Aplikacja na platformę Android do obsługi jElib

Cel projektu

Aplikacja na platformę Android realizująca operacje odczytu i zapisu na aplikacji jElib.

Odczyt nie wymagający uwierzytelnienia

Zapis wymaga uwierzytelnienia z użyciem Global Platform

Obsługa NFC w Androidzie

W niektórych modelach telefonów

Wymaga podania uprawnień do manifestu

```
<uses-feature android:required="true"
    android:name="android.hardware.nfc"/>
    <uses-permission android:name="android.permission.NFC"/>
<intent-filter>
            <action
    android:name="android.nfc.action.TAG_DISCOVERED" />
 </intent-filter>
```

Inicjacja połączenia NFC w kodzie

Inicjacja aplikacji

mNfcAdapter = NfcAdapter.getDefaultAdapter(this);

W nadpisanej metodzie OnNewIntent

```
Tag tag = intent.getParcelableExtra(NfcAdapter.EXTRA_TAG);

    IsoDep iso = IsoDep.get(tag);
    //byte[] ats = iso.getHistoricalBytes();
    iso.connect();
    iso.setTimeout(1000);
```

Realizacja poleceń APDU

Polecenia możemy zapisywać w postaci tablicy bajtów.
Wykonanie poprzez metodę transcieve obiektu klasy IsoDep
Metoda zwraca odpowiedź karty
Przykład:

```
// Initialize update
    byte[] sec_res = iso.transceive(new byte[]{
            (byte) 0x80, // CLA Class
            (byte) 0x50, // INS Instruction
            (byte) 0x00, // P1  Parameter 1
            (byte) 0x00, // P2  Parameter 2
            (byte) 0x08, // Lc
            (byte) 0x06, (byte) 0x50, (byte) 0x04, (byte) 0x50,(byte)0x46,
     (byte)0x42,(byte)0x4E,(byte)0x4E,// Challenge (losowe bajty transmisji)
            (byte) 0x00 // Le
    });
```

Realizacja uwierzytelnienia hosta w Global Platform

Dwa polecenia APDU

- Initialize Update
- External Authenticate

Initialize Update

Wywołanie tego polecenia powoduje inicjację Secure Channel Session.

W poleceniu podajemy między innymi losowy Host Challenge.

Otrzymujemy informacje pomocne w dywersyfikacji klucza, informacje o kluczu, wartość losową z karty (card challenge) oraz kryptogram karty.

Otrzymane wartości pozwalają nam na obliczenie kryptogramu

External Authenticate

Służy do uwierzytelniania hosta i do zdeterminowania poziomu zabezpieczeń potrzebnego dla późniejszych komend.

W naszej wersji poziom zabezpieczeń wynosił 1.

Podajemy między innymi kryptogram hosta i MAC każdy po 8 bajtów, które łączymy ze sobą.

Otrzymujemy je z korzystając z algorytmu Full Triple DES. Które skracają ich długość do 8 bajtów.

Dodanie pozycji w jElib

ADD ENTRY (INS = 0x37)

Dodaje nową parę ASN.1 <klucz, wartość>
Wykorzystywane w plikach EF.CONFIG i EF.ID.

Dodanie pary o kluczu 0x212223 i wartości 0x010203.
C0 37 02 00 0A 51 03 21 22 23 53 03 01 02 03 00

Wymaga uwierzytelnienia Global Platform

Odczytanie ID użytkownika z określonej biblioteki

GET ENTRY VALUE (INS = 0x33)

Pobranie wartości parametru z pliku EF.ID dla klucza ’PFBN’ (0x5046424E).
C0 33 02 00 06 51 04 50 46 42 4E 00