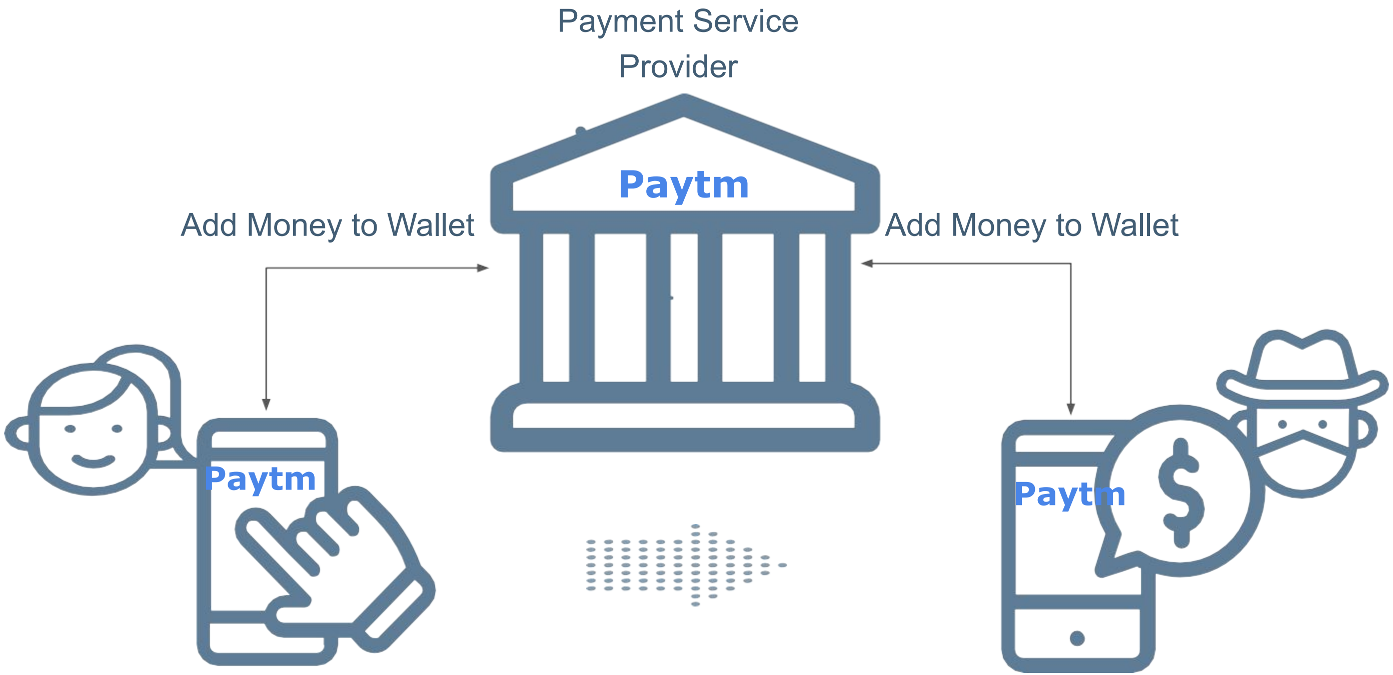
Security Analysis of Unified Payments Interface and Payment Apps in India

Jash Jain 2019130021
Kashish Jain 2019130022

Early Indian Payments Apps - Wallets



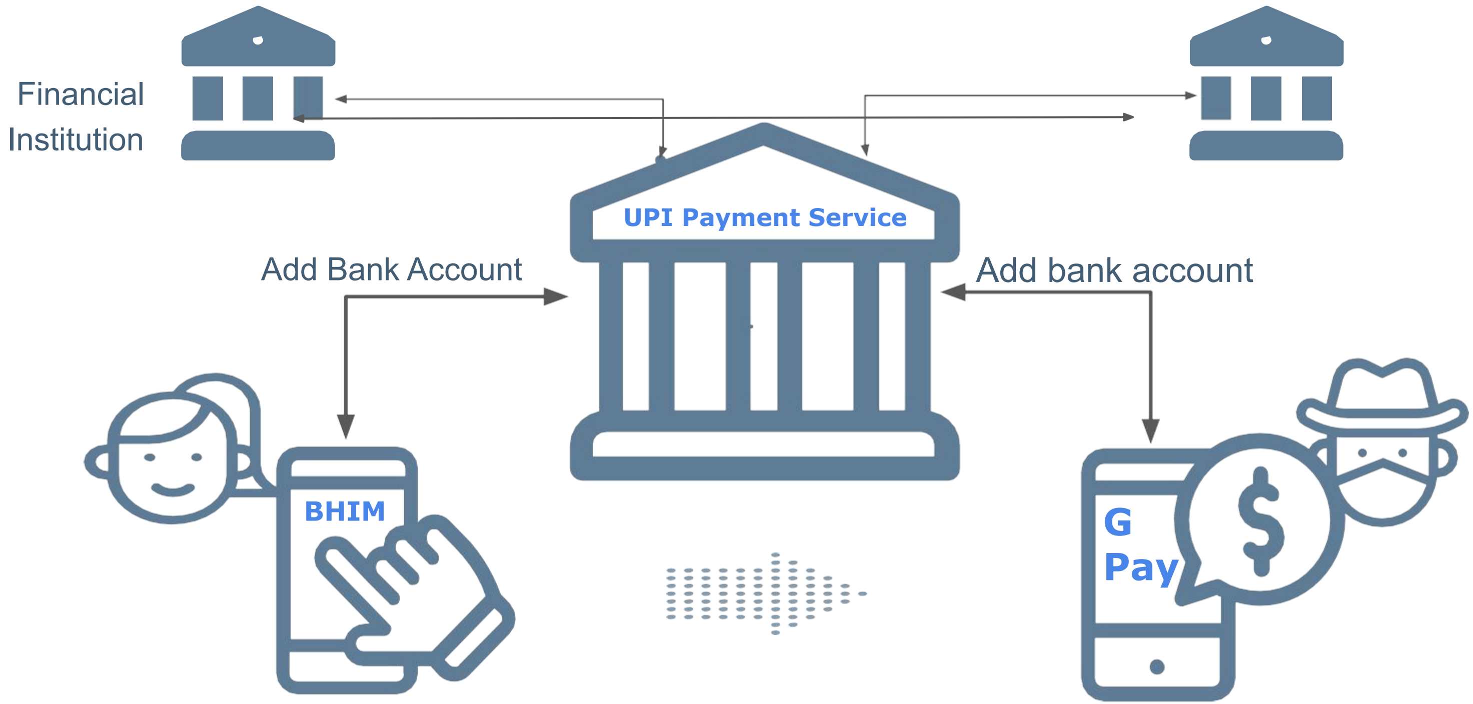
Payment Service Provider

Add Money to Wallet

Add Money to Wallet

India was predominantly a cash-based economy and while payment app existed, they were not the chosen mode of payment

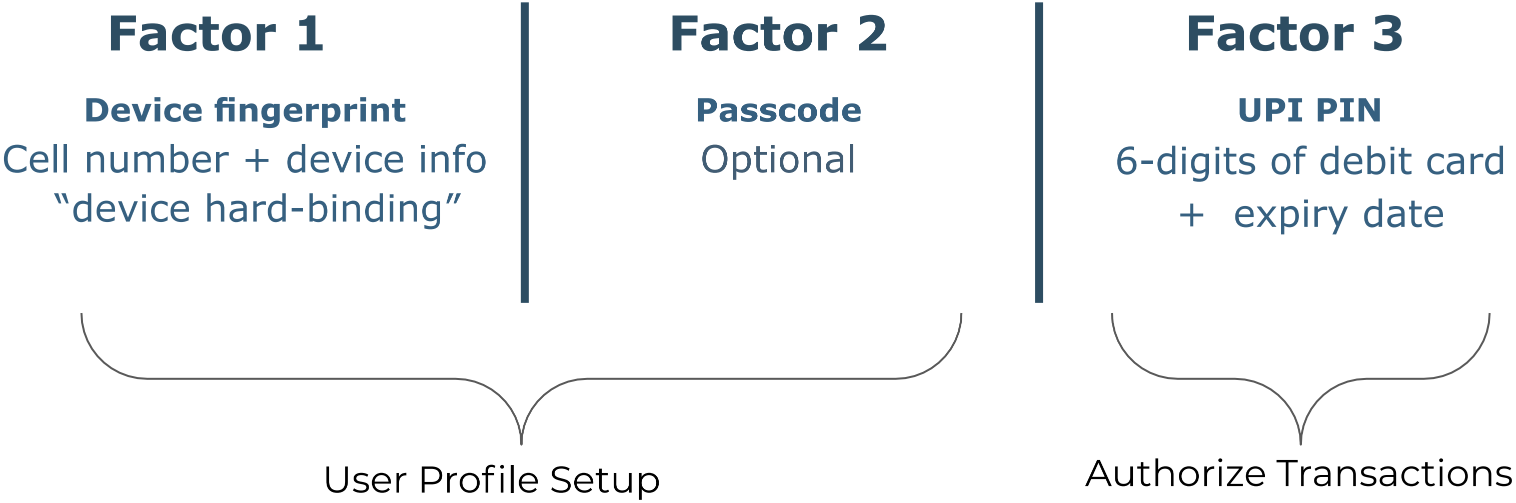
Mobile Payments using Unified Payments Interface



In 2016, the National Payments Corporation of India launched UPI to enable free instant micro-payments from a mobile platform.

UPI’s “Broad Guidelines”

User’s primary cell number (UPI ID) must be registered with the bank out-of-band



Factor 1

Device fingerprint

Cell number + device info

“device hard-binding”

Factor 2

Passcode

Optional

Factor 3

UPI PIN

6-digits of debit card

+ expiry date

User Profile Setup

Authorize Transactions

Reverse Engineering Barriers

Protocol Analysis

Unpublished protocol and no back-end access to UPI servers.
Analyze the protocol through the lens of UPI apps.

Evading App Defenses

Security defenses are many and differ for each app

Evading App Defenses

Defenses:

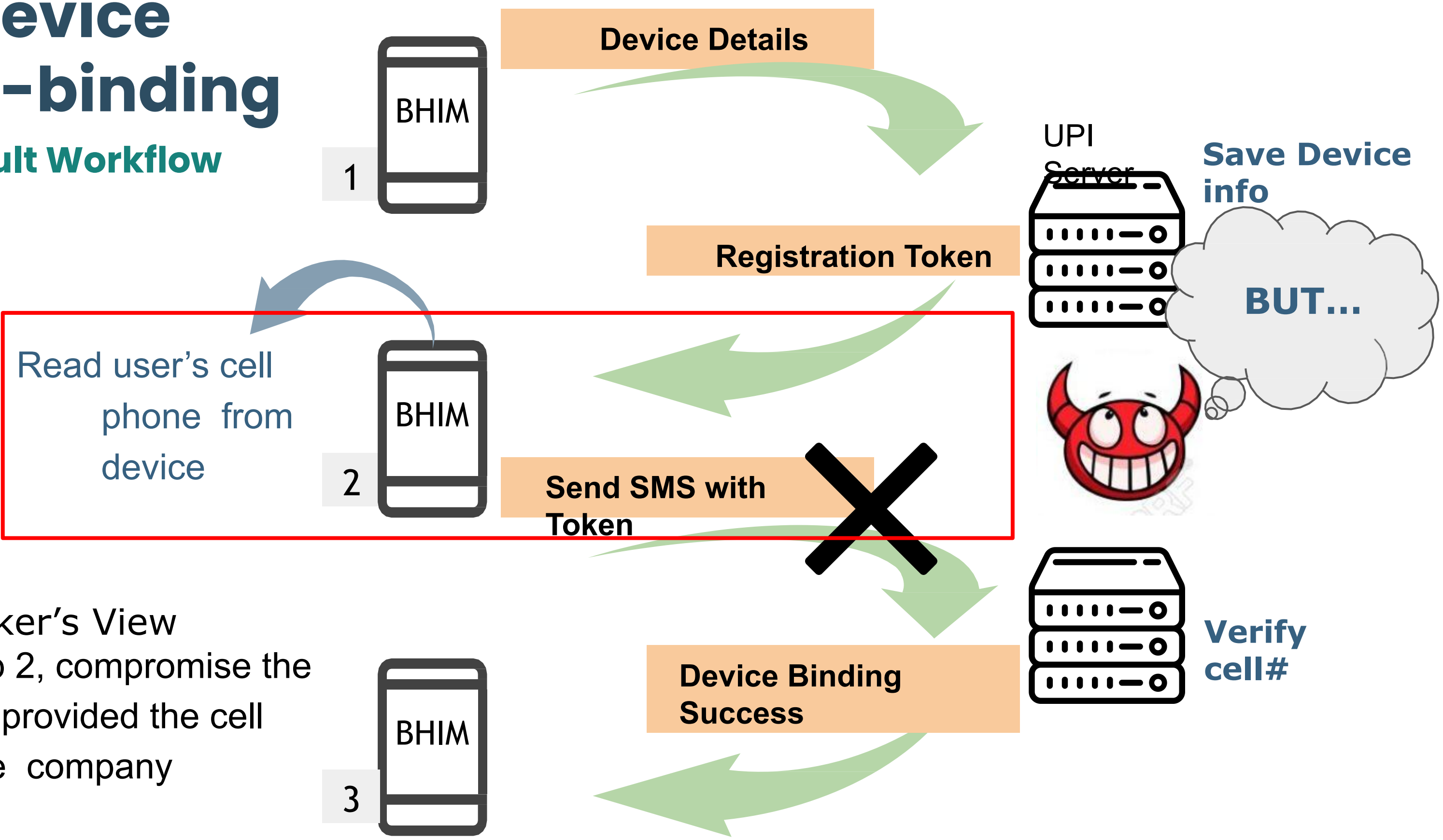
- Obfuscated
- Use encrypted communication
- Emulator detection built-in
- Requires a physical SIM card to be present on the phone
  - Makes dynamic analysis difficult
- UPI apps undergo a thorough security review in India

Approach:

A combination of static reverse-engineering, code instrumentation and traffic analysis

Device Hard-binding

Default Workflow



1 BHIM

Device Details

UPI Server

Save Device info

BUT…

Registration Token

Read user's cell phone from device

2 BHIM

Send SMS with Token

Verify cell#

Device Binding Success

3 BHIM

Attacker's View

To attack Step 2, compromise the protections provided the cell phone company

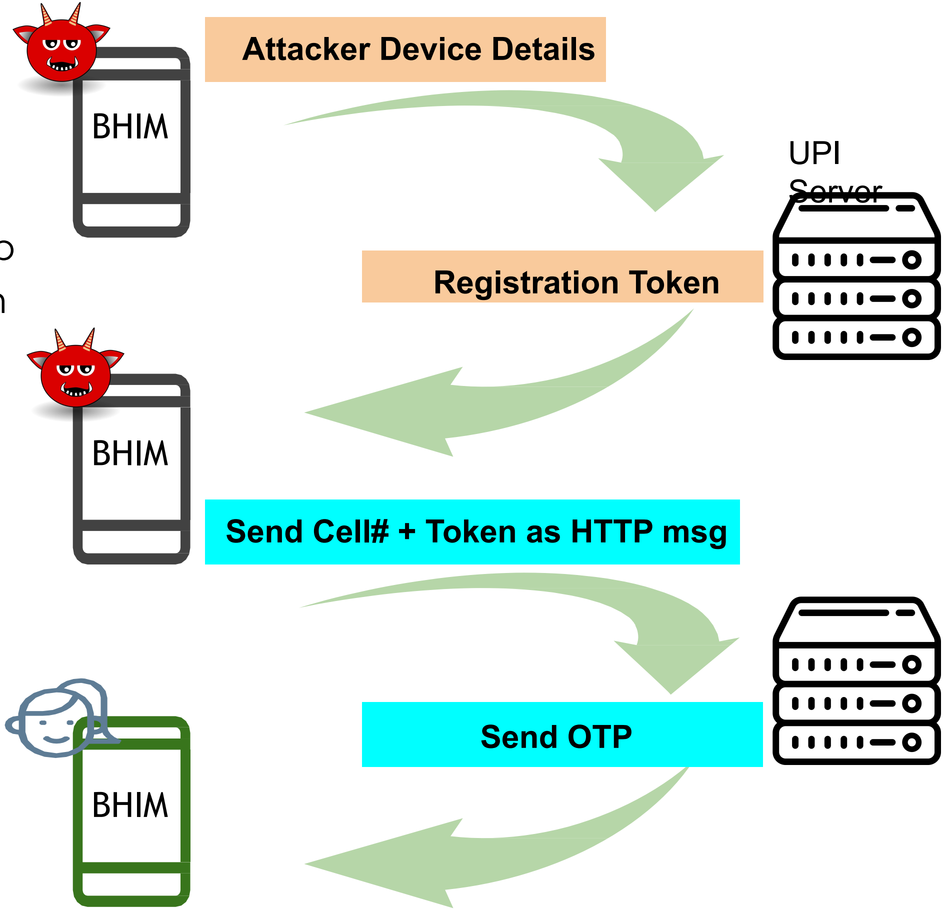
Device Hard-binding

Alternate Workflow

Attacker can induce a failure in step 2 of default workflow by turning on airplane mode

Attacker enters victim cell number from an attacker device

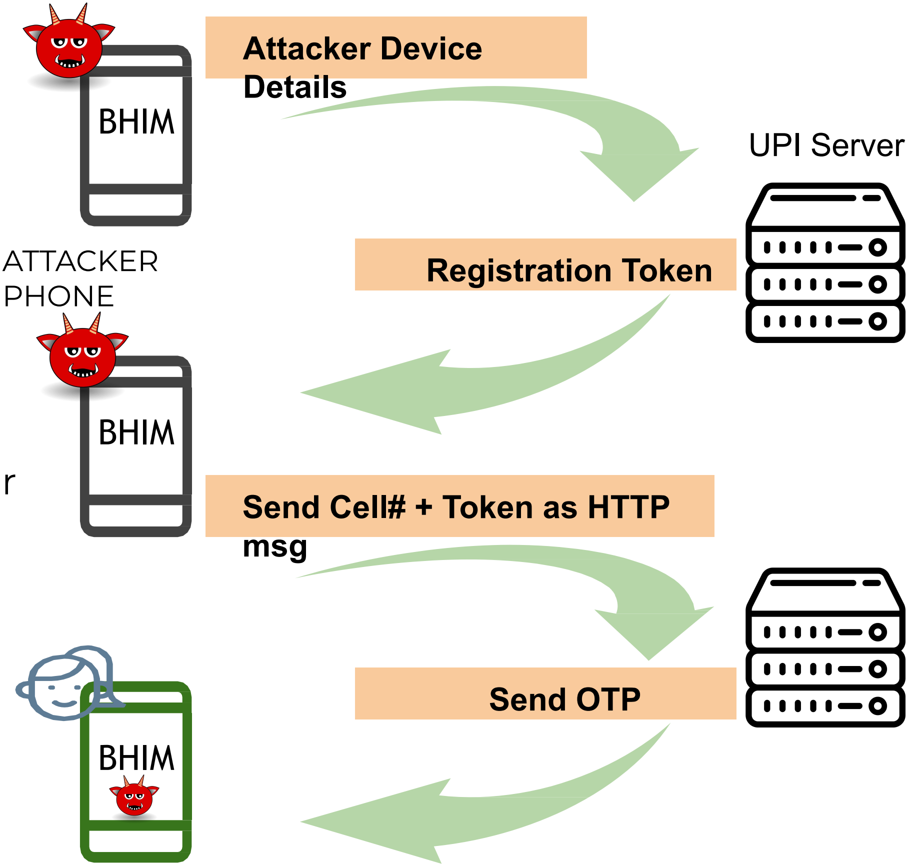
Alternate workflow may allow an attacker to bind her cell phone with a cell number registered to bank account of another user



Breaking Device Binding

Attacker enters victim's cell number

Trojan needs RECEIVE_SMS permission to read OTP



Attacker Device Details

Registration Token

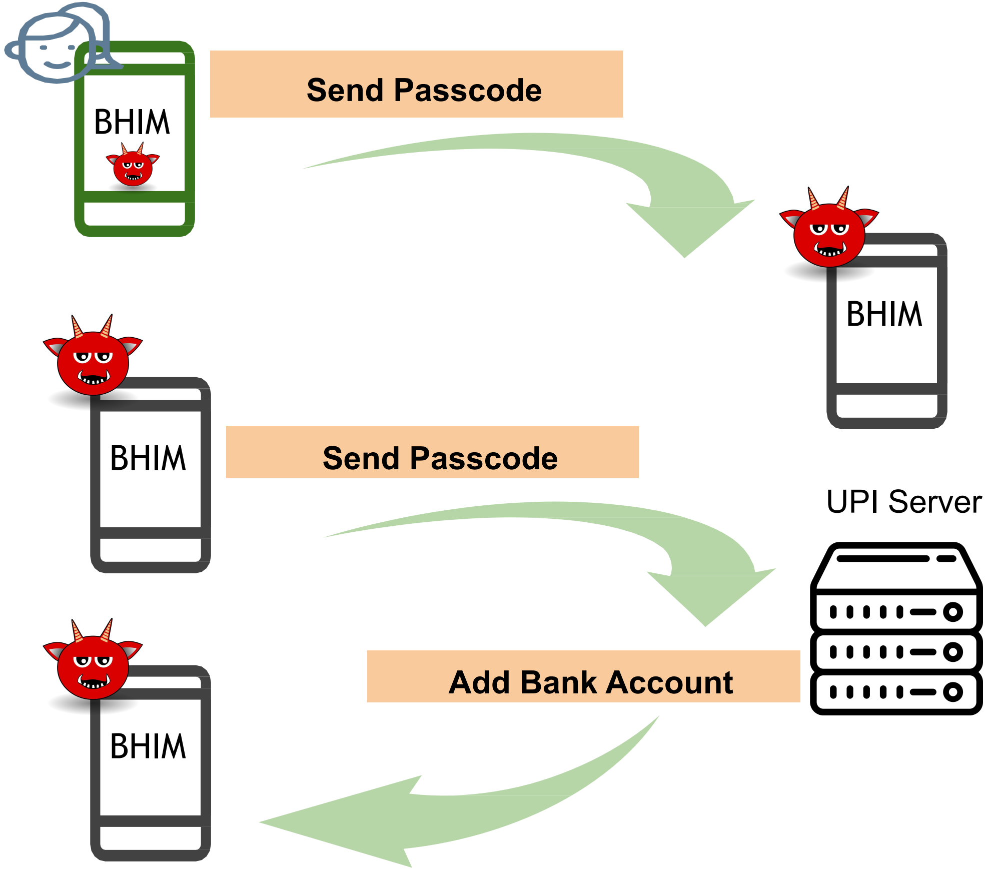
Send Cell# + Token as HTTP msg

Send OTP

Leak Passcode

Use an overlay on BHIM's passcode entry screen

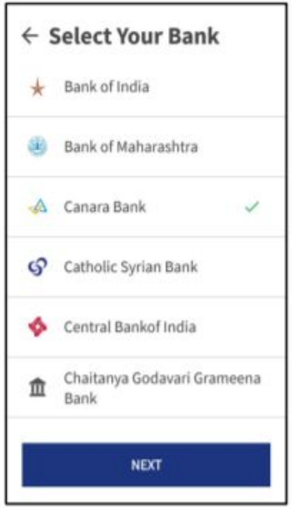
No additional permissions required



Add Bank Account

UPI server appears to allow brute-force attacks. An attacker can learn of all bank accounts of a user

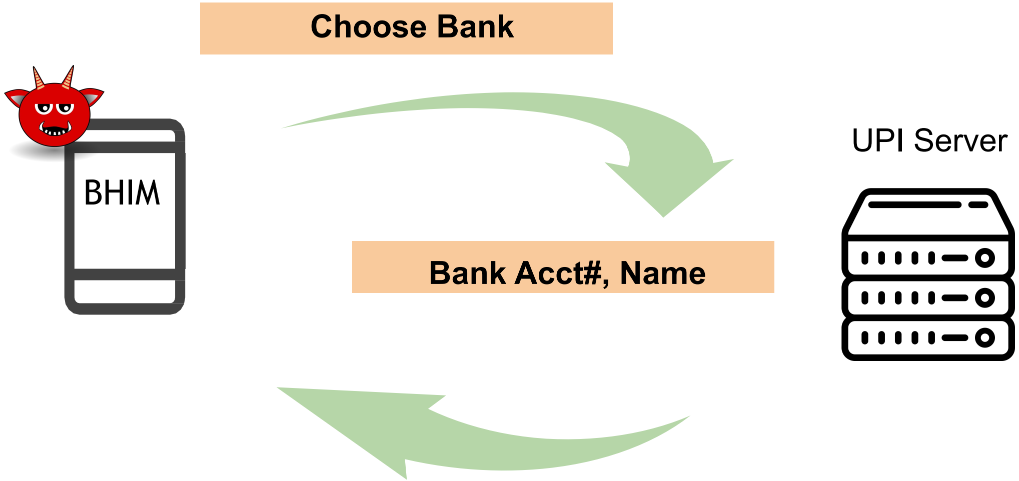
UPI server reveals sensitive bank info without the user providing any bank specific secrets



← Select Your Bank

- Bank of India
- Bank of Maharashtra
- Canara Bank ✓
- Catholic Syrian Bank
- Central Bankof India
- Chaitanya Godavari Grameena Bank

NEXT



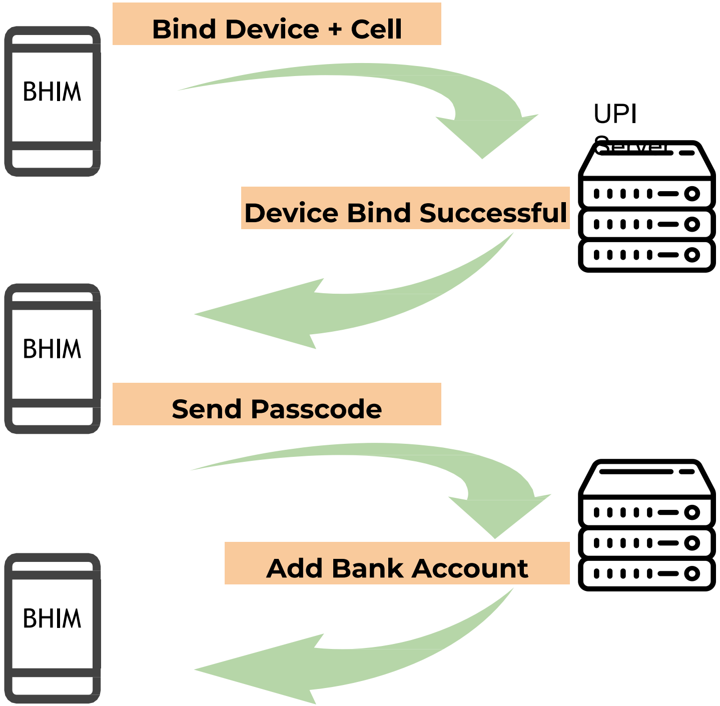
Choose Bank

Bank Acct#, Name

Attacker can start bruteforcing with the most popular banks

New UPI User vs. Existing User

For an existing user, attacker can sync a user's bank account through UPI without providing any bank-related secrets



BHIM

Bind Device + Cell

UPI Server

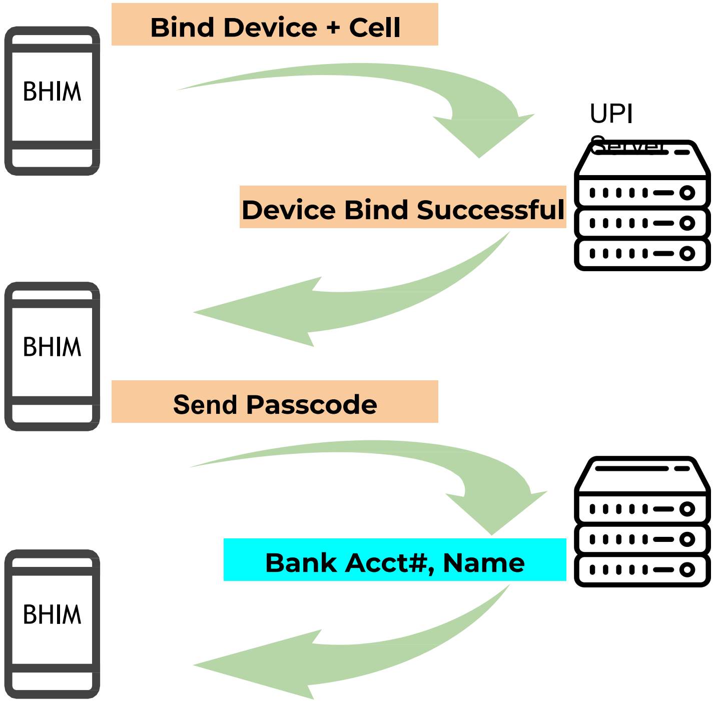
Device Bind Successful

BHIM

Send Passcode

Add Bank Account

BHIM



BHIM

Bind Device + Cell

UPI Server

Device Bind Successful

BHIM

Send Passcode

Bank Acct#, Name

BHIM

Authorize Transaction: UPI PIN

- UPI PIN can be leaked the same way as the passcode.

Setting UPI PIN

- Requires partial card details printed on a card
- Transactions require complete card number + secret PIN shared with the bank

Setting UPI PIN requires only partial debit card info and NO secret - a lower bar in India

Conclusion

- They uncover core security holes in the workflow of UPI 1.0
  - Using an attacker-controlled app, we show how an attacker can attack a user's bank account and steal money from him
- They responsibly disclosed the vulnerabilities to CERT-IN and makers of UPI in 2017
  - Contacted all the app vendors
- UPI 2.0 released in August 2018
  - Fixed the alternate workflow we exploit, but other security holes remain
- Other attack vectors that could potentially compromise UPI 2.0
  - SMS spoofing, loss of user's device or compromising the system
- Calls for proper security vetting of the proprietary protocol since discussions are on to make UPI global

References

[1] Manal Adham, Amir Azodi, Yvo Desmedt, and Ioannis Karaolis. How to attack two-factor authentication internet banking. In Financial Cryptography and Data Security, pages 322–328, 2013.

[2] Mohammed Aamir Ali and Aad van Moorsel. Designed to be broken: A reverse engineering study of the 3D Secure 2.0 Payment Protocol. In Financial Cryptography and Data Security, pages 201–221, 2019.

[3] Samaher AlJudaibi. Research paper for mobile devices security. 2016. https://www.researchgate.net/p ublication/309675787_Research_Paper_for_Mob ile_Devices_Security.

[4] APKTOOL. https://ibotpeaches.github.io/Ap ktool/, 2018. [Online; accessed October-2018].

[5] BHIM. https://play.google.com/store/apps/d etails?id=in.org.npci.upiapp, 2016. [Online; accessed October-2018].