International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 26 Issue 7 – DECEMBER 2019.

Merging Multi Cloud Deployment with Multi Bank Payment with Security

R.Vimal Raja[#1] and S.Rajasekar [*2]

[#] *Dept. of CSE, CK College of Engineering & Technology,Cuddalore, Tamil Nadu, India*

[*] *Dept. of CSE, CK College of Engineering & Technology,Cuddalore, Tamil Nadu, India*

***Abstract*— The main aim of this Computer science engineering project is to computerize the bank process, which is used to reduce the customer transaction time as well as user can able to access accounts detail from anywhere by just using mobile phone. This project also mainly intended for the multiple bank account merged with one application to transfer the amount on another account. We cannot go bank for every bank related work like checking our balance, doing transaction and for other small things. To solve this problem a mobile solution was presented which is "Merging Multi Bank Payment System". This application also allows the user to pay bills like mobile, utility etc. through the Mobile. By using mobile banking system the customer can easily transfer money to Someone's bank account. Account tracker android project explains about implementing an app for android mobiles which will help users to know about bank balances in different banks and there transactions information. In present trend usage of apps had became a new trend because of availability of web services on mobiles. By considering these improvements in mobile technology knowing information of money transactions through mobile in less time can be useful application for users. In this application initially users need to install app and update details like listing out different banks and adding new bank accounts.**

***Index Terms*— POS, NFC, MERGING MULTIBANK ACCOUNT, CLOUD SECURITY, HCE**

I. INTRODUCTION

The earliest mobile banking services used SMS, a service known as SMS banking. With the introduction of smart phones with WAP support enabling the use of the mobile web in 1999, the first European banks started to offer mobile banking on this platform to their customers. Mobile banking before 2010 was most often performed via SMS or the mobile web. Apple's initial success with iPhone and the rapid growth of phones based on Google's Android (operating system) have led to increasing use of special mobile apps, downloaded to the mobile device. With that said, advancements in web technologies such as HTML5, CSS3 and JavaScript have seen more banks launching mobile web based services to complement native applications. A recent study (May 2012) by Map Research suggests that over a third of banks have mobile device detection upon visiting the banks' main website. A number of things can happen on mobile detection such as redirecting to an app store, redirection to a mobile banking specific website or providing a menu of mobile banking options for the user to choose from.

Mobile banking is a service provided by a bank or other financial institution that allows its customers to conduct financial transactions remotely using a mobile device such as a mobile phone or tablet. It uses software, usually called an app, provided by the financial institution for the purpose. Mobile banking is usually available on a 24-hour basis. Some financial institutions have restrictions on which accounts may be accessed through mobile banking, as well as a limit on the amount that can be transacted.

Transactions through mobile banking may include obtaining account balances and lists of latest transactions, electronic bill payments, and funds transfers between a customer's or another's accounts. Some apps also enable copies of statements to be downloaded and sometimes printed at the customer's premises; and some banks charge a fee for mailing hardcopies of bank statements.

From the bank's point of view, mobile banking reduces the cost of handling transactions by reducing the need for customers to visit a bank branch for non-cash withdrawal and deposit transactions. Mobile banking does not handle transactions involving cash, and a customer needs to visit an ATM or bank branch for cash withdrawals or deposits. Many apps now have a remote deposit option; using the device's camera to digitally transmit cheque to their financial institution.

Mobile banking differs from mobile payments, which involves the use of a mobile device to pay for goods or services either at the point of sale or remotely, analogously to the use of a debit or credit card to effect an EFTPOS payment.

II. MOBILE BANKING CONCEPT

In one academic model, mobile banking is defined as

Mobile Banking refers to provision and ailment of banking- and financial services with the help of mobile telecommunication devices. The scope of offered services may include facilities to conduct bank and stock market transactions, to administer accounts and to access customized information."

According to this model mobile banking can be said to consist of three inter-related concepts:

Mobile accounting

Mobile brokerage

Mobile financial information services

Most services in the categories designated accounting and brokerage are transaction-based. The non-transaction-based

services of an informational nature are however essential for conducting transactions - for instance, balance inquiries might be needed before committing a money remittance. The accounting and brokerage services are therefore offered invariably in combination with information services. Information services, on the other hand, may be offered as an independent module.

Mobile banking may also be used to help in business situations as well as financial.

A. Account information

1. Mini-statements and checking of account history
2. Alerts on account activity or passing of set thresholds
3. Monitoring of term deposits4. Access to loan statements
5. Access to card statements
6 .Mutual funds / equity statements
7. Insurance policy management

B. Transaction

1. Funds transfers between the customer's linked accounts
2. Paying third parties, including bill payments and third party fund transfer.
3. Check Remote Deposit

C. Future functionalities in mobile banking

Based on the 'International Review of Business Research Papers' from World business Institute, Australia, following are the key functional trends possible in world of Mobile Banking.

With the advent of technology and increasing use of smart phone and tablet based devices, the use of Mobile Banking functionality would enable customer connect across entire customer life cycle much comprehensively than before. With this scenario, current mobile banking objectives of say building relationships, reducing cost, achieving new revenue stream will transform to enable new objectives targeting higher level goals such as building brand of the banking organization. Emerging technology and functionalities would enable to create new ways of lead generation, prospecting as well as developing deep customer relationship and mobile banking world would achieve superior customer experience with bi-directional communications. Among digital channels, mobile banking is a clear IT investment priority in 2013 as retail banks attempt to capitalize on the features unique to mobile, such as location-based services.

III. PAYMENT PROCESS

Payment process is used to describe the process and service that automates payments transaction between the customer and merchant. It is usually a third

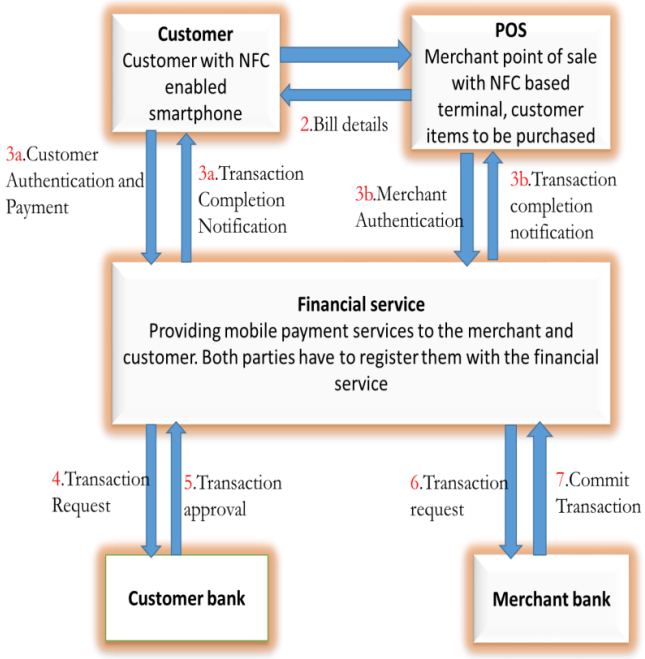


Figure 1 describes the payment process

party service that is actually a system of computer processes that process, verify, and accept of decline credit card transaction on behalf of the merchant through secure internet connection.

A. The Process – Payment Process

Step 1: Handshaking Customer arrives at point of sale counter after picking up groceries .Customer place it NFC enabled smart phone in front of the point of Sale NFC receiver with physical shopping mobile application being open. A hand shake (communication link formation) between. Customer application id and shop id is exchange between the two devices.

Step 2 Bill Details Customer Items are scanned on the POS and Customer application display list of items to be purchased by the customer along with total amount of the Bill.

Step 3 Authentication Customer and Merchant authentication is performed by the financial institution. Customer and merchant are authenticated via pin authentication number, which is verified at the financial institution.

Step 4 Transaction Request a transaction with the customer's authentication details shall be sent to the bank for account related matters.

Step 5 Transaction Approval Once it has been verified that the transaction can be made as per account details and bill requirements, the made and preserved.

Step 6 Transaction Request Same as step 4.

Step 7 Transaction Commit and Notification: Once the transaction is committed and preserved, the relevant notifications are relayed to the customer and the merchant.

IV. SYSTEM ARCHITECTURE

The experimental platform comprises the following components:

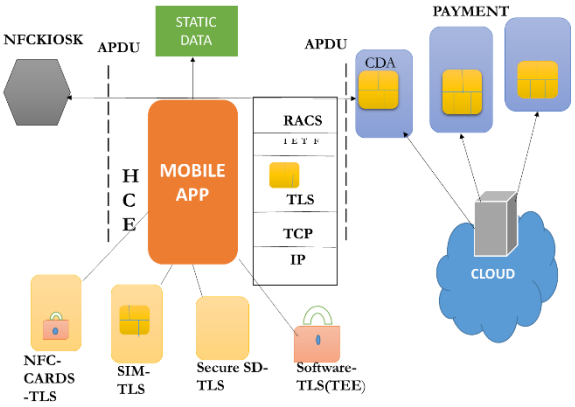
An android (whose version is 4.4 or later) mobile equipped with a NFC interface and supporting the HCE facility. When

the SIM is used as secure element the *OpenMobileAPI* package, specified by the *SIMAlliance* consortium is required. This library although not supported by all Android devices, is widely available. Its presence is associated to specific permissions, logged in dedicated public databases.

A secure and trusted TLS stack needed by the RACS protocol. The minimum security level should be a trusted storage and computing of the private key used for the RACS client authentication. In previous papers we detailed open java card implementations of TLS stacks in secure elements such as SIM, NFC controller or Secure SD. In section B we list secure elements or trusted devices that could be involved for TLS trusted computing.

- A RACS server hosting a NFC reader and a contactless genuine bank card. This implementation has been more precisely described in a previous paper.

A mobile application realizing the logical glue between the HCE interface and RACS server. As explained in section six some iso7816 requests dealing with static information are locally processed while EMV cryptographic commands (such as *Generate AC*) are forwarded to the RACS server.



-

Figure 2 mobile bank payment experimental platform

A. *Trust Model*

As we previously mentioned it, mobile terminal trust is the cornerstone for the *payment in the cloud* paradigm. Mutual authentication is performed thanks to the TLS protocol. Trust is enforced by TLS processing in secure elements or **Trusted Execution Environment** (TEE,) infrastructure. Most of secure elements are supporting a ***Java Virtual Machine* (JVM).** A java card TLS application size is about 20 Kbytes, and requires execution time ranging from 1-10 seconds for the TLS full mode and 0,2-5s for the TLS abbreviated mode.

In a mobile operator context the java card TLS stack runs in a SIM device. It can be downloaded according to the ***Over the Air* (OTA)** technology, or provisioned during the device manufacturing. According the *OpenMobileAPI* standard the hash of the certificate used for signature of authorized

Figure 3. A Secure SD card development kit

Another alternative is the use of a *SecureSD* memory including a secure element. These components are already available in the market and include a javacard device. Figure 5 shows a development kit for SecureSD components, with NFC interface.

External NFC Card is a secure element that can be used to boot a TLS session (see [28]), which is afterwards transferred to the mobile application (i.e. ephemeral session keys and their associated cipher suite). This technology works with all NFC mobiles. The activation of the remote bank card starts when the phone is tapped against this device.

Trusted Execution Environment (TEE,) is based on the *Trusted Zone* concept, in which a processor works according to two contexts the normal mode (used by the rich operating system) and the secure mode. Some Android operating systems, like Samsung KNOX, support this feature. TLS stack may run in this environment, which is resistant to malware attacks thanks to hardware isolation mechanisms.

B. *SECURITY MODEL*

Security in mobile payment system is the provision of confidentiality, integrity, authentication, authorization, assurance, and non-repudiation in every transaction. Security architecture can be defined as the design artifacts that describe how the security controls are positioned, and how they relate to the overall information system architecture.

Critical data involved in financial transaction must be stored securely in the mobile device or in issuer's storage infrastructure. Mobile payment security architecture examines the way security is built into mobile payment system architecture in order to achieve mobile payment security requirements.

Cryptographic key management helps to prevent the mobile payment system from being compromised by an attacker. The study of this security architecture will help to identify the existing security measures built into mobile payment system; assess how these measures are able to secure the system, and also provide insight to the vulnerabilities that still need to be mitigated.

The Figure 2 shows mobile payment security architecture with placement of security controls. We assume that the existing control is similar to what we have in traditional EMV payment architecture. Payment information provisioning and personalization processes between mobile device and issuer are protected based on Public Key Infrastructure (PKI) system using Secure Socket Layer version 3 (SSLv3) or Transport Layer Security (TLS). Transport layer security and its predecessor, the secure socket layer, are cryptographic protocols that provide secure communication for Card-Not-Present (CNP) transactions over the internet. SSL is used to provision the EMV card data to the mobile phone. Subsequently, Payment information is protected by the emulated EMV certificates and the EMV secret key provisioned into the secure element in the phone by the issuer . A Public Key Infrastructure (PKI) is a system consisting of set of hardware and software used for the management of public key and distribution of digital certificates which are used to verify that particular public key belongs to a certain entity.

The PKI creates digital certificates which map public keys to entities, securely stores these certificates in a central repository, and revokes them if needed when it is not in use [21]. Figure 2 shows how security controls are placed in mobile payment system architecture.

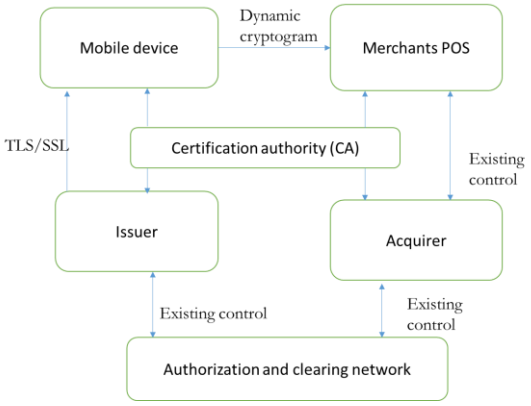


Figure 4 mobile payment security architecture

Certification Authority (CA) is trusted entity that issues digital certificates to users within a PKI system and provide status information about the certificates the CA has issued. The digital certificate certifies the ownership of a public key by the named subject of the certificate.

Both issuer and acquirer have their individual public key pairs; therefore generates digital certificates. The CA authenticates the public keys of both the issuer and acquirer. CA certifies the public key of the issuer using its private key.

The POS terminal retrieves its stored copy of the CA public key and used it to verify the issuer's public key certificate. Subsequently, the POS terminal also gets the issuer's public key from the issuer public key certificate and used it to verify the dynamically signed mobile payment data. The CA's public key is distributed to the acquirer and the POS terminal. POS terminal used the public key to verify that the issuer's public key was certified by the CA.

Mobile phone (emulated EMV card) authentication to merchant's POS terminal is similar to EMV card authentication. Dynamic Data Authentication (DDA), Combined Dynamic Data Authentication (CDA) and Fast Dynamic Data Authentication (FDDA) are authentication methods that can occur in mobile payment system. DDA makes each mobile payment transaction unique to protect payment data from customer phone to POS terminal.

For each transaction, the POS terminal requests that the mobile phone generate a cryptogram based on a random data element sent to it, a valid cryptogram is generated and verified when the transaction is authorized. This cryptographic value and transaction-specific data is validated by the POS terminal to protect against data breach. The mobile phone must be present to generate a valid cryptogram which is verified offline or online during transaction authorization stage.

Dynamic data authentication method used by mobile phone will lower payment fraud because stolen payment card information will not be used to make counterfeit cards or fraudulent online transaction. Dynamic cryptogram provided by issuer improves mobile payment security.

V. EXPERIMENTAL PLATFORM

We test the experimental platform with a genuine NFC payment card. Payment transactions were performed in France, over 3G/4G cellular networks, in shops or with vending machines.

A legacy contactless transaction consumes about 400ms, and requires eight ISO7816 requests, which are detailed below:

r1) Selection of the PPSE application.

r2) Selection of the NFC payment application.

r3) Issuance of the GPO command.

r4-r7) Four *Read Record* commands used for collecting four files located in two records.

r8) One GenerateAC request, realizing a CDA operation.

About 50% of the transaction time (200 ms) is consumed by the CDA computing.

The mobile application manages a cache; the seven first iso7816 requests, which return static information, are locally processed by the smartphone. Each operation needs about 30ms; therefore seven APDUs cost 210ms, which is nearly equivalent to the legacy transaction. The last request (*GenerateAC*) is forwarded to the remote server, which implies a delay ranging between 350 and 650 ms, according to the following repartition:

200 ms are burnt by the remote CDA operation

100-250 ms are spent by the platform components (mobile phone, server operating system and network components)

50-200 ms are consumed by the latency of 3G/4G cellular network

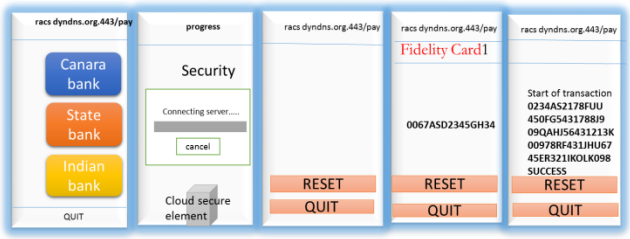


Figure 5. Mobile payment user's experiment

Figure 6 details the user's experience and introduces the mixing of fidelity procedures with the payment transaction.

The user starts the payment application and selects a payment card. A TLS connection is afterwards started with the RACS server. Upon success a RACS script performs the two following operations

Power on the selected bank card. selection of the payment application (r2 request).

At this step the mobile is ready for a payment transaction; the HCE NFC interface is unlocked.

A simple but very useful future feature is illustrated by figure 6. A virtual fidelity card is associated with an application AID registered with the payment application.

The merchant terminal selects this virtual card (via the dedicated iso7816 SELECT command) before the transaction. The returned information includes a card number to which the payment would be bound.

The merchant terminal performs the NFC payment procedure, iso7816 requests are processed locally until the reception of the *GenerateAC* APDU, which is forwarded to the server thanks to a RACS script in charge of the following operations:

Issuance of the GPO command (r3 request)

Execution of the *GenerateAC* command (r8 request)

Thereafter the result of the CDA procedure is returned from the server to the merchant terminal.

The use of the local cache is not mandatory. In the transparent mode every iso7816 request is forwarded to the server, what leads to an extra time cost of about 250ms (in average) per APDU (leading to a total duration of about 8x250+400= 2400ms).

VI. CONCLUSION

The mobile payment is a very attractive service, since it adds a new and useful application to mobile phones.

The focus is on the vulnerabilities in the mobile phone as the payment token and the merging multibank payment application.

From the user's point of view the security is increased, the mobile phone acts as a bank card equipped with a screen, the amount of the transaction may be displayed and the transaction is logged. Furthermore, in case of lost or stolen mobile, the access certificate (typically handled by a secure element) is revoked, but the bank card is still working.

From the business point of view, a new entity is created, the server that provides payment in the cloud. This is an opportunity for classical pay per use schemes, but also for new relationships between customers and cloud services including fidelity programs, advertisements, or information logging.

This paper demonstrates the technical credibility of an open model dealing with payments in the cloud. We hope to extend this first experiment with new partners, in order evaluate scalability issues and market opportunities.

REFERENCES

[1] Pascal UrienTelecom ParisTech23 avenue d'Italie, 75013, aris, FrancePascal.Urien@telecomparistech.fr

[2] https://www.banquefrance.fr/fileadmin/user_upload/banque_de_france/Economie_et_Statistiques/Bilan-cartographie-des-moyens-de-paiement-2014-donnees-2013.pdf

[3] http://www.fbf.fr/fr/files/87BCNH/Chiffres-cles-mdp-France-22012015.pdf

[4] Svigals, J.; Ziegler, H.A., "Magnetic-stripe credit cards: Big business in the offings", Spectrum, IEEE, 1974, Volume: 11, Issue: 12

[5] Smith, D.F.; Donnelly, T.; Mapps, D.J., "The credit card as a mass storage medium", IEE Colloquium on Document Image Processing and Multimedia Environments, 1995

[6] ISO/IEC 7813 Standard information technology -- Identification cards -- Financial transaction cards

[7] http://krebsonsecurity.com/2014/01/a-first-look-at-the-targetintrusion-malware/

[8] EMV Books, 1 - Application Independent ICC to Terminal Interface Requirement and Application Selection, Book 2 - Security and Key Management, Book 3 - Application Specification, Book 4 - Cardholder, Attendant and Acquirer Interface Require, www.emvco.org

[9] ISO 7816, "Cards Identification - Integrated Circuit Cards with Contacts".

[10] Jurgensen, T.M. ET. al., Smart Cards: The Developer's Toolkit, Prentice Hall PTR, 2002, ISBN 0130937304

[11] Z. Chen, "Java Card Technology for Smart Cards: Architecture and Programmer's Guide" Addison Wesley Longman, Inc., 2000

[12] Visa Contactless Payment Specification, Version 2.0.2, July 2006

[13] MasterCard World Wide Pay Pass - Mag Stripe, Technical Specifications, Version 3.3 December 2007