Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 7, July 2014, pg.888 – 904

RESEARCH ARTICLE

A Novel Interoperable Mobile Wallet Model with Capability Based Access Control Framework

Neeharika P [1, 2], V N Sastry[2]

[1]School of Computer and Information Science
University Of Hyderabad
Hyderabad, 500 046
Telangana, India
neeharika2290@gmail.com

[2]Mobile Banking Security Lab [MBSL]
Institute for Development and Research in Banking Technology
Road No.1, Castle Hills, Masab Tank
Hyderabad – 500 057
Telangana, India
vnsastry@idrbt.ac.in

Abstract— Initially mobile phones were used only for calls and messaging services. Nowadays almost all the basic utility devices around us have been replaced by mobile phones, ranging from simple alarm clock to controlling ubiquitous devices remotely. Mobile phones nowadays are much smarter compared to the devices used for payment processing in the early ages of banking. The plastic cards that we carry in our wallets like financial cards, membership cards, driving license etc all hold digital data. This gave inception to the idea of placing the plastic cards onto a mobile phone. There are a large number of mobile wallet initiatives currently. We have given the existing challenges that the current initiatives are facing. In this paper we have given a model for the development of a mobile wallet that can work across various platforms. Security is the major concern when it comes to finance related information. To address the security issues of our proposed mobile wallet model, we have also given an access control model that works with our interoperable mobile wallet in detail.

Keywords — Mobile Wallet, Digital Wallet, Access Control, NFC [Near field communication], Mobile Banking

---

I. INTRODUCTION

A Mobile Wallet is functionality on the mobile device that can securely interact with digitized valuables such as financial data, Identity or Mobile Commerce [1]. Mobile wallets can be looked at as a level above the traditional mobile banking applications. Some of the functions of a mobile wallet can be given as,

1. Banking Services
2. Loyalty Card management
3. Storing offers
4. Utility Bills
5. Online payment services [E-commerce]
6. Provide transaction based Statistics

Currently there are over 150 mobile wallet initiatives in the world [2]. Google Wallet and passbook were launched by the tech titans Google and Apple. The mobile digital wallet has not only grabbed the attention of such technology based industries but also payment providers such as PayPal, commerce giants such as Amazon, telecom operators like Airtel [Airtel money], Vodafone [mpesa] to mention a few. However it has not yet made the impact that was expected out of it and a lot of developments and initiatives have been going on in that direction.

Payment channels that the mobile wallets can work with are, Proximity based or NFC [Near field communication], GPRS, SMS, USSD and Bluetooth.

Majority of the mobile wallets focus on NFC and GPRS based payments. In developing countries, it would be more appreciated if they can cater payments based on low end channels such as SMS and USSD.

We begin the discussion by looking at the existing challenges in the industry. We will look into the proposed model along with the algorithms to support the model. To cover the security aspects of the model, we have also proposed an access control model for the interoperable mobile wallet.

II. RELATED WORK

Stig Frode Mjolsnes and Chunming Rong, 2001 [3], have given an architecture which protects the sensitive user credentials such as a bank cards details. The problem was addressed by the introduction of a local credential manager who is responsible to take care of all the user credentials.

Ernst-Joachim Steffens, Axel Nennker et al, 2009 [4], have given an architecture to combine the plastic cards with the easy and convenient mobile network and NFC. They have discussed the requirements of technology used and its implementation. They explained the working of a mobile wallet based on the secure element enabled SIM and the UICC (Universal Integrated Circuit Card).

Alan cole, Scott McFaddin et al, 2009 [5], have explained the eco system of a mobile wallet, the roles and responsibilities of the players involved. They have also provided the design considerations for the development of a mobile digital wallet and also explained the prototype launched.

Hao Zhao and Sead Muftic, 2011 [6] have given a design for the implementation of a secure mobile wallet. The wallet data is stored securely on the secure element provided in the secure element enabled SIM and UICC.

III. EXISTING CHALLENGES

1. Security of the sensitive data when it is OTA[over the air] and also when it is stored
2. Interoperability across Financial institutions, Mobile network operators, Card associations, Payment processors and retailers.
3. Hardware restrictions [Need for secure element]
4. Lack of proper standards
5. Usage Restrictions
6. Ownership of the wallet
7. Trusted and well established organization as wallet host in order for it to be adopted across a substantial number of partners and consumers.
8. Do not work with rooted devices, which is a problem for developers.
9. Availability of the wallet services globally
10. Mobile operating systems it can work with
11. Adoption of NFC worldwide
12. Support for a multitude of payment channels from high end proximity based payments to Bluetooth to low end SMS/USSD based
13. Cluster of mobile wallets separately by various organizations, which reduce overall usage. Customers would want to use one mobile wallet across all platforms rather than being forced to use different mobile wallets for different purposes.

IV. PROPOSED MODEL FOR INTEROPERABLE MOBILE WALLET

Our proposed model is a cloud based wallet. It does not store any data onto the device. Every user has a unique wallet id and a default wallet account. Payments can be made directly using cards or through wallet account. It supports the following functions,

1. Bank Cards
    a. Addition of cards onto wallet
    b. Deletion of cards from wallet
    c. Balance verification
2. Loyalty card management
    a. Addition of cards onto wallet
    b. Deletion of cards from wallet
    c. Storing offers
    d. Balance verification
3. Utility bills
4. Statistics
5. Wallet
    a. Funds transfer
    b. Check Balance

We have given algorithms for the working of each of these above services. We have also incorporated access control layer to the model and given a set of algorithms for control the access to the sensitive data stored on the TSM cloud.

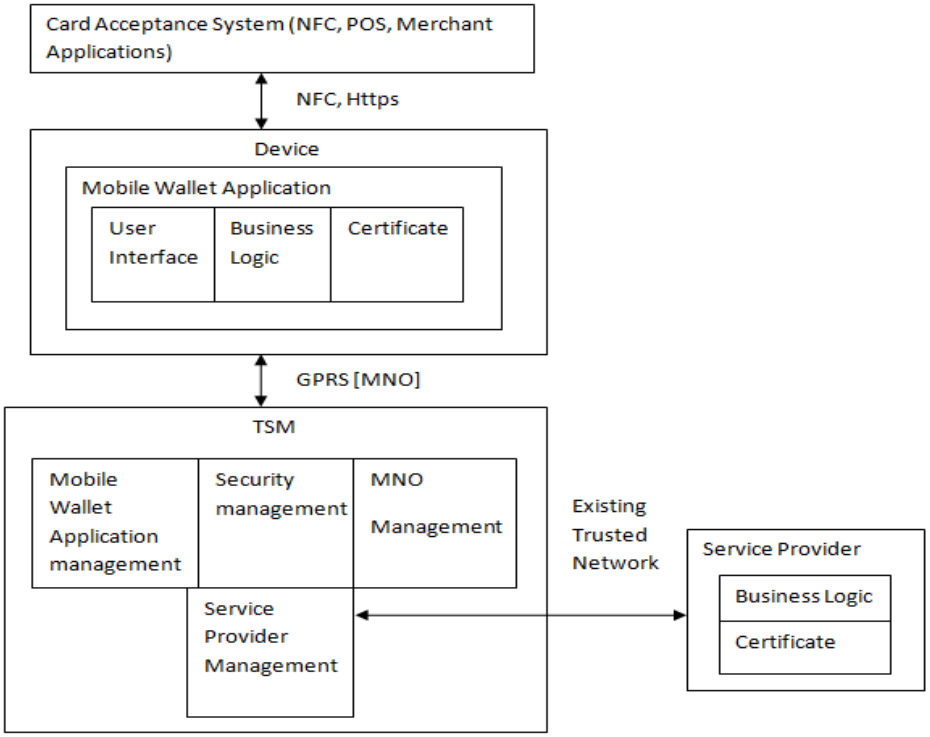

Fig 1: Proposed model for interoperable mobile wallet

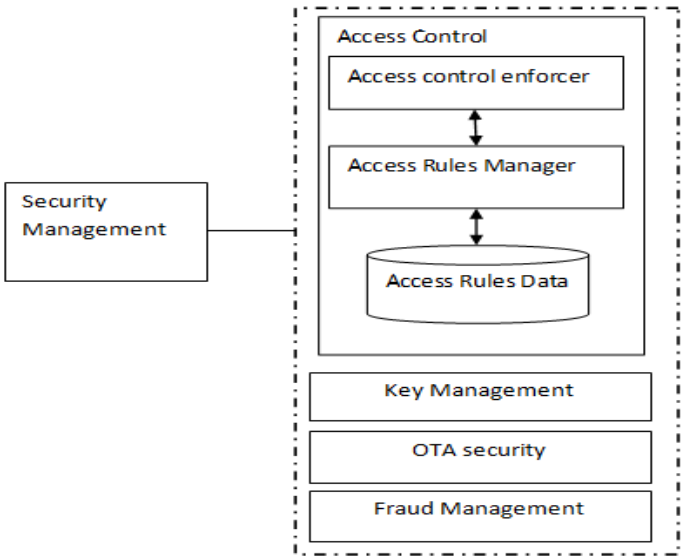

Fig 2: Security management module

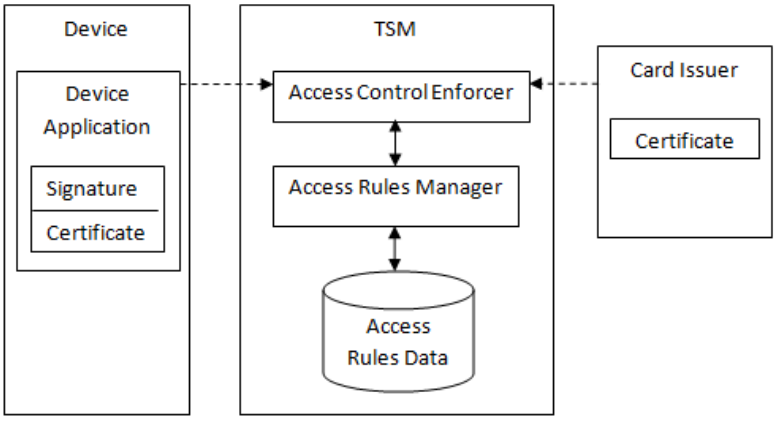

Fig 3: Supporting access control model

TSM [Trusted service manager]: This is the most critical component of this framework. The entire management of the interoperable mobile wallet lies within the scope of TSM. All the other entities; Devices, Mobile network operators, Service providers are connected only to the TSM. They need to have the contractual agreements established only with the TSM and can work with all the other entities seamlessly without having to know about the organizations at the other end. The inclusion of TSM is what introduces the concept of interoperability omitting any restrictions based on compatibility as TSM creates a common platform for a variety of parties to work with. The concept of TSM is similar to aggregators but with added functionality [7].

A. Roles and functions of TSM

1. Authors are free to extend the main body text and sections as appropriate with suitable section/subsections. Do not include unnecessary spaces Contractual Relations and establishment of Business Rules with all the parties involved such as MNO, Service providers, Retailers, Billing Organizations
2. Billing & Reporting for all the partners
3. Secure end to end connections for all transactions
4. Registration across all partners
5. Wallet data preparation and staging
6. Wallet Hosting
7. Wallet content management

8. Data centre hosting
9. Fraud Management
10. Disaster recovery
11. End Customer Support
12. Application testing and Certification
13. Key management
14. Access Control, Physical and digital
15. OTA Provisioning
16. Call centre

Interoperability can be achieved only when the TSM is chosen in such a way that it can connect to a huge variety of stakeholders. In our analysis of the existing mobile wallets it was observed that, the mobile wallet offers are too fragmented and the competition across entities has even led to new restrictions in terms of interoperability. Our suggestion would be to make a trusted institution of the country such as the central bank the TSM of the country and establish common standards across a nation. Only then would complete interoperability be achieved.

B. Security aspects of the interoperable mobile wallet

The security needs of the mobile wallet can be grouped as follows,

Security on the device: This module takes care of the security of the application, user and the data stored onto the device.

Tampering the application: Application may be altered with the purpose of using it as a vehicle for malicious activity if it is not secured properly. Every application irrespective of the mobile operating system it is built to work on has a certificate corresponding to it. TSM signs the application's certificate using its private key and places the signature along with the application. At the time of transaction, the signature can be verified to see if the request is originating from the right application.

Using reverse engineering in case of android applications the application source can be extracted from the application file. Developers have to make sure that all the vulnerabilities that lead to reverse engineering are closed.

User level security: Authentic users should only be able to access the data corresponding to them. The user authentication and authorization is placed within the access control module of the TSM which will be discussed later.

Data stored on the device: The credentials entered by the user at the time of registration are encrypted immediately and no where the data is stored in its raw form. For the encryption of data, the keys are to be placed onto the device. Security of these keys is to be ensured. The non sensitive information that is being stored in the device can be sandboxed to the application as in majority of the mobile operating systems.

Security of data OTA (Over the air): The sensitive data is to be transmitted over the air. Security keys are to be provided at both the ends to prevent various attacks. The data transmitted is to be encrypted end to end so that even if it is captured over the air by any means its security is not compromised.

Security of the data at TSM: This is the biggest challenge in the entire implementation of the mobile wallet. A huge amount of sensitive data is held by the TSM. The data is collected from various financial and retail sectors and there should not be any loopholes in their security.

Security at TSM can be provided by using proper access control mechanism for the data, strong security for the database server; physical and logical, fraud prevention and detection, proper backup of data to avoid denial of service and disaster recovery mechanisms.

C. Algorithms for the development of various modules

Notations:

IMEI: International Mobile Station Equipment Identity
TSM: Trusted Service manager
MNO: Mobile network operator
NFC: near field communications
OTA: over the air transmission
SP: Service provider
APP_CERT: Certificate of the issued wallet application
UID: Wallet account id for the user
ERROR_CODE: Every error is associated with a code and an explanation. An appropriate error code can be sent at the occurrence of an error

ACK: Acknowledgement, varies according to the transaction
SK_{UT}: Shared secret key between user and TSM for OTA transmission.
SK_{ST}: Shared Secret key between TSM and service provider for OTA transmission. [We assume a key agreement protocol has been in use for symmetric key distribution]
PK_{SP}: Public key of the service provider
PrK_{SP}: Private Key of service provider
PK_{TSM}: public key of TSM
PrK_{TSM}: Private Key of TSM
UN: Username chosen by the user at the time of registration.
PW: Password chosen by the user at the time of registration.
IMG: An agreed image between the TSM and user at the time of registration. This is used for the identification of the server.
MSG: An agreed message between the TSM and user at the time of registration. This is used for the identification of the server.
Sign: {APP_CERT} PrKTSM
CID: unique card id that is stored on the device which can be used to identify the card.
SPID: unique id for the service provider that is stored on the device which can be used to identify the card.
ACE: Access control enforcer
ACM: Access control manager
AR: access rules
ARF: Access rules file
OBJ: Object the capability refers to
OPS: operations that the capability corresponds to

Methods:
GET: This method is used to receive the data from server/Device
POST: This method sends the data to the server/Device
VERIFY: Validates the data given is matching with the existing data.
SELECT: Locates the data
CONCAT: combine one or more entities into one.
STORE: insert an entry into database
DELETE: remove a row from database
ENTER: Feed data into a form

Data structures:
WALLET_USER: Database of the card numbers and user details who is using wallet services, present at the service provider. Service provider can store all the transactions done by the user on the particular card using wallet.
CARDS: present in the device and stored the non sensitive information of a card which can be used to identify the card among users and TSM.
CARDS_TSM: present in the TSM and stores card data. [Excluding CVV and expiry date]
USERS: database consisting of user wallet account details. Present in the TSM
TXN: Holds the transactions of various users using different cards
USERS_CAP_TOKEN: contains the Capability_token corresponding to the users. Present in ACE
USERS_CAP: contains the capabilities corresponding to the Capability_token of the users. Preset in ARF
SP_CAP: contains the capabilities corresponding to the Capability_token of the SP. Present in ARF
CAP_DEF: Contains the definitions of individual capabilities. Present in ARF

Working of the Model:

The steps involved in the working of a module and the various modules involved from the perspective of a user.

Step 1: Install the wallet application
Step 2: User Registration
Step 3: User Login
Step 4: While (User Logged in)
Choose a Service
{Credit/Debit Cards, Loyalty Cards, Statistics, Pay Bills, Wallet}
If (Credit/Debit Cards)
Choose a Service
{Add Card, Delete Card, Check Balance}
If (Loyalty Cards)
Choose a Service
{Add Card, Delete Card, Check Balance, Check offers}
If (Statistics)
Statistics
If (Pay Bills)
Choose bill
Payment of Bills
If (Wallet)
Check Balance
Funds Transfer {Load Wallet, Transfer to another wallet}

Working at TSM with respect to access control:

Step 1: User Login [Mutual Authentication]
Step 2: Dynamic session key Generation and Verification
Step 3: If (Service Required)
Invoke ACE
If (Access Grant)
Provide Service
Else ERROR_CODE

ALGORITHM 1: USER REGISTRATION

The user has to register for the wallet service on first use. Only one user can register on one device as IMEI of the device is used for the identification of user. Once registered the option for registration will no longer be available for the user. User also selects an authentication image and message during registration. This lets him identify the server.

Objective: To create an account for the user

Inputs: Name, Address, Phone Number, Username, Password, IMEI, APP_CERT, Signature, IMG, MSG
Output: Entry for user in USERS
Step 1: User: ENTER Name, Address, Phone Number, Username, Password, IMG, MSG
POST {Name, Address, Phone Number, Username, Password, IMEI, IMG, MSG, APP_CERT, Signature} SK_{UT} to TSM
Step 2: TSM: IF (VERIFY APP_CERT, Signature)
Generate UID
STORE Name, Address, Phone Number, Username, Password, IMEI, IMG, MSG, UID in USERS
POST {UID} SK_{UT} to Device
Else POST ERROR_CODE to Device

ALGORITHM 2: USER LOGIN

Objective: Mutual authentication of User and TSM and establishment of session ID for the user.

Notation:
SALT: Random number added to UID for dynamic session keys
UID_D: UID for the next transaction request

Inputs: UN, PW, IMEI
Output: UID_D
Step 1: User: ENTER UN
POST {UN, IMEI, Sign, APP_CERT} SK_{UT} to TSM
Step 2: TSM: IF (VERIFY Sign, APP_CERT)
IF (VERIFY UN, IMEI in USERS)
SELECT IMG, MSG from USERS
POST { IMG, MSG} SK_{UT} to Device
Else POST ERROR_CODE to Device
Else POST ERROR_CODE to Device
Step 3: Device: IF (VERIFY IMG, MSG)
POST { PW, IMEI, Sign, APP_CERT} SK_{UT} to TSM
Else POST {ERROR_CODE } SK_{UT} to TSM
Step 4: TSM: IF (VERIFY Sign, APP_CERT)
IF (VERIFY PW, IMEI in USERS)
Generate SALT
UID_D : CONCAT{UID ,Hash{CONCAT(UID,SALT)}}
POST { UID_D } SK_{UT} to Device
Else POST ERROR_CODE to Device
Else POST ERROR_CODE to Device

ALGORITHM 3: ADDITION OF CARD

Objective: Verify the card details with the corresponding card association, bank and add the card to the user's wallet account

Notations
UID_D : Session key for that transaction
ACK_A: Accept/ Reject the card between user and TSM
ACK_B: Accept/ Reject the card between TSM and service provider

Inputs: Card Number, CVV, Expiry date, Bank, Branch, Card holder name, Card Association, UID_D
Output: Addition/ Rejection of card
Step 1: User: POST { UID_D ,{CVV, Expiry date}PK_{SP}, Card Number, Bank, Branch, Card holder name, card association} SK_{UT} to TSM
Step 2: TSM: POST {{CVV, Expiry date}PK_{SP}, Card Number, Bank, Branch, Card holder name, card association} SK_{ST} to SP
Step 3: SP: {{CVV, Expiry date}PK_{SP}}PrK_{SP}
IF(VERIFY Card Number, Expiry date, Bank, Branch, Card holder name, card association, CVV)
Add Card to WALLET_USER
POST {ACK_B, Balance, Offers} SK_{ST} to TSM
Else POST { ACK_B } SK_{ST} to TSM
Step 4:TSM: IF (ACK_B == "Accept")
Generate CID
STORE UID, CID, Card Number, Bank, Branch, Card holder name, card association, Balance, Offers
POST {CID, New UID_D, ACK_A} SK_{UT} to Device
Else POST {ACK_A, New UID_D } SK_{UT} to Device
Step 5: Device: IF(ACK_A == "Accept")
STORE CID, Bank, Branch, card association in CARDS
STORE New UID_D

ALGORITHM 4: DELETION OF CARD

Objective: Verify the card CVV with the corresponding bank and delete the card from the user's wallet account

Notations

UID_D : Session key for that transaction

ACK_A: Accept/ Reject the deletion of card between device and TSM

ACK_B: Accept/ Reject the deletion of card between TSM and service provider

Inputs: CID, CVV, Expiry date

Output: Deletion / Reject deletion

Step 1: User: Choose card to delete
Enter CVV, Expiry date of the card
POST { UID_D ,CID, {CVV, Expiry date }PK_{SP} } SK_{UT} to TSM

Step 2: TSM: POST {{CVV, Expiry date }PK_{SP}, Card Number} SK_{ST} to SP

Step 3: SP: {{CVV, Expiry date }PK_{SP}}PrK_{SP}
IF(VERIFY Card Number, Expiry date, CVV)
DELETE Card from WALLET_USER
POST ACK_B to TSM

Step 4: IF (ACK_B == "Accept")
DELETE Card entry from CARDS_TSM
ACK_A = "Accept"
Else ACK_A = "Reject"

Step 5: POST {ACK_A, New UID_D } SK_{UT} to Device

ALGORITHM 5: VERIFICATION OF BALANCE

Objective: Retrieve wallet/ Card balance

Notations:

CID: Unique card Id. For Wallet balance, CID == UID

Inputs: CID, UID_D

Output: Balance

Step 1: User: Choose account whose balance is to be enquired
POST {CID, UID_D } SK_{UT} to TSM

Step 2: IF(VERIFY UID_D)
POST {Balance, New UID_D} SK_{UT} to Device
Else {ERROR_CODE, New UID_D } SK_{UT} to Device

ALGORITHM 6: USER'S PAYMENT TRANSACTION STATISTICS

Objective: Generate Retailer based, Card based or Expenditure based statistics

Notations:

EXP_{MIN} - EXP_{MAX} : Expenditure range required for statistics

Inputs: Type, Period, SPID, CID, UID_D
Outputs: Statistical graphs
Step 1: Device: Choose type of statistics
Choose Period
Step 2: Device: Switch(type)
Case Retailer Based:
Choose Retailer
POST {type, SPID, Period, UID_D } SK_{UT} to TSM
Case Card Based:
Choose card
POST {type, CID, Period, UID_D } SK_{UT} to TSM
Case Expenditure Based:
Enter Range of Expenditure
POST {type, CID, EXP_{MIN}, EXP_{MAX}, Period, UID_D } SK_{UT} to TSM
Step 3: TSM: IF(VERIFY UID_D)
Switch(type)
Case Retailer Based:
SELECT Bills from TXN for SPID During Period given
POST {Bills, New UID_D} SK_{UT} to Device
Case Card Based:
SELECT Bills from TXN for CID During Period given
POST {Bills, New UID_D} SK_{UT} to Device
Case Expenditure Based:
SELECT Bills from TXN in the range EXP_{MIN} to EXP_{MAX} during Period
POST {Bills, New UID_D} SK_{UT} to Device
Else POST {ERROR_CODE, New UID_D } SK_{UT} to Device
Step 4: Device: Display different graphs for Bills

ALGORITHM 7: TRANSFER OF FUNDS

Objective: Load wallet balance, Transfer of funds between wallets

Notations:

SID: Account Id of the sender
RID: Account Id of the receiver
UID_R: Username of the receiving wallet
CONF: Yes if receiver agrees/ No

Inputs: SID, RID, Amount, type, CVV, Expiry date, UID_D
Outputs: Transfer of balance from one account to another
Step 1: User: Enter Amount
Choose the type of transfer
Step 2: User: Switch(type)
Case LOAD_WALLET:
Choose card to transfer from
SID = Chosen CID
RID = UID
POST {type, SID, RID, Amount, UID_D, {CVV, Expiry date }PK_{SP} } SK_{UT} to TSM
Case WALLET_TO_WALLET:
Enter Username of receiving wallet
SID = UID

RID = UID_R
POST {type, SID, RID, Amount, UID_D } SK_{UT} to TSM

Step 3: TSM: IF(VERIFY UID_D)
Switch(type)
Case LOAD_WALLET:
POST {Card Number, Amount,{CVV, Expiry date }PK_{SP} } to Bank
Bank: {{CVV, Expiry date }PK_{SP}}PrK_{SP}
IF(VERIFY Card Number, Expiry date, CVV)
IF((BAL-BAL_{Min}) >= Amount)
UPDATE BAL
POST {ACK_B, BAL} SK_{ST} to TSM
Else POST {ACK_B = "Low Balance"} SK_{ST} to TSM
Else POST {ACK_B = "Invalid credentials"} SK_{ST} to TSM
IF (ACK_B == "Accept")
UPDATE Balance in CARDS_TSM & USERS for SID & RID
POST {ACK_A, Balance, New UID_D } SK_{UT} to Device
Else POST { ACK_A , New UID_D } SK_{UT} to Device
Case WALLET_TO_WALLET:
IF (SID.BAL >= Amount)
IF(VERIFY RID)
NOTIFY RID
IF(CONF)
SID.BAL = SID.BAL - Amount
RID.BAL = RID.BAL + Amount
Else POST {ACKA = "Receiver declined", New UID_D} SK_{UT} to Device
Else POST {ACKA = "no such receiving account", New UID_D } SK_{UT} to Device
Else POST {ACKA = "Insufficient wallet balance", New UID_D } SK_{UT} to Device
Else POST {ERROR_CODE, New UID_D } SK_{UT} to Device

D. Payment Models

Payments can be done using NFC, Initiated at e-commerce website or they can be pre registered within the mobile wallet.

NFC Based Scenario:

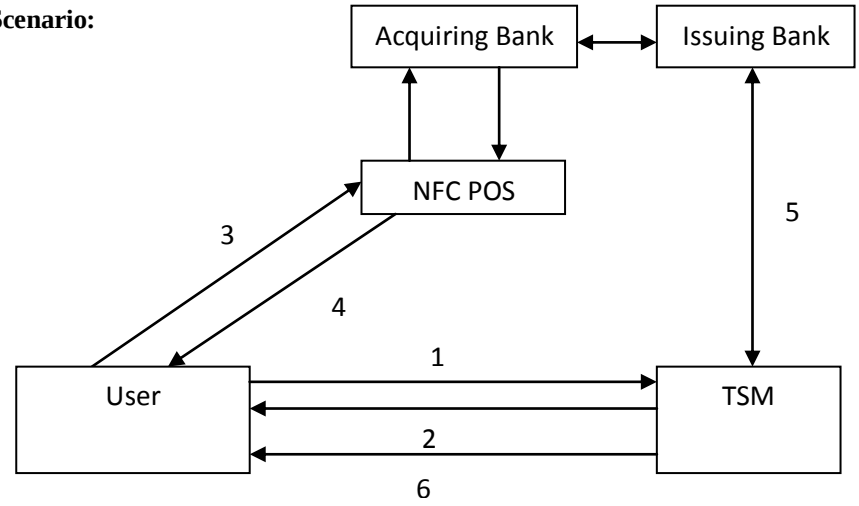

Fig 4: Payment model using NFC

Step 1: User chooses the card that he wants to pay with and requests the details of the card from TSM
Step 2: TSM returns card details and prompts user to enter his CVV and expiry date

Step 3: User taps at the NFC POS terminal and the card details are transferred to the POS along with the offers which could be applied for the transaction if any exist. User enters a 4-digit PIN at the POS as with the traditional card payments
The communication between POS, acquiring bank and issuing bank is done using the existing financial networks and is out of the scope of this paper.
Step 4: On successful payment POS prints a physical receipt for the transaction which is handed to the user
Step 5: Issuing bank send a receipt to the TSM regarding the transaction
Step 6: TSM enters the transaction in the users wallet account and pushes the receipt onto the mobile device for conformation

E-commerce website:

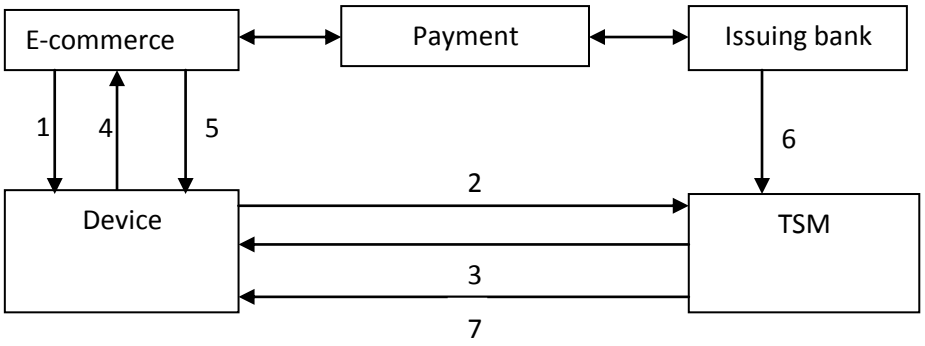

Fig 5: Payment model through e-commerce website

Step 1: User selects the pay with wallet option at the merchant site. The site redirects to the wallet application on the user's device
Step 2: User logs into the mobile wallet chooses card to pay and requests the card details from the TSM
Step 3: TSM returns card details and prompts user to enter his CVV and expiry date
Step 4: The card details are redirected to the merchant website from the wallet along with the offers which could be applied for the transaction if any exist.
The payment processing between the merchant site and issuing bank is proceeded with the existing secure financial network and is beyond the scope of this paper.
Step 5: Upon successful payment the merchant site proceeds with the order, generates the digital invoice and displays it to the user
Step 6: Issuing bank send a receipt to the TSM regarding the transaction
Step 7: TSM enters the transaction in the users wallet account and pushes the receipt onto the mobile device for conformation

Registered bills:

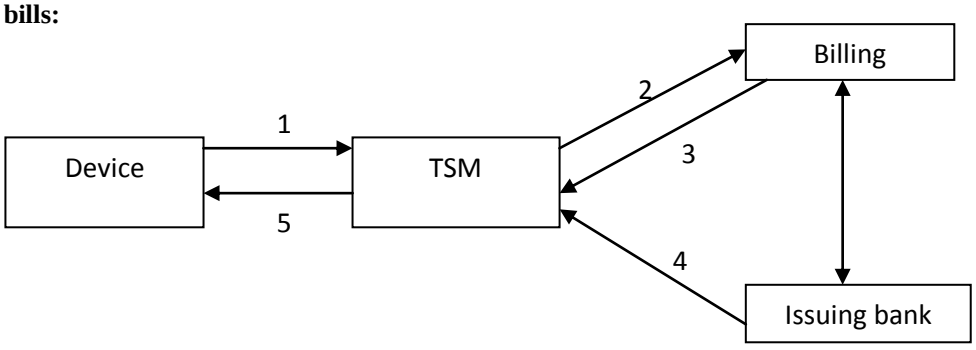

Fig 6: Payment model for pre registered bills

Step 1: User selects the bill that he would like to pay and the card he wants to use for the payment. He sends the payment request to TSM along with the CVV and expiry date of the card which is encrypted with banks public key. TSM and billing organization cannot see the CVV and expiry date

Step 2: TSM forwards the payment request to the appropriate billing organization along with the encrypted card details

Payment processing between billing organization and issuing bank happens through the existing secure financial network and is beyond the scope of this paper

Step 3: Upon successful payment the billing organization clears the user's bill and sends a confirmation to TSM.

Step 4: Upon successful payment issuing bank sends a bill invoice to TSM.

TSM clears the bill from user's bills and add the transaction to his wallet account upon receipt of confirmation from billing organization and issuing bank

Step 5: TSM send a confirmation message and invoice to the customer

E. Proposed Access Control Model

Access control is the process of mediating every request to resources and data maintained by a system and determining whether the request should be granted or denied [8].

This model provides access control for mobile wallet model proposed. It supports access management by TSM and allows each entity to access its data and ensures the security of the content.

Access Control Enforcer:

The role of the access control enforcer is to,

1. Identify and authenticate a user/Service Provider
2. Verify if the user/service provider is authorized to perform the requested action.

Identification and Authentication:

a) User identification is shown in the Login module mentioned above.
b) Service provider identification is done based on certificates
   i. TSM maintains a list of trusted certificates of the SP and also certificate revocation list which is synchronized with the information at corresponding certifying authority
   ii. When a request arises from SP, the certificate of the SP signed with the private key of SP is sent to the TSM.
   iii. TSM verifies the signature, checks the certificate with the list of trusted certificates and accordingly grants access

Authorization:

After the authentication of the entities, it is to be checked if they are authorized to perform the given action. Capability lists are used to validate authorization. Capability lists are subject rights that are provided as a mapping to objects [9]. The reasons for choosing capability lists are,

1. Finer grained Control
2. Follows principle of least privilege
3. Superior revocation facilities on a per subject basis

ALGORITHM 8: ACCESS CONTROL ENFORCER

Objective: Grant/Deny access

Notations:
REQ: request sent by the user/service provider

Input: REQ
Output: Grant/Deny access
Step 1: Fetch Capability_token
Step 2: POST {Capability_token, UID, SPID, OBJ, OPS } to ACM
Step 3: Fetch AR from ACM
Step 4: IF (REQ is in AR)
Grant Access
Else Deny Access

ALGORITHM 9: RETRIEVING CAPABILITY_TOKEN

Objective: Fetch the Capability_token for the user
Inputs: UID_D, SALT Outputs: Capability_token Step 1: IF (VERIFY UID_D using SALT) Fetch corresponding UID SELECT Capability_token from USERS_CAP_TOKEN for UID Else Deny

Access control manager

The number of parties and the data involved maybe a very large and hence one access rules data file may not suffice to the requirements of the system. Access Rules Manager module is introduced to route between access control enforcer and the corresponding access rules data file. The access control enforcer need to connect only with the access rules manager and further challenges such as

1. Locating the files that have the corresponding access rules
2. Fetching the respective access rules from the files are delegated to this module.

The whole management of the Access rules data lies within the scope of access rules manager. It can choose various indexing techniques for locating the files more efficiently.

Objective: Fetch the corresponding access control rules from the access control rules data.
Inputs: Capability_token, UID, SPID, OBJ, OPS **Outputs:** Rules **Step 1:** IF (UID != 0) Select USERS_CAP for UID Else Select SP_CAP for SPID **Step 2:** FOR(Capability_token, OBJ, OPS in USERS_CAP /SP_CAP) IF(capability.OBJ == OBJ && capability.OPS == OPS) Select capabilities **Step 3:** FOR(capabilities in CAP_DEF) Select Rules Return Rules

Access Rules File

Access rules define the actions that can be performed on possession of a given capability.
Subjects: Users & Service Providers
Objects: Card data & Transaction information
Assumptions: RDBMS [Relational database management system] is used for the storage of data to be protected.

Sample Cards data

Table CARDS
{
PAN [Primary account number] VARCHAR [16],
CardHolderName VARCHAR [45],
Validity VARCHAR [4],
Bank VARCHAR [45],
Branch VARCHAR [45],
UID VARCHAR [10],
SPID VARCHAR [10],
CardType VARCHAR [45],
Balance VARCHAR [10],
Offers VARCHAR [200]
}

Sample Transaction data

```
Table TXN
{
BillOrg VARCHAR [45],
PAN VARCHAR [16],
Bank VARCHAR [45],
UID VARCHAR [10],
SPID VARCHAR [10],
Amount INT,
Date DATE
}
```

Sample Access rules:

1. Users (PAN, CardHolderName, Validity, Bank, Branch, CardType)
   a) Add to Cards
   b) Find user specific cards (UID)
   c) Delete user specific cards (UID)
   d) Find his transactions

2. Service Providers (PAN, CardHolderName, Validity, Bank, Branch, CardType,Balance, Offers)
   a) Find cards corresponding to the Service provider
   b) Update only the offers & balance in Cards Table
   c) Find data related to its corresponding transactions on Transaction database

Each Capability_token corresponds to various capabilities that can be found in USERS_CAP and SP_CAP files for Users and service providers respectively

Example: Capability_token [User] -> {ADD_CARD_RES, FIND_CARD_RES, DELETE_CARD, FIND_TXN_RES}

a) ADD_CARD_RES: capability corresponds to the restricted addition of cards into CARDS
b) FIND_CARD_RES: capability corresponds to the restricted selection of cards from CARDS
c) DELETE_CARD: capability corresponds to the deletion of a card entry from CARDS
d) FIND_TXN_RES: capability corresponds to the restricted selection of transactions from TXN

Sample Data Structures for Capabilities [CAP_DEF]

```
Struct Capability
{
String OBJ;
String OPS;
String[] Access = {Object attributes list on which the operations can be done};
}
```

Example:

```
Struct ADD_CARD_RES
{
String OBJ = "Cards";
String OPS = "Insert";
String[] Access = {Cards.PAN, Cards.CardHolderName, Cards.Validity, Cards.Bank, Cards.Branch, Cards.CardType };
}
```


Proposed model for access control depicting the data exchanged and steps

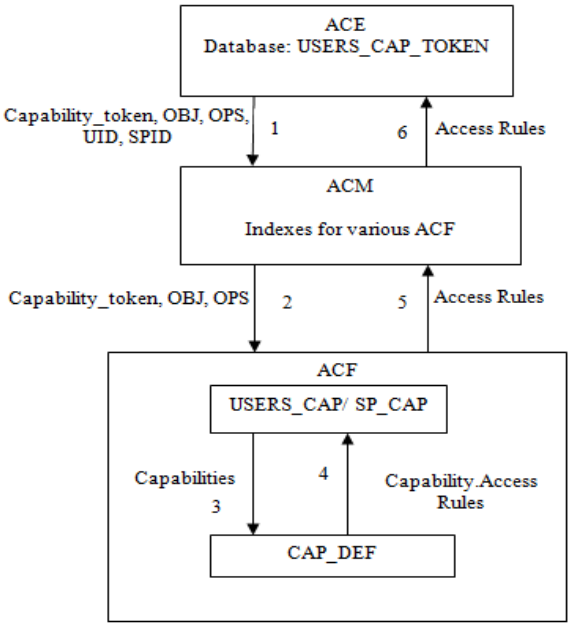

Fig 7: Proposed model for access control depicting the data exchanged and steps

V. SIGNIFICANCE OF THE PROPOSED MODEL

1. It is a cloud based wallet avoiding device level restrictions.
2. NFC can be used for payments [Proximity based] although any mobile device that can support GPRS works well with the model [Remote]. It is flexible to incorporate SMS/USSD/Bluetooth also in future
3. The sensitive data is shifted from the device. User need not worry about the loss of data in the case of theft. His data is securely stored with the TSM
4. CVV and expiry date are not stored anywhere, so even in the case of breakage of centralized server, data remains safe
5. *One connection many services:* The users, retailers, mobile network operators and financial institutions are only concerned with their connections to the TSM and they will be automatically connected to all the other entities.
6. *Interoperability:* Customer need not maintain different applications for financial institutions and retailers all can be clubbed into one platform. Our interoperable mobile wallet is mobile network operator, retailer, financial institution and payment processor agnostic unlike the existing wallets.
7. Scope of the wallet is very wide as it can combine utility payments, online payments, POS payments and banking services all into one application.
8. Cryptographic tools are incorporated to provide end to end security. We have used symmetric keys but PKI [Public key infrastructure] can also be used with the model.
9. The CVV and Expiry date of the card are not stored by the TSM and they are encrypted end to end between the user and the respective financial institution.
10. Direct to cardholder marketing
11. Capability lists are used to provide finer grained superior access control
12. All the access control responsibilities are in the hands of a central owner [TSM]. Hence no transfer of ownership or modifications in capabilities by any other entities
13. Dynamic session keys are generated for authentication which avoid various attacks
14. The inclusion of a random number which is known only to the TSM and not known to the user can solve the issue of transfer of capabilities
15. The model is flexible to changes dynamically
16. It provides principle of least privilege
17. Flow of information is controlled
18. Two factor authentication is provided during login to avoid phishing attacks

VI. CONCLUSION

Development of the mobile wallet model as mentioned above is still quite a challenge technically as well as in terms of business logic. Achieving complete interoperability requires a lot of changes in the usually not flexible industries. However the market potential of the wallet that is fully interoperable is extremely promising. We have started a path towards the development of a mobile wallet that could change the face of mobile payments all over. In the long run, our attempt towards a secure and interoperable mobile wallet could let customers get rid of the plastic cards in their wallet and also lead to a wider acceptance of mobile wallets globally.

REFERENCES

[1] "Control Points in Mobile Wallet", Mobey forum white paper, [Online] Available: http://www.mobeyforum.org/Knowledge-Center/Mobey-White-Papers [Feb 2012]
[2] http://www.ericsson.com/res/thecompany/docs/press/media_kits/mobile_wallet.pdf
[3] Mjolsnes, S.F.; Chunming Rong, "Localized credentials for server assisted mobile wallet," Proceedings for International Conference on Computer Networks and Mobile Computing, 2001 , pp.203-208
[4] Steffens, E-J., et al. "The SIM-based mobile wallet", 13th International Conference on Intelligence in Next Generation Networks, ICIN. IEEE, 2009. pp.1-6
[5] Cole, Alan, et al. "Toward a Mobile Digital Wallet" Vol. 16. IBM Research Report, 2009.
[6] Zhao, Hao, and Sead Muftic. "The concept of secure mobile wallet." World Congress on Internet Security (WorldCIS), IEEE, 2011.pp.54-58
[7] "Inside the Mobile Wallet: What It Means for Merchants and Card Issuers", First data white paper, [Online] Available: http://files.firstdata.com/downloads/thought-leadership/MobileWalletWP.pdf [2012]
[8] Samarati, Pierangela, and Sabrina Capitani de Vimercati. "Access control: Policies, models, and mechanisms." *Foundations of Security Analysis and Design*. Springer Berlin Heidelberg, 2001. 137-196.
[9] Ausanka-Cures, R. "Methods for access control: advances and limitations."*Harvey Mudd College* 301 (2001).

