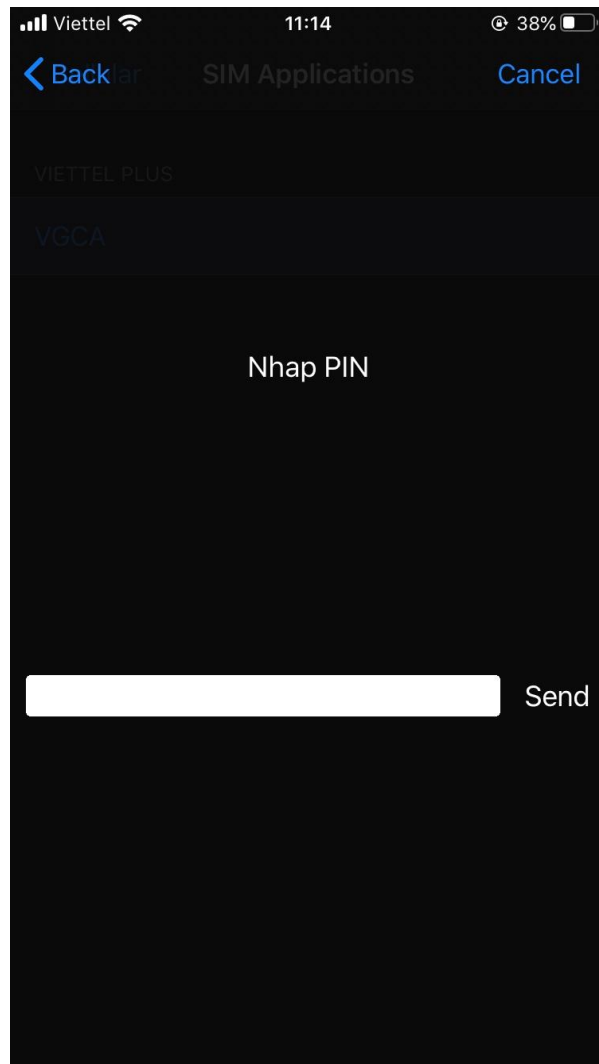


- **Bước 5:** Trên giao diện VGCA, người dùng chọn Dõi PIN:

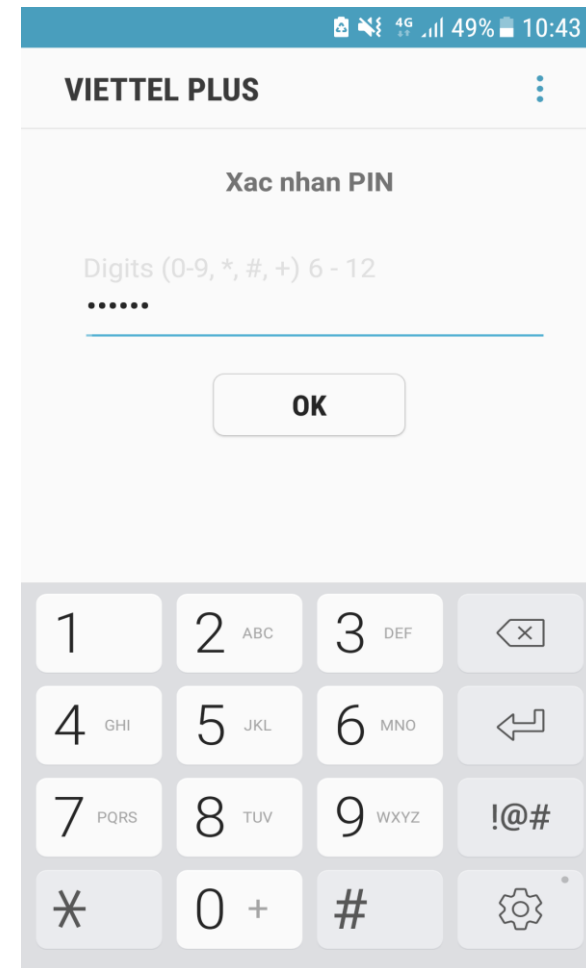
- **Bước 5:** Người dùng nhập PIN mới và bấm OK



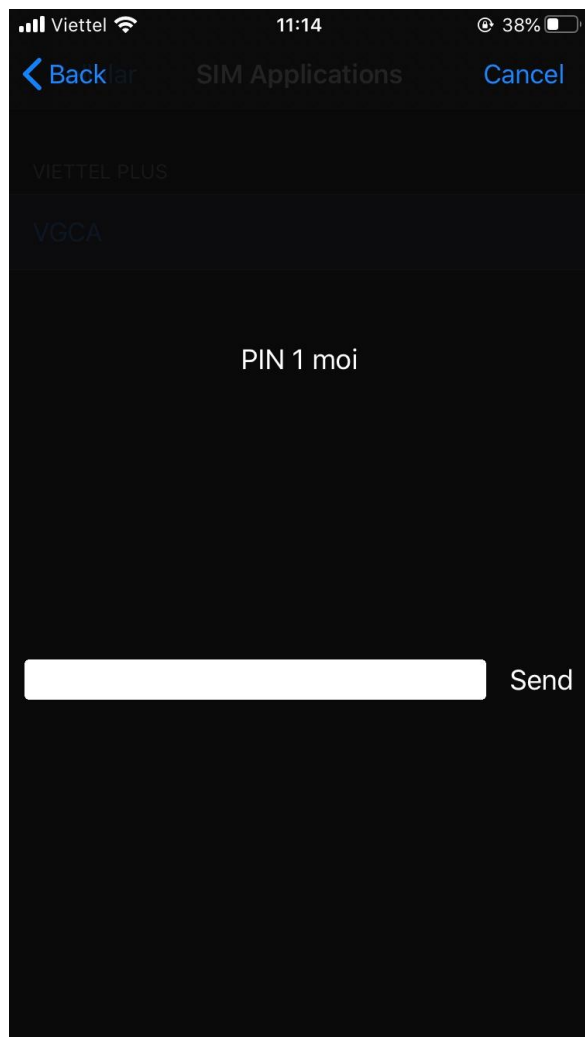
- **Bước 6:** Người dùng nhập mã PIN đang sử dụng và bấm **Send**



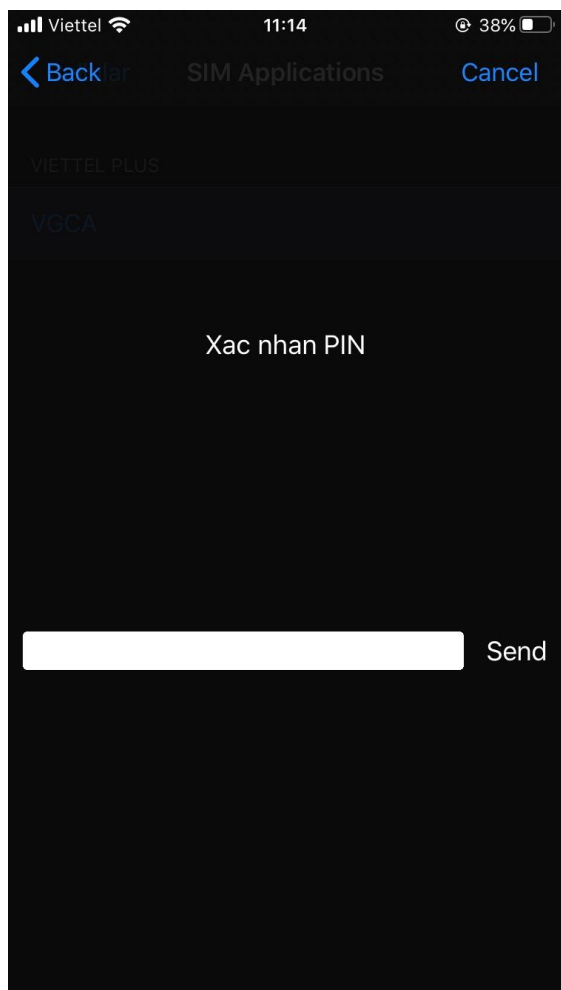
- **Bước 6:** Người dùng nhập lại mã PIN mới để xác nhận và bấm **OK**



- **Bước 7:** Người dùng nhập PIN mới và bấm **Send**

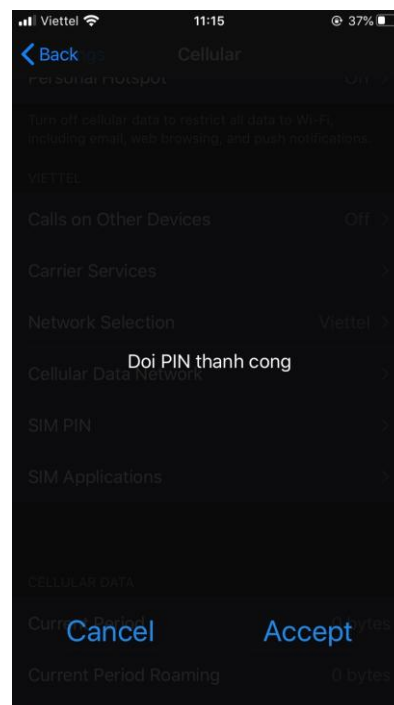


- Bước 8: Người dùng nhập lại mã PIN mới để xác nhận và bấm **Send**



The screenshot shows a mobile application interface on a dark background. At the top, the status bar displays 'Viettel', signal strength, Wi-Fi, time '11:14', and battery '38%'. Below the status bar is a navigation bar with a blue back arrow, the text 'SIM Applications', and a blue 'Cancel' button. The main content area lists two SIM cards: 'VIETTEL PLUS' and 'VGCA'. Below the list, the text 'Xác nhận PIN' (Confirm PIN) is centered. At the bottom, there is a white rectangular input field for the PIN and a 'Send' button to its right.

*** Thông báo đổi PIN thành công:**



Chú ý: Mã PIN của người dùng chỉ bao gồm ký tự số, có độ dài từ 6 đến 10 ký tự số.

2. Hướng dẫn bật, tắt chuông báo của SIM-PKI

Để bật, tắt chuông báo của SIM-PKI, người dùng mở giao diện cấu hình VGCA theo các bước 1 đến bước 5 hướng dẫn đổi mã PIN ở trên đối với thiết bị iOS và từ bước 1 đến bước 3 đối với thiết bị Android.

2.1. Các bước thực hiện bật/tắt chuông báo của SIM-PKI

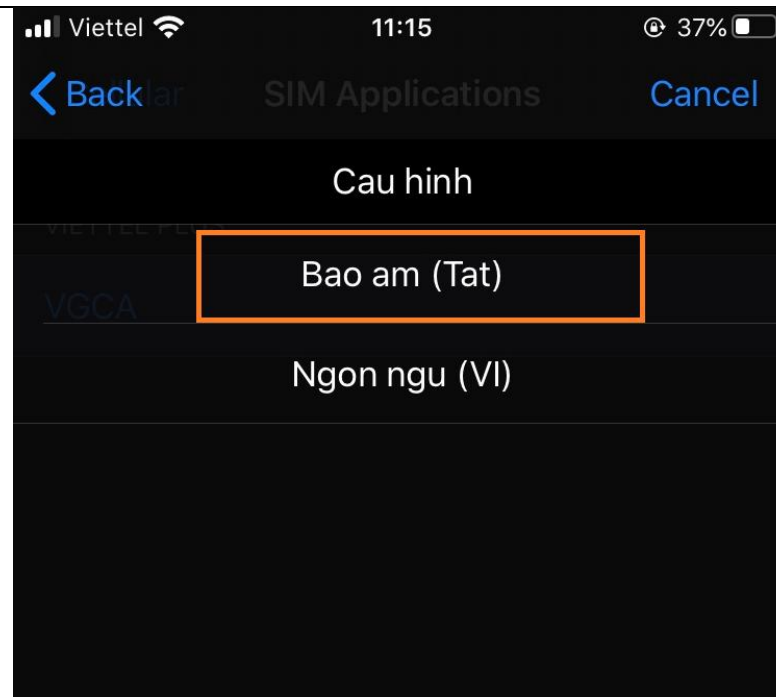
- **Bước 1:** Trên cửa sổ menu VGCA, người dùng chọn menu “Cau hình”:



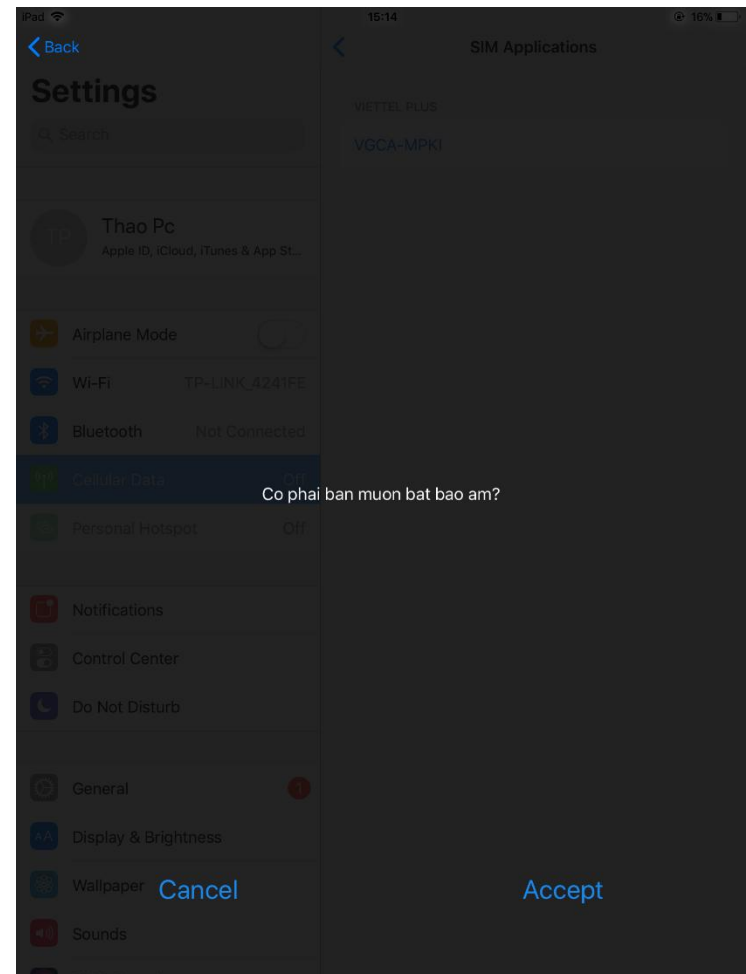
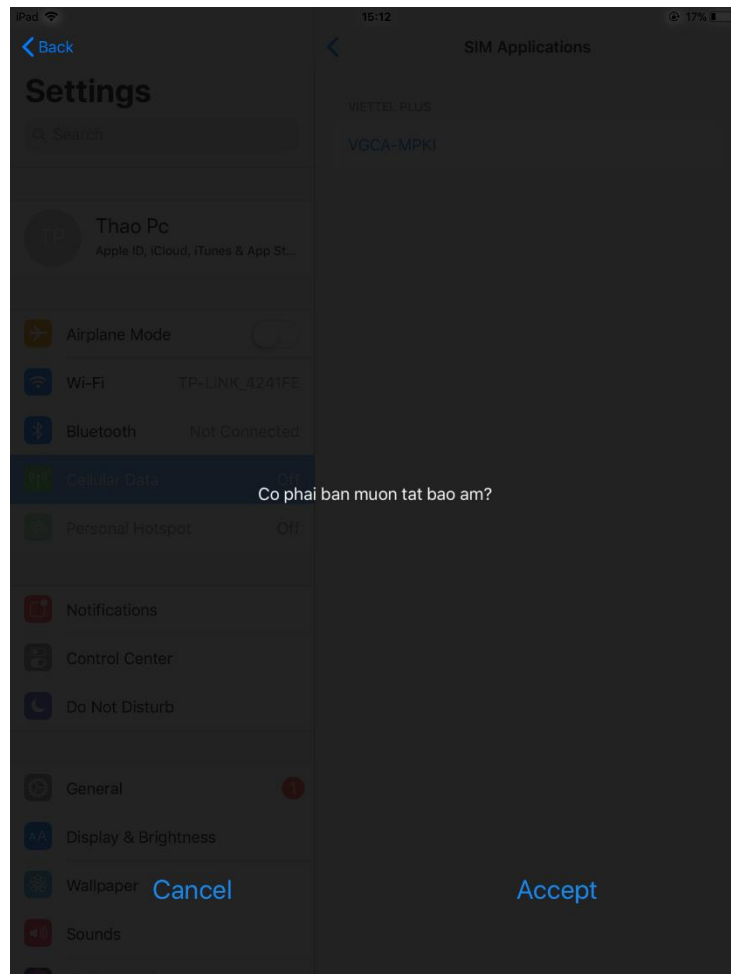
Tắt chuông báo	Bật chuông báo
- Bước 2: Mặc định chuông báo của SIM-PKI được tắt (Bao am (Tat)). Chọn menu Bao am (Bat) để Tắt chuông báo	- Bước 2: Người dùng Chọn menu Bao am (Tat) để Bật chuông báo:



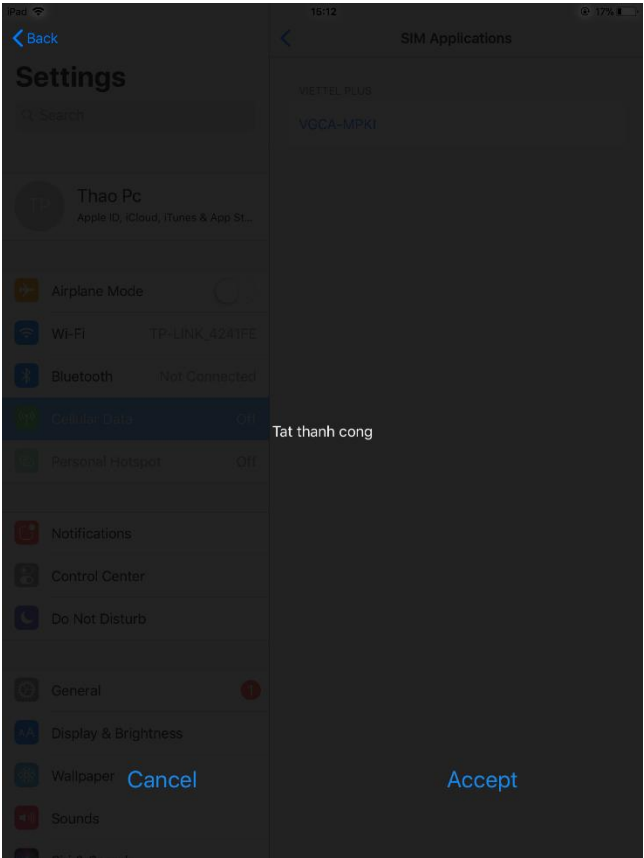
- **Bước 3:** Người dùng bấm chọn Accept trên thông báo “Có phải bạn muốn tắt bảo hiểm?”:



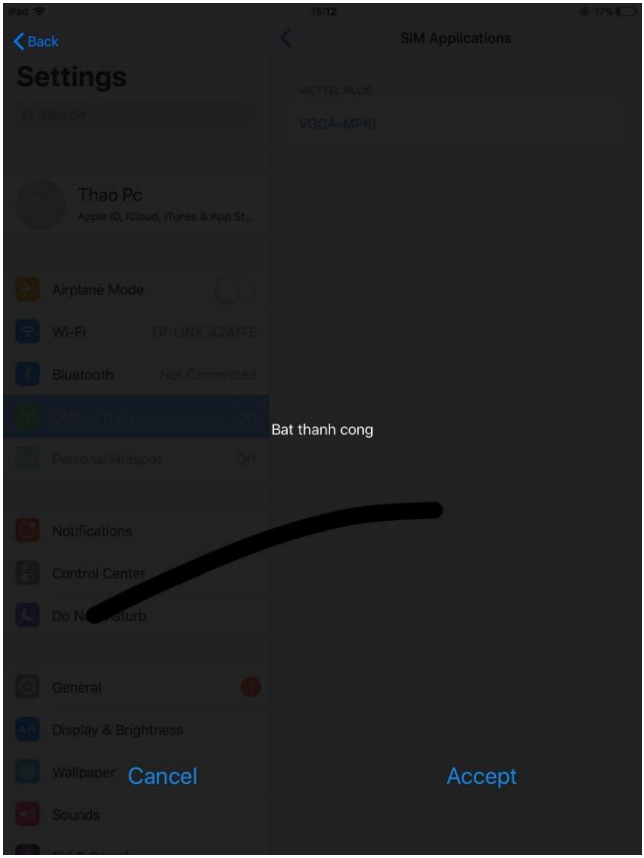
- **Bước 3:** Người dùng bấm chọn Accept trên thông báo “Có phải bạn muốn bật bảo hiểm?”:



- **Bước 4:** Tắt báo âm thành công sẽ có thông báo “Tat thanh cong”. Người dùng bấm chọn Accept để kết thúc:



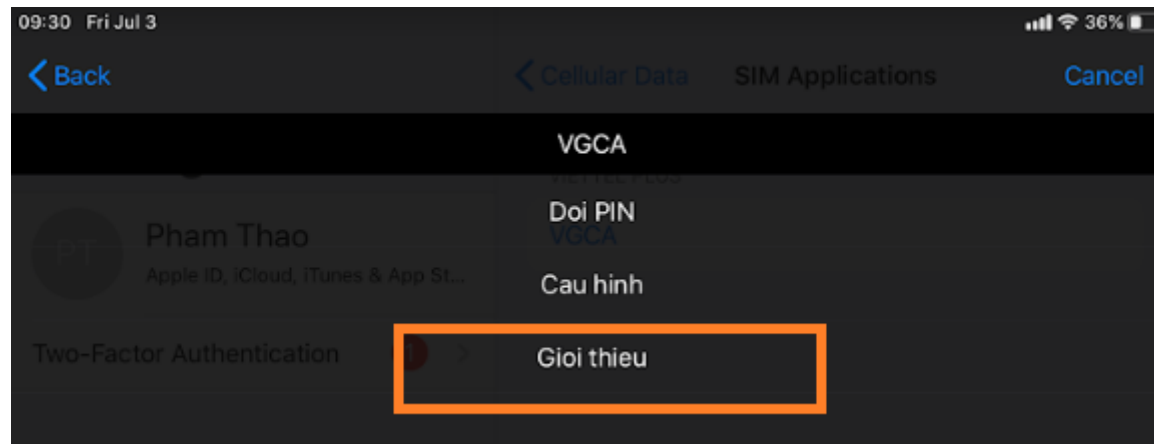
- **Bước 4:** Bật báo âm thành công sẽ có thông báo “Bat thanh cong”. Người dùng bấm chọn Accept để kết thúc:



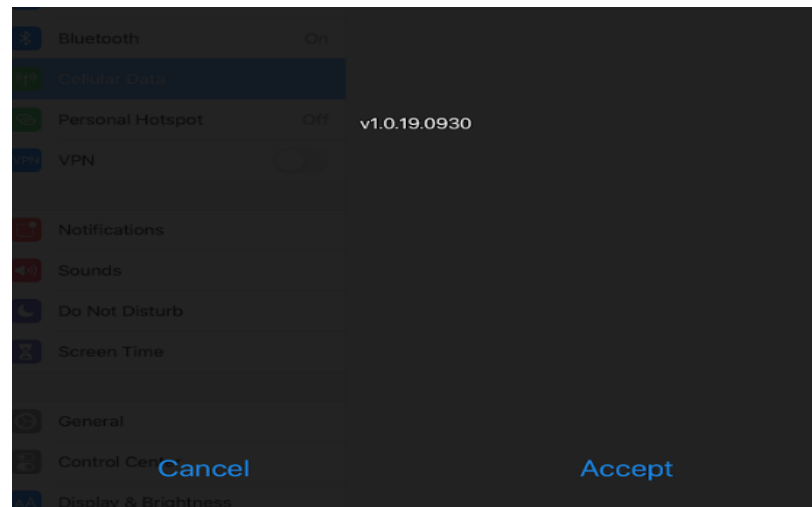
3. Hướng dẫn xem phiên bản SIM-PKI

- Phục vụ quá trình kiểm tra khắc phục sự cố đối với SIM-PKI trong quá trình ký số
- Các bước thực hiện:

Bước 1: Trên cửa sổ menu VGCA, người dùng chọn menu “Giới thiệu”:



Bước 2: Thông tin phiên bản SIM-PKI sẽ được hiển thị trên màn hình thiết bị di động.



Bước 3: Khi được yêu cầu, người dùng đọc mã phiên bản SIM-PKI cho cán bộ hỗ trợ kỹ thuật xử lý sự cố trong quá trình ký số SIM.