Security Issues of Unified Payments Interface and Challenges: Case Study

Conference Paper · May 2021

DOI: 10.1109/ICSCCC51823.2021.9478078

3 authors, including:

Yash Madwanna
National Institute of Technology Karnataka
2 PUBLICATIONS **3** CITATIONS

SEE PROFILE

Security Issues of Unified Payments Interface and Challenges: Case Study

Yash Madwanna, Mayur Khadse, B. R. Chandavarkar
Department of Computer Science and Engineering
National Institute of Technology Karnataka
Surathkal, Mangalore, India
yashmadwanna1997@gmail.com, khadse.mayur58@gmail.com, brcnitk@gmail.com

***Abstract*—NPCI, which stands for National Payment Corporation of India, was the organisation behind the idea of UPI, a user-friendly system in which funds can be directly transferred from the bank account to the account using a mobile phone. UPI is based on the concept of 1 click 2-factor authentication. The first factor is the user's mobile phone itself, and the second factor is MPIN or bio-metrics. It is based on the IMPS(Immediate Payment Service), but there are considerable differences between both services, and we will observe it. With a foresight to make the Indian economy cashless, it helps people transfer funds in an immediate and real-time process. It has played a major role in the revolution of cashless transactions in India. Although significant UPI users are minor and much lesser compared to the Indian population, over 2.07 billion transactions per month have been made by UPI by October 2020, which makes it our essential part of our day-to-day life. This paper will discuss the working of UPI, how UPI is different from conventional cashless transaction methods. After that, we will discuss how the attacker can find the UPI's loopholes (here we reviewed UPI BHIM 1.0) and empty the victim's bank account. The attacker can make these attacks remotely, and these attacks can affect a single user to multiple users. We will also discuss how the attacker can achieve his/her goal using a malicious App. In the end, we will see how UPI BHIM 2.0 update was successful in covering this security loophole.**

Keywords:UPI, MPIN, NPCI, IMPS, BHIM

I. Introduction

National Payment Corporation of India(NPCI) [1] released UPI [2] as a easy-to-use option for online payment in April 2016 under proper guidelines from the Reserve Bank of India(RBI). After the government of India announced the demonetisation of currency notes, the popularity of UPI- based payments skyrocketed. Various third-party UPI apps like PhonePe, PayTM, Google Pay(Tez) were introduced to facilitate the UPI transactions. The purpose of UPI is to achieve the following goals: [2]

1) Universal electronic payments.
2) Cashless Society.
3) One platform to access multiple bank accounts.

The UPI was designed for the needs of the Indian population; it follows the mobile-first approach as the majority of the population in India is connected to the internet via smartphones. UPI has only one or unique ID of sender and recipient. It is one of the world's sophisticated public payment infrastructures and a suitable option to replace mobile wallets. Making payments through UPI is very easy either by just sending directly to the contact number or entering the Virtual Payment Address(VPA) or UPI ID of the receiver, which is generated when the user registers for the UPI service and the funds are transferred instantly. In contrast, other payment methods like NEFT and IMPS requires sensitive information like Bank account number, IFSC number. One more significant feature of UPI is that a user can register as many bank accounts as he or she has and toggle between them to send and receive money. Each account will have its unique virtual ID.

Unlike the traditional online payment methodologies, like IMPS, NEFT which requires users to give sensitive information like account number, IFSC code; UPI facilitates VPA, Virtual Payment Address (VPA: A VPA is an ID, which the user creates by linking their accounts to the mobile application of bank) as identification of payment to give and take of money. A single click and 2-factor (2-fa) authentication is the basis of the working of UPI. Scanning QR codes or using VPA addresses for on-the-spot transactions makes it one of the simplest and user-friendly payment methods; it also safeguards us from the hassles of cash on delivery. It can also be used for merchandise payments, e-commerce payments, P2P(peer to peer) transactions; transaction requests can be scheduled and paid according to our requirement and convenience. [2] [3] 12

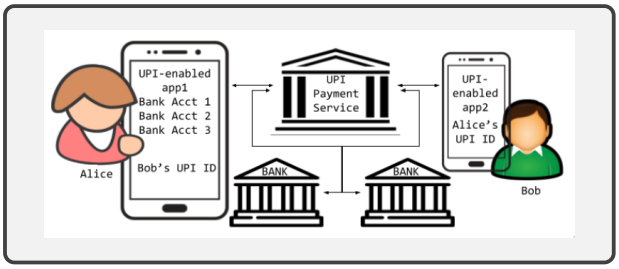


Fig. 1: UPI based funds transfer [4]

Fig. 1 shows UPI-based money transfer system. With its easy to integrate features with third-party applications in smartphones, UPI proved to be a major advantage in promoting cashless transactions across India compared to other digital payment methods like internet banking. Payer and payee must have an account in the UPI-enabled apps, e.g. Google Pay, PayU, BHIM, Etc. In those apps, the user will register his/her bank account(s); during this registration, VPA will be created, and the authenticity of

the user and the bank account will be verified by checking information provided by the user and user's mobile number which he/she has registered with mentioned bank accounts in the UPI enabled apps. Payer and payee's apps are linked to their respective PSP(Payment Service Provider), which is connected to the interface provided by UPI. That is a slight overview of how an actual transaction happens. We will discuss it in detail in the upcoming sections.

As a vast majority of smartphone users in India use the Android operating system, the vulnerabilities in the Android system can be exploited by hackers to perform fraudulent transactions and steal money from users. Also, social engineering attacks have been successful in extracting money from unsuspecting users. In this paper, we focus on working on UPI and analysis of vulnerabilities of UPI. The UPI is a proprietary protocol, so its implementation details are not available, so its analysis is an arduous task. However, we try to use previous research to obtain more information on the UPI that is also significantly less and not sufficient.

II. Working of UPI

Before having overlook about few security challenges in UPI, Let us see how UPI works internally. Mobile banking transactions are not 100% secure. There are some concerns and vulnerabilities, but still, the banks and the government promote online banking and mobile banking platforms because of their availability, easy to use nature, mobility, and ability to reach many users. Nowadays, banks encourage users to use such online platforms instead of traditional methods like filling forms, making Demand Drafts, Etc. As such, methods are more convenient and provide more user convenience without compromising security aspects with low operational costs. We will face some common terms in this section constantly. Let us have an overview of them:

1) VPA: Virtual Private Address. An address of the format <mobile number>@upi used to transfer money using the UPI Apps. Users can create multiple VPA's. UPI-based fund transfer uses VPA internally to look up the account number. [5]
2) PSP: It stands for payment support providers. They are nothing but the Apps of different banks or other platforms that provide payment service to end user [6]

As we know UPI is IMPS based method, but there are some main differences in terms of their working and costing too. Refer to table I.

Now let us focus on our main topic, working of UPI:

A. *Architecture of UPI:*

Consider the given diagram Fig. 2 the term PSP we have already seen. NPCI provides us UPI interface. While P2P stands for peer-to-peer transaction here. The diagram shows in detail the internal working between Player PSP and UPI components which are shown in the following diagram.

Index	UPI	IMPS
1	No need of card/bank account details ,CVV , net banking passwords etc	Require details like bank account number , IFSC , account holder's name etc.
2	It uses VPA and mobile number.	It uses MMID (Mobile Money Identifier)a 7 –digit number.
3	It is the cheapest way of transaction with very minimal charges of less than 0.50Rs.	Charges can vary from 2.50 to 25 Rs.

TABLE I: Comparison between UPI and IMPS with respect to key features.

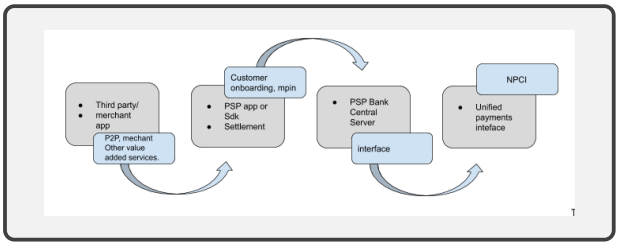


Fig. 2: Working of UPI

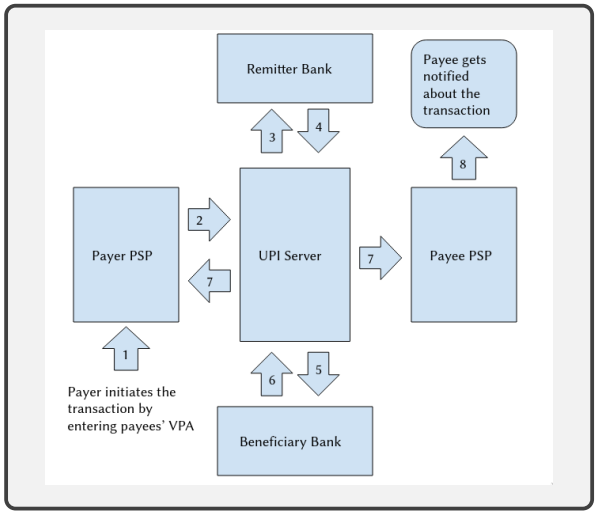
1) Push and Pull Transactions: UPI provides two types of transactions which are "push" transactions and "pull" transactions.

- In PUSH transactions, the transaction is initiated by the payer when he enters the VPA address of the payee and then initiates the transaction by entering the amount and UPI MPIN [7]. Then the request is transferred to the UPI servers who look up the payer and the payee bank account in their database, and then it will debit the amount from the payer's account, and it will credit to the payee's account. This way, the transaction is completed, and the payer and payee are informed about the transaction. The full flow is explained in Fig.3a
- In a PULL transaction, the payee initiates the transaction by entering the VPA address of the payer. The PSP transfers the request to the UPI servers, which then transfers the request to the payer's PSP, and the payer receives a request for payment to the payee. He can either reject the request or accept the request. If he accepts the request, then enters his MPIN, and the transaction is processed, and after the amount is transferred successfully from the payer to the payee, the notification is sent to both about the successful completion of the transaction; The flow is shown in Fig.3b

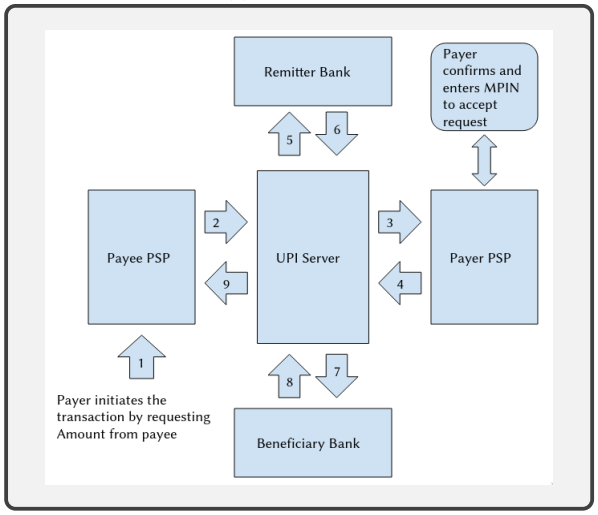
B. *User Registration*

A user can register for UPI service if he/she has a smartphone and has a bank account. Following mentioned steps can be followed for successful registration on UPI Apps [1]:

1) Install UPI PSP Apps from the respective app stores of different mobile platforms like android or ios like Phone Pe, PAYTM, Google Pay, Etc.
2) Select cell number associated with the bank account. It is used to send SMS containing Device



(a) Push Transaction



(b) Pull Transaction

Fig. 3: Different Transaction flows of UPI

fingerprint(information containing the IMEI number, Device Id, App Id) for doing the first-factor authentication. This SMS is encrypted with PKI [8] for security.

3) Select and link the desired bank account and ensure that the bank supports UPI.
4) With the expiry date and last six digits of the debit/credit card set the UPI pin. An OTP [9] message will be sent to the user's registered phone number for verification. This serves as a second-factor authentication.
5) User can also set up the screen lock for the App to increase the security
6) Finally choose a VPA for the account and then user have successfully registered for making transactions using UPI. [2] [5] [3]

III. Security Analysis of UPI

UPI has not published the protocol, and there is no back-end access to the server. So till today, whatever information we have is based on the literature survey of multiple research papers on this topic, information available about UPI app structure, and few concepts of reverse engineering we learned. [10] [11]

We had to understand how reverse engineering works, how app defences work actually. Each App has encrypted communication and undergoes thorough security analysis before getting published for public use.

A. Security Analysis and review on BHIM 1.0:

This security analysis is of the UPI BHIM 1.0. We have reviewed the following type of attack possible on the BHIM [12] 1)Trojan attack by unauthorised registration through the victim's phone: For this attack, we can assume that our victim is a descent user with knowledge of how to handle Apps on his/her phone and how to protect his/her device. While the attacker will use Trojan App or malicious App, which will enter the victim's phone by somehow pretending itself a legit App. It is an attack by a Pre-installed Harmful App(PHA) [13]. The overview of the attack occurs in the following manner. An attacker disables client-side security on his deceive version of BHIM. The attacker creates a (C & C) server, publishes the deceive version of the App on different App store platforms available for the public. Few users thinking it as an original App download and install it. The App has RECEIVE_ SMS permission [14]. A user thinks of it as an authorised version of BHIM and uses it on his/her phone. [4]

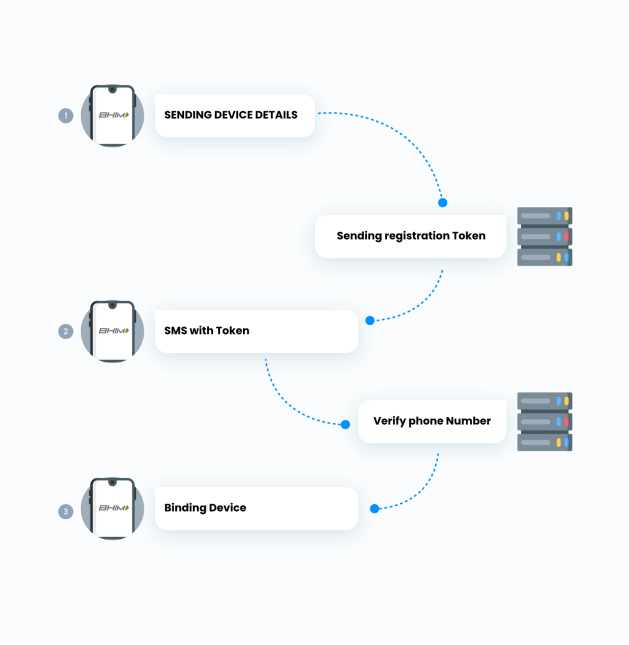


Fig. 4: Default working flow of BHIM UPI

To launch the attack victim's phone number is essential. It can be achieved by READ_PHONE_STATE permission. [14] This permission will help the attacker to get the phone number. It is the most common permission used by around 35% of the mobile applications available across the globe on different platforms. [14] The attacker can get the phone number in the following way: the attacker has a set S of all targeted phone numbers; the following steps happen before the attack: (i) The attacker sends an SMS

with the following attributes and content for each phone number included in the set S [receiver's phone number, "Any random text"] (ii) The attacker now makes a subset of the above set which lists numbers who have installed the malicious App. The App will search for the string "Mentioned in above SMS", and the App will inform the attacker about it. This number will be listed as the victim's phone number. In this way, READ_PHONE_STATE permission helps the attacker. From this point, let us assume the attacker's malicious App is installed on the phone, and it has reported the victim's phone number. Let us see the actual workflow of UPI BHIM refer Fig. 4:

1) BHIM App takes the user's information from the user and sends it to the UPI server. The server saves the information & sends the registration token back to the user.
2) App remaps it on the user's device and sends an SMS with a token to the server. The server verifies the phone number and binds it to the device information.
3) Confirming binding to the client.

Fig. 4 can help us to understand the default flow. We will see how the attacker can modify the above workflow and will see the attacker's point of view: For the attacker, it is the most convenient to compromise the second step. The attacker must compromise the protection provided by the mobile network company. Instead of doing it, the attacker can still compromise it more conveniently by changing some aspects of the above workflow. The attacker can impose or induce failure in step 2 by turning ON aeroplane mode sending his/her details to the UPI server. The attacker exploits the BHIM's workflow to bind his device to the victim's phone number. Refer to Fig. 5 to understand.

1) The attacker starts by putting his phone in aeroplane mode. We can connect to the internet through Wi-Fi, even without a mobile network. The attacker is doing the same here. The attacker's BHIM App initiates the process by sending the details of his mobile/device. The UPI server replies to it by sending a token of registration for the attacker. If we turn off the mobile network, SMS delivery will fail; that is nothing but the aeroplane mode we mentioned above. The attacker here uses the same basic method. The attacker gets help out by BHIM for log- in/ getting a phone number. The attacker loads victim's phone number [4]. The device registration token of the attacker's device is sent by BHIM as HTTPS SMS, and the victim's number is received by the UPI server. The UPI server uses it for hard binding. The UPI server sends the One Time Password to the victim.
2) This OTP is received by the malicious App on the attacker's phone. It forms an HTTPS message which contains the above OTP. Later it sends the HTTPS message to the C&C server.
3) An SMS containing the victim's phone number is created by the server and sent. The authenticity of the one-time password is usually verified by the BHIM. It accepts it only if it belongs to the legit UPI server. The attacker smartly bypassed this before launching the attack in the deceive version by turning off this safety feature. It is a significant security loophole in the workflow.
4) Now, four digits password/pin is needed at the attacker's side. Malicious App on the victim's device will use an overlay of BHIM's passcode entry section.

Fig. 5: Change of flow when an attacker exploits the default flow.

Our next attack extends the previous attack to perform transactions using UPI on the bank account of a victim even if the victim does not use UPI. To set the UPI MPIN [7] only the last six digits of the debit card and, in addition to it, the expiry date of the card. The debit card is often used in India for paying at restaurants and shopping centres or any commercial places Etc. They can also obtain phone numbers by keeping diaries for reviews and containing a section to provide phone numbers. Many users, without thinking give their number in such places. Generally, in such POS [15]transactions, the debit card and PIN both are required to execute a transaction successfully. Nevertheless, in the case of UPI, to reset the UPI MPIN, we need the last six digits of the debit card number in addition to it its expiry date. Debit cards generally have the name of the bank written on them. UPI does send OTP to the registered number to verify the MPIN change. However, if the victim also has a PHA or Trojan App installed on their phone, then, as discussed in the previous attack, the OTP can be intercepted by the attacker. Suppose the victim's debit card information is also known to the attacker. In that case, the attacker has all the information needed to access the victim's account and can transfer funds from the victims account to anywhere through UPI even though the victim never even uses UPI service.

B. BHIM 2.0 update security review:

UPI made available the first update, UPI 2.0, in August 2018. UPI 2.0 does prevent the attack we discussed above by surveying various articles and research papers. Let us

see how UPI 2.0 prevents the attack. We surveyed the UPI 2.0 version of BHIM; still, there is not much information available on the security analysis of UPI version 2.0. However, NPCI later made it compulsory to update the BHIM App to its latest version. In UPI 2.0, along with whatever was necessary for physical binding in 1.0, BHIM App also uses some other parameters. It now also uses IMEI of the device, type of the network, SIM number, Etc., as more information to the UPI server for device hard-binding. Due to which the security loophole we have seen is now patched, and App is more secure than it used to be before. [4]

C. *Offline UPI introduction:*

Here offline UPI means we are using basic functions of UPI even without having an internet connection [15]. It relies on USSD based mobile banking. To use this facility, all we have to do is to dial *99#. After this step, we will have different options like send money, request money, balance information, changing the UPI pin, Etc. Sending money is similar to the usual UPI. Here also, we need either a mobile number or a virtual address of the receiver. The Maximum amount of Rs.20,000 is allowed till now. We can also access details of the sender and the receiver, which means the transaction occurred between them. For doing both sending and receiving money, the pin is needed. After the transaction, we can see whether the money is debited or credited on our screen. The user can also have a look at his profile details. It is advantageous in a region where the internet is not easily accessible. The security analysis of this type of UPI has not covered in this paper.

IV. Conclusion

In this paper, we represented our literature survey about UPI to understand its working. Later we discussed the security loophole that existed in the BHIM 1.0 App. This review section was majorly based on the research work and research paper published by people from the University of Michigan [4] along with other literature resources. Further, our paper discusses how the attacker could exploit the security loophole present in the BHIM UPI 1.0. For the attack to happen, the victim had to install the attacker-controlled malicious App; so the attacker will be able to launch the attack. In later update 2.0, this security loophole was covered well. Unfortunately, there are some flaws present that we have not discussed here [4]. Discussion on offline UPI in an introductory manner is also there. What kind of services it can provide to its users and how a user can access offline UPI is also discussed in this paper. In the end, readers will get enough knowledge and information about UPI by reading this paper and, maybe in the future, and it will be helpful to others to implement some new payment infrastructure by avoiding the security loophole discussed in the paper.

References

[1] ”product overview”, https://www.npci.org.in/what-we-do/upi/product-overview, [Accessed: 13-10-2020].

[2] S. Mohapatra, Unified payment interface (upi):a cashless indian transaction process., International Journal of Applied Science and Engineering 5 (1) (2017) 29–42. doi:10.5958/2322-0465.2017.00004.1.

[3] Dr. Virshree Tungare, A study on customer insight towards upi (unified payment interface) - an advancement of mobile payment system, International Journal of Science and Research (IJSR) 8 (4) (2019) 1408–1412.

[4] R. Kumar, S. Kishore, H. Lu, A. Prakash, Security analysis of unified payments interface and payment apps in india, in: 29th USENIX Security Symposium (USENIX Security 20), USENIX Association, 2020, pp. 1499–1516.
URL https://www.usenix.org/conference/usenixsecurity20/presentation/kumar

[5] Dr. Abhijeet Chatterjee, R. Thomas, Unified payment interface (upi): A catalyst tool supporting digitalization – utility, prospects issues, International Journal of Innovative Research and Advanced Studies (IJIRAS) 4 (2) (2017) 192–195.

[6] K. K. Lakshmi, H. Gupta, J. Ranjan, Upi based mobile banking applications – security analysis and enhancements, in: 2019 Amity International Conference on Artificial Intelligence (AICAI), 2019, pp. 1–6. doi:10.1109/AICAI.2019.8701396.

[7] S. Mohan, ”what is the mpin in upi?”, https://razorpay.com/learn/generate-upi-pin/, [Accessed: 19-10-2020] (2019).

[8] R. Hunt, Pki and digital certification infrastructure, in: Proceedings. Ninth IEEE International Conference on Networks, ICON 2001., 2001, pp. 234–239. doi:10.1109/ICON.2001.962346.

[9] M. H. Eldefrawy, K. Alghathbar, M. K. Khan, Otp-based two-factor authentication using mobile phones, in: 2011 Eighth International Conference on Information Technology: New Generations, 2011, pp. 327–331. doi:10.1109/ITNG.2011.64.

[10] ”android app reverse engineering 101”, https://maddiestone.github.io/AndroidAppRE/, [Accessed: 18-11-2020].

[11] A. D. Anmol Misra, Reverse engineering android applications, https://www.oreilly.com/library/view/android-security/9781439896464/K14268_C006.xhtml, [Accessed: 24-11-20202].

[12] ”how to transact using bhim”, https://www.bhimupi.org.in/, [Accessed: 15-10-2020].

[13] Y. L. Koh, Investigating potentially harmful applications on android (07 2018).

[14] M. Base-Bursey, What are app permissions - a look into android app permissions, https://www.wandera.com/mobile-security/app-and-data-leaks/app-permissions/., [accessed: 26-11-2020].

[15] K. Nguyen, What is pos transaction? the basics explained, https://blog.magestore.com/pos-transaction/, [Accessed: 13-3-2021] (2021).