2014

Code Engn SmartApp 3

4.Z320

elttzero@gmail.com

Challenges : SmartApp 03

Author : 보안프로젝트 / Link

Korean :
키값을 찾으시오!

English :
Find a key

Download

키값을 찾으라는 문제입니다.

~ ?

: 576349792

: 3

어플을 실행해 보면 위쪽에는 큰 숫자와 아래에는 0이 있으며 버튼을 누르면 그 숫자가 1씩 증가하는 것을 확인할 수 있습니다. 따라서 아래의 숫자가 위 숫자와 동일해지면 키값이 나옴을 추측할 수 있습니다.

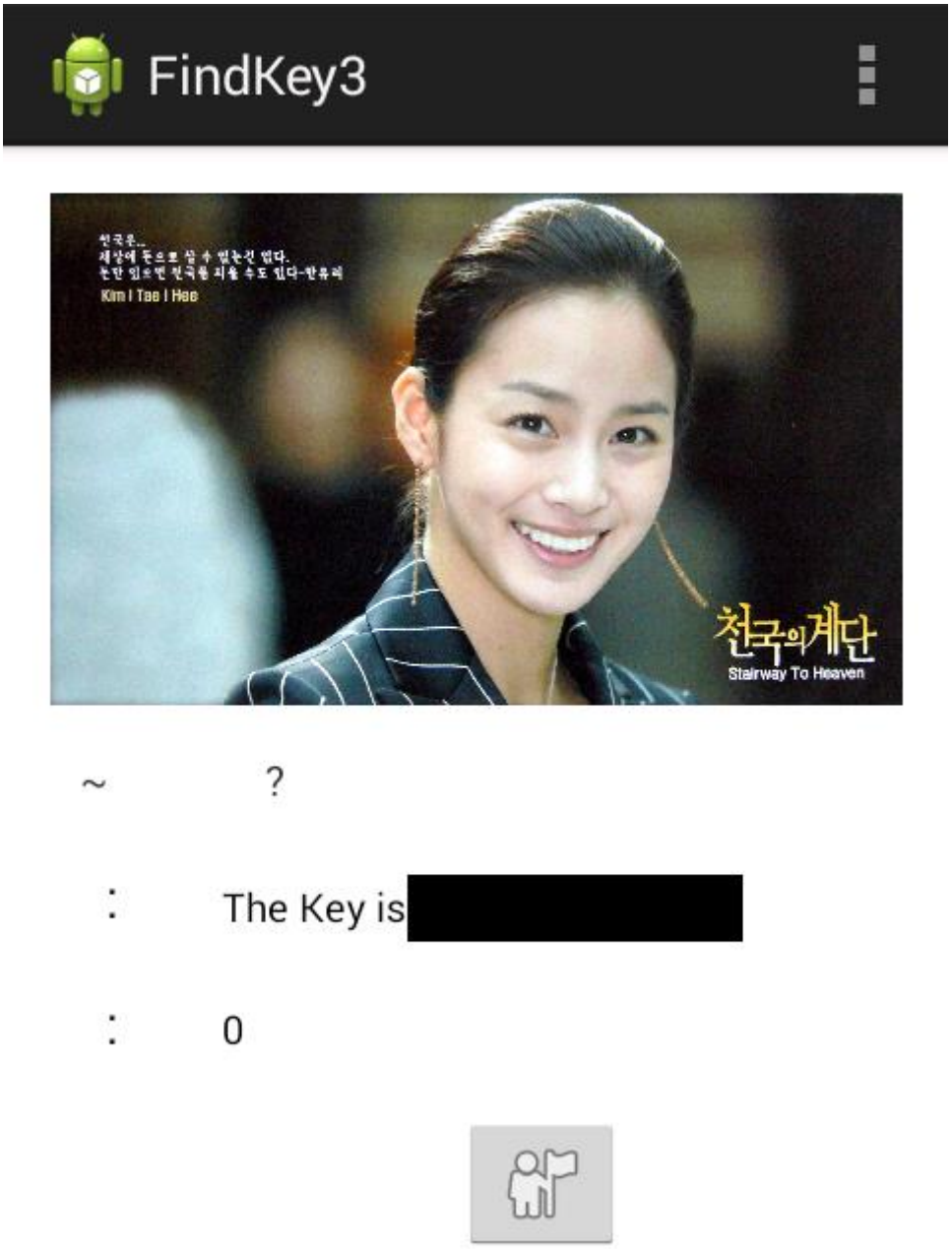
```
    arrayOfObject[0] = Integer.valueOf(randomRange(44444));
    localTextView.setText(String.format("%d", arrayOfObject));
    this.bView.setText("0");
    this.cView.setText(myString());
    this.button.setOnClickListener(new View.OnClickListener()
    {
      Integer myStairs = Integer.valueOf(Integer.parseInt(MainActivity.this.bView.getText()));
      Integer stairs = Integer.valueOf(Integer.parseInt(MainActivity.this.aView.getText()));

      public void onClick(View paramAnonymousView)
      {
        if (-1 + this.stairs.intValue() != this.myStairs.intValue())
        {
          this.myStairs = Integer.valueOf(1 + this.myStairs.intValue());
          MainActivity.this.bView.setText(this.myStairs.toString());
          return;
        }
        MainActivity.this.aView.setText(Security.DecryptStr("2736f6055dbad2d42f6d5b0135395cb29e(
        MainActivity.this.bView.setText("0");
      }
    });
  }
```

언팩 후 dex파일을 jar로 변환 후 소스코드를 확인해 보면 aView의 stairs와 bView의 myStairs의 값이 동일하면 키값이 복호화 후 출력됨을 알 수 있으며 bView의 값은 randomRange메서드를 통해 정해짐을 확인할 수 있으며 randomRange메서드는 아래와 같습니다.

```
public int randomRange(int paramInt)
{
  return (int)(10000.0D * Math.random()) * (paramInt << 2);
}
```

```
const v4, 0xad9c
invoke-virtual {p0, v4}, Lcom/namdaehyeon/findkey3/MainActivity;->randomRange(I)I
move-result v4 ➡ const v4, 0x2
invoke-static {v4}, Ljava/lang/Integer;->valueOf(I)Ljava/lang/Integer;
```

smali코드에서 randomRange를 통해 arrayOfObject에 값을 세팅하는 부분을 찾아 항상 같은 값(2)을 세팅하도록 변경합니다.



이후 어플을 설치한 후 버튼으로 위에서 정한 값으로 변경해 주면 키값이 나오는 것을 확인할 수 있습니다.