International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 9, Issue 7, July 2020 ||

Security in Unified Payments Interface and Payment Apps in India

Mr. Sachin Santu Shende

Lecturer, Department of Computer Technology, Jayawantrao Sawant Polytechnic Hadapsar, Pune, India

ABSTRACT: Since 2016, with a solid push from the Government of India, cell phone based installment applications have become standard, with over $50 billion executed through these applications in 2018. Huge numbers of these applications utilize a typical foundation presented by the Indian government, called the Unified Payments Interface (UPI), however there has been no security examination of this basic bit of framework that bolsters cash moves. This paper utilizes a principled philosophy to do a point by point security examination of the UPI convention by figuring out the plan of this convention through seven well known UPI applications. We find beforehand unreported multifaceted confirmation configuration level defects in the UPI 1.0 particular that can prompt noteworthy assaults when joined with an introduced assailant controlled application. In an outrageous form of the assault, the blemishes could permit a casualty's ledger to be connected and purged, regardless of whether a casualty had never utilized an UPI application. The potential assaults were versatile and should be possible remotely. We examine our procedure and detail how we conquered difficulties in figuring out this unpublished application layer convention, including that all UPI applications experience a thorough security audit in India and are intended to oppose examination. The work brought about a few CVEs, and a key assault vector that we announced was later tended to in UPI 2.0.

I. INTRODUCTION

Installment applications have become a standard installment instrument in India, with the Indian Government effectively reassuring its residents to utilize electronic installment strategies after a demonetization of huge money notes in 2016. To encourage computerized smaller scale installments at scale, the National Payments Corporation of India (NPCI), a consortium of Indian banks, presented the Unified Payment Interface (UPI) to empower free and moment cash moves between financial balances of various clients. Starting at July 2019, the estimation of UPI exchanges has reached about $21 billion. UPI's open backend engineering that empowers simple reconciliation and interoperability of new installment applications is a noteworthy empowering agent. Presently, there are around 88 UPI installment applications and more than 140 banks that empower exchanges with those applications by means of UPI . This paper centers around vulnerabilities in the plan of UPI and UPI's utilization by installment applications. We note that programmers are exceptionally energetic with regards to cash, so revealing any structure vulnerabilities in installment frameworks and tending to them is significant. For example, an ongoing overview expresses a 37% expansion in money related misrepresentation and fraud in 2019 in India [12]. Social designing assaults to separate delicate data, for example, once passwords and ledger numbers are normal

Installment applications, including Indian installment applications, have been investigated previously, with vulnerabilities found [9,], and an Indian versatile financial assistance was found to have PIN recuperation imperfections . Notwithstanding, in these investigations, portable applications didn't share a typical installment interface. To the extent we know, an investigation of a typical interface utilized by numerous installment applications has not been done previously. Such an examination is significant in light of the fact that security defects in them can affect clients of various banks and different applications, paying little heed to other more grounded security highlights utilized. We center around the security investigation of the brought together installment interface utilized by numerous Indian installment applications and its structure decisions.

In this work, we utilize a principled way to deal with break down UPI 1.0, beating huge difficulties. A key test is that the convention subtleties are not accessible, however a large number of clients in India use it. We additionally didn't approach the UPI servers. We in this manner needed to figure out the UPI convention through the UPI applications that pre-owned it and needed to sidestep different security resistances of each application, including code confusion and against copying procedures. In spite of the fact that we expand on procedures utilized in the past for security investigation of applications [9] our way to deal with remove the convention subtleties differs dependent on the guards the applications use. We cautiously analyze each phase of the UPI convention to reveal the accreditations required to advance in each stage, discover exchange work processes for validation, and find spillage of client explicit qualities

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 9, Issue 7, July 2020 ||

that could be helpful at a later stage. We present outcomes from the investigation of the UPI convention as observed by seven of the most well known UPI applications in India recorded in Table 1. Of the seven applications we examine, four UPI applications—Google Pay (Tez), PhonePe, Paytm, and BHIM—have a consolidated piece of the overall industry of 88% and are generally acknowledged at many shopping destinations. From a sum of 88 UPI applications, many are minor varieties of BHIM, the lead application discharged by NPCI (likewise the fashioners of UPI). Near 48 banks today issue a bank-marked variant of the BHIM application. Since Android possesses over 90% of the Indian portable piece of the pie [13], we concentrated on the Android forms of these applications.

II. LITERATURE SURVEY

Early mobile payment apps in India were wallet-only apps. They could withdraw money from a user's bank account by asking a user to enter a debit card number, but not deposit money back into the bank account. Post demonetization (in 2016), to encourage cashless transactions, a consortium of Indian banks called the National Payments Corporation of India (NPCI), backed by the Indian government, introduced the Unified Payments Interface (UPI) that allows NPCI-certified mobile apps to do free instant money transfers between bank accounts of different users. UPI apps can inter-operate with each other since they all share the same payment interface. A user of BHIM, for instance, can transfer money instantly for a small purchase from her bank account to the bank account of a shopkeeper who uses Google Pay. Because of this, most stores in India accept mobile payments through UPI apps. Depending on the app, a user can do unlimited transactions up to $1500 per transaction.. Figure 1b shows the UPI money transfer system when compared with the traditional Internet banking system in Figure 1a.

2.1 User Registration on a UPI App

The UPI payment system requires Alice to register her primary cell phone (or cell) number with her bank account(s)out-of-band to send or receive money. UPI uses the cell number (i) as a proxy for a user's digital identity with the bank to look up a bank account given a cell number; (ii) as a factor in authentication via SMS one-time passcodes (OTP); and (iii) to alert users on transactions. The Government of India requires cell phone providers to get copies of government-issued IDs, manually verify the IDs, and do biometric verification before issuing a cell number 1

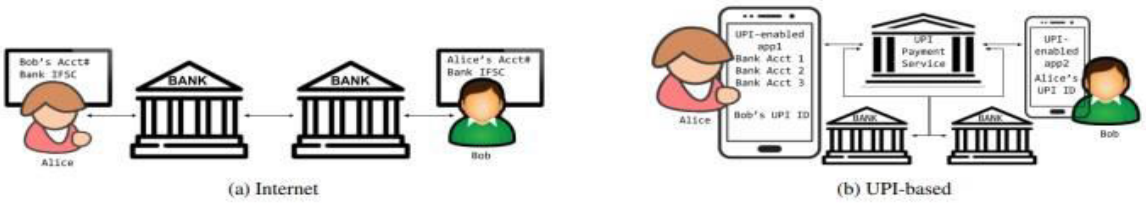

Figure 1: Internet vs. UPI-based Money Transfer

III. UPI SPECS FOR USER REGISTRATION

The UPI specifications released by NPCI [44] provide "broad guidelines" on the client-server handshake between a UPI app and the UPI server. We discuss the protocol details available to us from the specification.

1. **Set up a UPI user profile:** Once a UPI app gets a user's cell number, the app must send an outbound encrypted SMS from Alice's phone to the UPI server. This process is automated and does not involve the user in order to guarantee a strong association between a user's cell phone and her device. According to UPI, this is the "most critical security requirement" of the protocol since all money transactions from a user's device are first verified based on this association. UPI calls this association of a user's device (identified by parameters such as Device ID, App ID, and IMEI number) with her cell number as device hard-binding. The combined cell number and device information (that represents this binding) is called the device fingerprint, which per the UPI spec is the first factor of authentication.

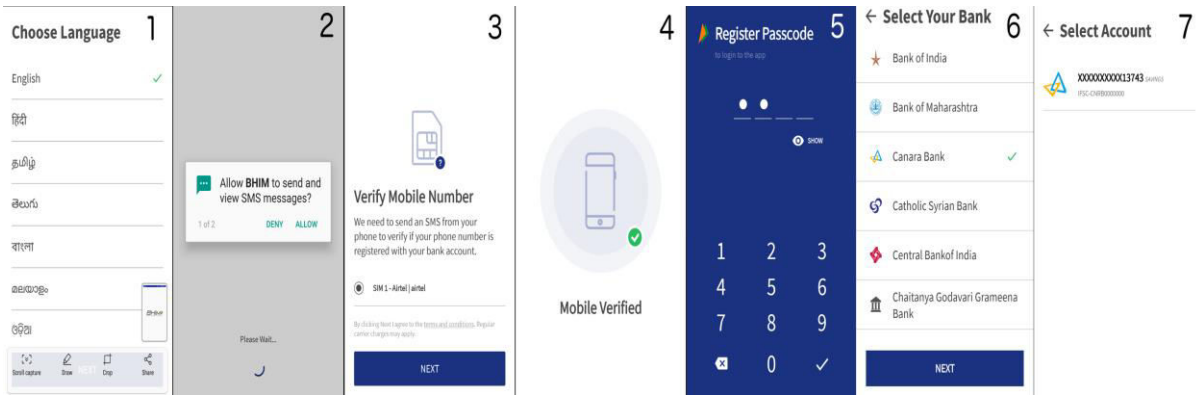

Figure 2: BHIM User Registration Using 3FA

Passcode. The UPI spec considers application passcode as optional and does not undertake responsibility for passcode authentication. UPI leaves it up to a UPI app vendor to authenticate the passcode. Thus, the responsibility to completely authenticate a user is shared betweentwo servers— the UPI server (that verifies devicefingerprint and UPI PIN), and a payment app server (thatverifies an app passcode).

2. Add a bank account: A user's request to add a bank mustbe from the device registered with UPI. Internally, UPIfetches the chosen bank's account number and IFSCcode based on a user's cell number for later transactionsthrough the UPI app.
3. Enable transactions: UPI allows transactions to be doneeither using a cell number or an account number andIFSC code or any UPI ID. UPI spec mandates that alltransactions must at least be 2FA using a cell phone (thedevice fingerprint) as one factor and the UPI PIN asthe second. The spec considers a cell phone as a "whatyou have" factor, which allows UPI to provide "1-click2-Factor Authentication" using the said two factors.

For apps that integrate with UPI, NPCI enforces application security via a code review and certification process. All communication with the UPI server is over a PKI-based encrypted connection. Currently, UPI has become the de facto standard for mobile transactions.

IV. CONCLUSION

In this paper, we used a principled approach to analyze the UPI 1.0 protocol and uncovered core design weaknesses in its unpublished multi-factor authentication workflow that can severely impact a user. We showed attacks that have devastating implications and only require victims to have installed an attacker-controlled app, regardless of whether they use a UPI app or not. All the vulnerabilities identified were responsibly disclosed. A subsequent software update to UPI 2.0 prevents the discussed attack vectors for an exploit. Unfortunately, several underlying security flaws remain that suggest a need for further vetting and security analysis of UPI 2.0, given the protocol's importance for mobile payments in India. We discussed the lessons learned and potential mitigation strategies. Finally, we expect our findings to be useful to other countries that look to implement a common backend infrastructure for financial apps.

REFERENCES

1. Manal Adham, Amir Azodi, Yvo Desmedt, and IoannisKaraolis. How to attack two-factor authentication internetbanking. In Financial Cryptography and DataSecurity, pages 322–328, 2013.
2. Mohammed Aamir Ali and Aad van Moorsel. Designedto be broken: A reverse engineering study of the 3D Secure2.0 Payment Protocol. In Financial Cryptographyand Data Security, pages 201–221, 2019.
3. Samaher AlJudaibi. Research paper for mobile devicessecurity. 2016. https://www.researchgate.net/publication/309675787_Research_Paper_for_Mobile_Devices_Security.
4. APKTOOL. https://ibotpeaches.github.io/Apktool/, 2018. [Online; accessed October-2018].
5. BHIM. https://play.google.com/store/apps/details?id=in.org.npci.upiapp, 2016. [Online; accessedOctober-2018].
6. S. Bojjagani and V. N. Sastry. VAPTAi: a threat modelfor vulnerability assessment and penetration testing ofAndroid and iOS mobile banking apps. In 2017 IEEE3rd International Conference on Collaboration and InternetComputing (CIC), pages 77–86, 10 2017.

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 9, Issue 7, July 2020 ||

7. Sriramulu Bojjagani and V. N. Sastry. STAMBA: securitytesting for Android mobile banking apps. In Advancesin Signal Processing and Intelligent RecognitionSystems, pages 671–683, 2016.
8. Mike Bond, Omar Choudary, Steven J. Murdoch, SergeiSkorobogatov, and Ross Anderson. Chip and Skim:Cloning EMV cards with the pre-play attack. In Proceedingsof the 2014 IEEE Symposium on Security andPrivacy, SP '14, pages 49–64. IEEE Computer Society,2014.
9. Sam Castle, Fahad Pervaiz, GalenWeld, Franziska Roesner,and Richard Anderson. Let's talk money: Evaluatingthe security challenges of mobile money in thedeveloping world. In Proceedings of the 7th AnnualSymposium on Computing for Development, ACM DEV'16, 2016.
10. Kelvin Chikomo, Ming Ki Chong, Alapan Arnab, andAndrew Hutchison. Security of mobile banking. 012006.
11. Tom Chothia, Flavio D. Garcia, Chris Heppel, and ChrisMcMahon Stone. Why banker Bob (still) can't get TLSright: A security analysis of TLS in leading UK bankingapps. pages 579–597, 01 2017.
12. Express Computer, 2019. https://www.expresscomputer.in/news/financial-cybercrime-and-I dentity-theft-in-india-are-increasing-fis/35099/.w to Manage the Economic Fallout of the Coronavirus. *Change*.