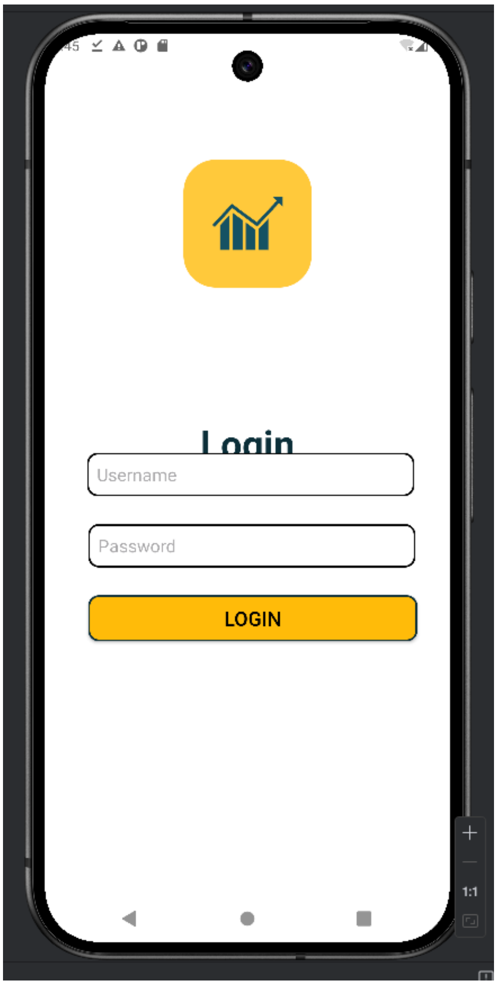
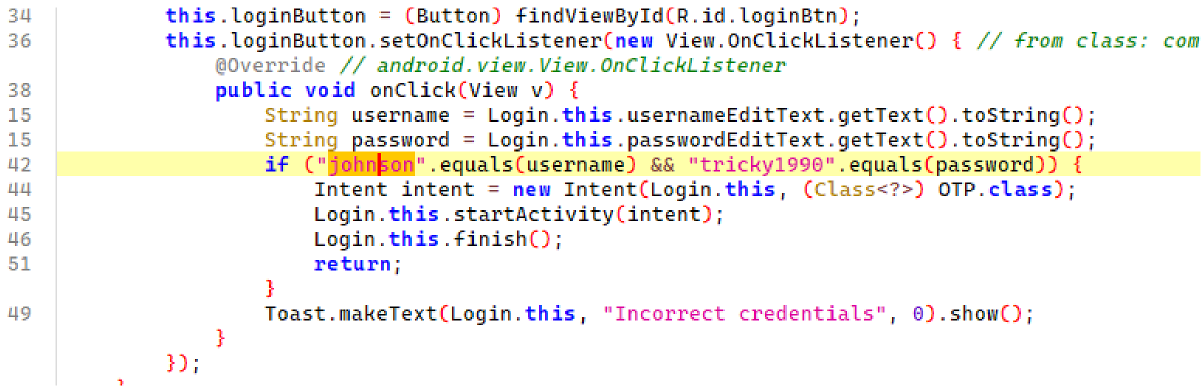
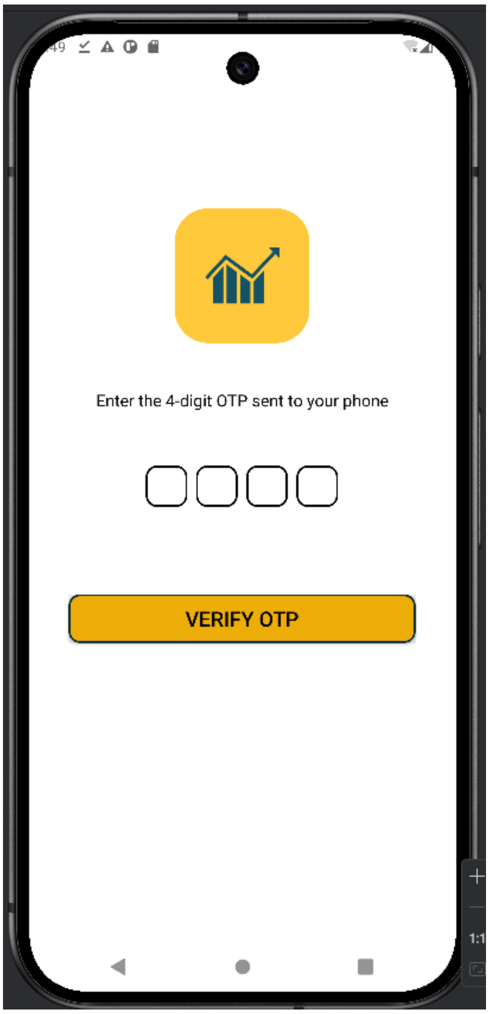


주어진 서버에서 pico bank apk 파일을 제공한다.



```
        this.loginButton = (Button) findViewById(R.id.loginBtn);
        this.loginButton.setOnClickListener(new View.OnClickListener() { // from class: com
            @Override // android.view.View.OnClickListener
            public void onClick(View v) {
                String username = Login.this.usernameEditText.getText().toString();
                String password = Login.this.passwordEditText.getText().toString();
                if ("johnson".equals(username) && "tricky1990".equals(password)) {
                    Intent intent = new Intent(Login.this, (Class<?>) OTP.class);
                    Login.this.startActivity(intent);
                    Login.this.finish();
                    return;
                }
                Toast.makeText(Login.this, "Incorrect credentials", 0).show();
            }
        });
```

jadx에서 Login 관련 코드를 보면 계정이 하드코딩 되어있는걸 볼 수 있다.



Enter the 4-digit OTP sent to your phone

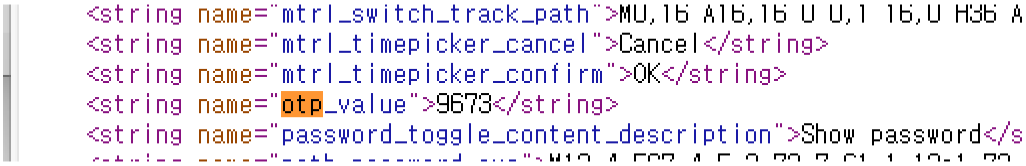
□□□□

해당 계정으로 로그인을 하면 OTP입력을 해야한다.

```
/* JADX INFO: Access modifiers changed from: private */
    public void verifyOtp(String otp) {
        String endpoint = "your server url/verify-otp";
        if
(getResources().getString(R.string.otp_value).equals(otp)) {
            Intent intent = new Intent(this, (Class<?>)
MainActivity.class);
            startActivity(intent);
            finish();
        } else {
            Toast.makeText(this, "Invalid OTP", 0).show();
        }
```

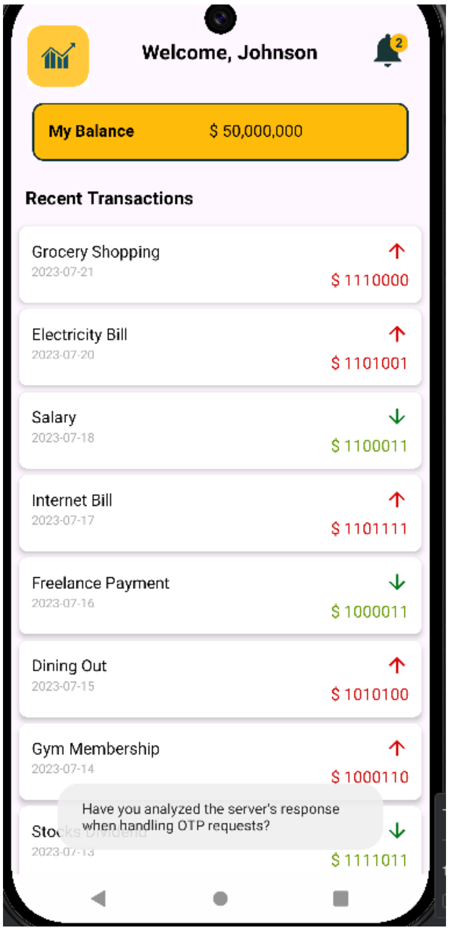
로컬에서 값을 가져와서 otp 값을 비교하는 것 같다.

apktool d pico-bank.apk -o out



```
<string name="mtrl_switch_track_path">M0,16 A16,16 0 0,1 16,0 H36 A
<string name="mtrl_timepicker_cancel">Cancel</string>
<string name="mtrl_timepicker_confirm">OK</string>
<string name="otp_value">9673</string>
<string name="password_toggle_content_description">Show password</s
```

apk 파일 압축을 풀고, res/values/strings.xml 파일을 보니 otp 가 하드코딩 되어있었다.



Welcome, Johnson

My Balance $ 50,000,000

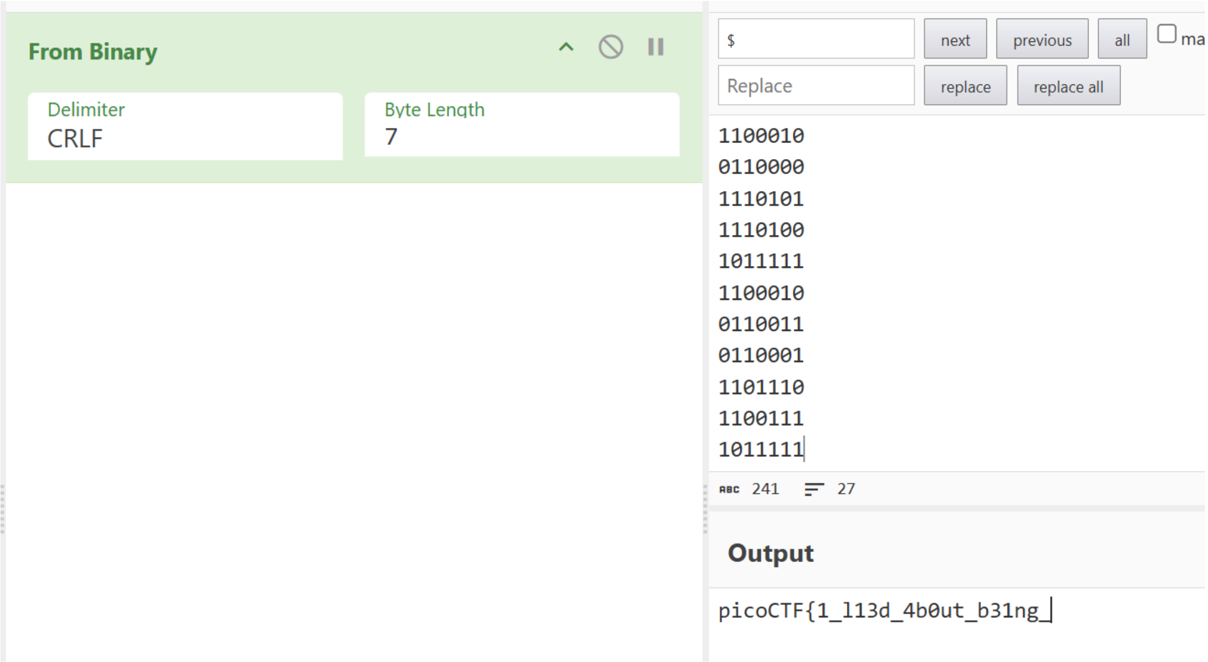
Recent Transactions

Grocery Shopping 2023-07-21	$ 1110000
Electricity Bill 2023-07-20	$ 1101001
Salary 2023-07-18	$ 1100011
Internet Bill 2023-07-17	$ 1101111
Freelance Payment 2023-07-16	$ 1000011
Dining Out 2023-07-15	$ 1010100
Gym Membership 2023-07-14	$ 1000110
Stoc 2023-07-13	$ 1111011

Have you analyzed the server's response when handling OTP requests?

OTP 코드를 입력하고 로그인을 하면 사람들별로 거래한 흔적이 보인다.

1과 0으로 이루어진 것을 봐서 이진수인 것 같다.



Delimiter CRLF

Byte Length 7

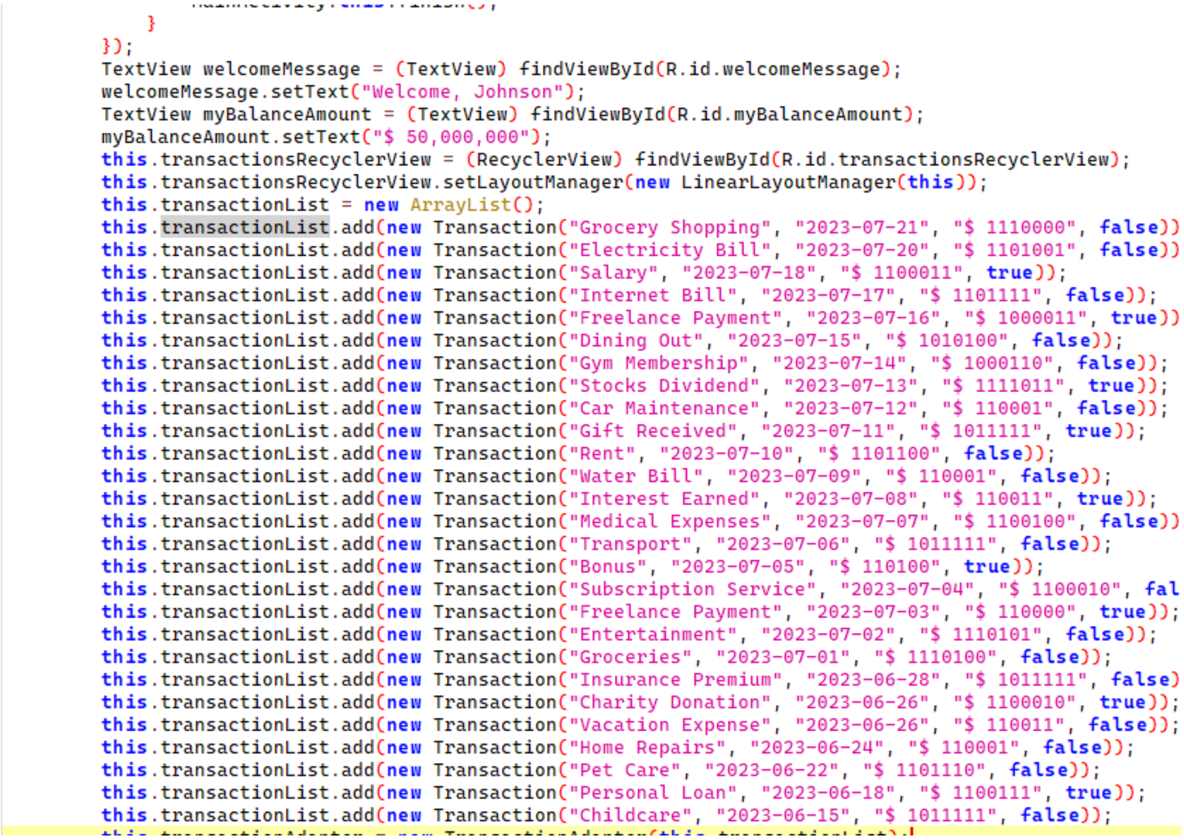
$

```
1100010
0110000
1110101
1110100
1011111
1100010
0110011
0110001
1101110
1100111
1011111
```

ABC 241 ☰ 27

```
picoCTF{1_l13d_4b0ut_b31ng_
```

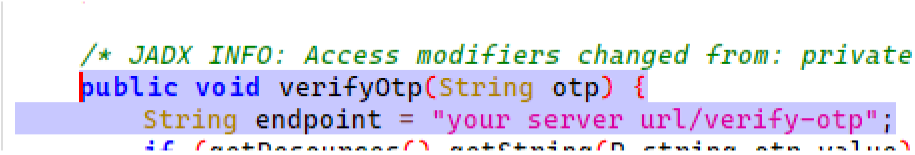
해당 값을 문자로 변환하면 FLAG의 일부가 나온다.



```
                MainActivity.this.finish();
            }
        });
        TextView welcomeMessage = (TextView) findViewById(R.id.welcomeMessage);
        welcomeMessage.setText("Welcome, Johnson");
        TextView myBalanceAmount = (TextView) findViewById(R.id.myBalanceAmount);
        myBalanceAmount.setText("$ 50,000,000");
        this.transactionsRecyclerView = (RecyclerView) findViewById(R.id.transactionsRecyclerView);
        this.transactionsRecyclerView.setLayoutManager(new LinearLayoutManager(this));
        this.transactionList = new ArrayList();
        this.transactionList.add(new Transaction("Grocery Shopping", "2023-07-21", "$ 1110000", false))
        this.transactionList.add(new Transaction("Electricity Bill", "2023-07-20", "$ 1101001", false))
        this.transactionList.add(new Transaction("Salary", "2023-07-18", "$ 1100011", true));
        this.transactionList.add(new Transaction("Internet Bill", "2023-07-17", "$ 1101111", false));
        this.transactionList.add(new Transaction("Freelance Payment", "2023-07-16", "$ 1000011", true))
        this.transactionList.add(new Transaction("Dining Out", "2023-07-15", "$ 1010100", false));
        this.transactionList.add(new Transaction("Gym Membership", "2023-07-14", "$ 1000110", false));
        this.transactionList.add(new Transaction("Stocks Dividend", "2023-07-13", "$ 1111011", true));
        this.transactionList.add(new Transaction("Car Maintenance", "2023-07-12", "$ 110001", false));
        this.transactionList.add(new Transaction("Gift Received", "2023-07-11", "$ 1011111", true));
        this.transactionList.add(new Transaction("Rent", "2023-07-10", "$ 1101100", false));
        this.transactionList.add(new Transaction("Water Bill", "2023-07-09", "$ 110001", false));
        this.transactionList.add(new Transaction("Interest Earned", "2023-07-08", "$ 110011", true));
        this.transactionList.add(new Transaction("Medical Expenses", "2023-07-07", "$ 1100100", false))
        this.transactionList.add(new Transaction("Transport", "2023-07-06", "$ 1011111", false));
        this.transactionList.add(new Transaction("Bonus", "2023-07-05", "$ 110100", true));
        this.transactionList.add(new Transaction("Subscription Service", "2023-07-04", "$ 1100010", fal
        this.transactionList.add(new Transaction("Freelance Payment", "2023-07-03", "$ 110000", true));
        this.transactionList.add(new Transaction("Entertainment", "2023-07-02", "$ 1110101", false));
        this.transactionList.add(new Transaction("Groceries", "2023-07-01", "$ 1110100", false));
        this.transactionList.add(new Transaction("Insurance Premium", "2023-06-28", "$ 1011111", false)
        this.transactionList.add(new Transaction("Charity Donation", "2023-06-26", "$ 1100010", true));
        this.transactionList.add(new Transaction("Vacation Expense", "2023-06-26", "$ 110011", false));
        this.transactionList.add(new Transaction("Home Repairs", "2023-06-24", "$ 110001", false));
        this.transactionList.add(new Transaction("Pet Care", "2023-06-22", "$ 1101110", false));
        this.transactionList.add(new Transaction("Personal Loan", "2023-06-18", "$ 1100111", true));
        this.transactionList.add(new Transaction("Childcare", "2023-06-15", "$ 1011111", false));
        this.transactionAdapter = new TransactionAdapter(this.transactionList);
```

해당 값은 MainActivity에 하드코딩 되어있다.

그 다음 나머지 flag도 찾아야하는데, 힌트를 보니 OTP를 입력하고 네트워크 응답 값을 참조하라고 한다.

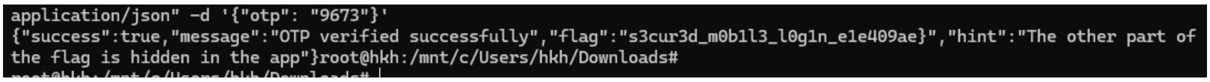


```
    /* JADX INFO: Access modifiers changed from: private */
    public void verifyOtp(String otp) {
        String endpoint = "your server url/verify-otp";
        if (getResources().getString(R.string.otp_value).equals(otp))
```

OTP 코드에 verify-otp 경로가 endpoint로 박혀있다. 그래서 서버를 열고 otp 값을 보내면 flag 를 얻을 수 있다.

curl -X POST http://amiable-citadel.picoctf.net:51759/verify-otp -H "Content-Type: application/json" -d '{"otp": "9673"}'

그래서 서버를 열고, 해당 서버값



```
application/json" -d '{"otp": "9673"}'
{"success":true,"message":"OTP verified successfully","flag":"s3cur3d_m0b1l3_l0g1n_e1e409ae}","hint":"The other part of
the flag is hidden in the app"}root@hkh:/mnt/c/Users/hkh/Downloads#
root@hkh:/mnt/c/Users/hkh/Downloads#
```