

BIOMETRIC ELECTRONIC WALLET FOR DIGITAL CURRENCY

Suhas M S¹, Abhilash C B², Vikas K C³, Amit Pareek⁴

¹Dept. of Computer Science and Engineering, JSS Academy of Technical Education, Bangalore, India

²Assistant Professor, Computer Science and Engineering Dept., JSS Academy of Technical Education, Bangalore

³Dept. of Computer Science and Engineering, JSS Academy of Technical Education, Bangalore, India

⁴Dept. of Computer Science and Engineering, JSS Academy of Technical Education, Bangalore, India

Abstract

Bitcoin is an experimental, decentralized digital currency that enables instant payments to anyone, anywhere in the world. Bitcoin uses peer-to-peer technology to operate with no central authority. Wallet is the place where Bitcoins are stored. Once Bitcoin wallet is installed on a computer or mobile phone, it will generate initial Bitcoin address and then the user can create an address for each transaction. Wallet first generates a private key and next it will convert private key to Bitcoin address. Wallet keeps track of private keys, usually by storing them in an encrypted wallet file, either on hard drive, on a server on the Internet, or elsewhere. If the private key to an address is lost (for example, in a hard drive crash, fire or other natural disaster), any associated Bitcoins are effectively lost forever. Existing solutions in the market use simple encryption mechanism to access sensitive information in electronic wallets. There is a need for a high security wallet that uses advanced encryption (AES) and unique access control (like finger print or facial recognition technology) as there is a high chance of security breach where hackers can steal sensitive information or electronic cash. The overall objective of this paper is to propose a biometric electronic wallet to store and transfer digital currencies with high security/encryption using Biometric External USB Sensor. All client side data in mobile Bitcoin wallets will be stored in a file in mobile device only. So, anybody can access that file. Therefore, the main objective is to provide a security for client side file by using Biometric External USB Sensor and AES algorithm.

Keywords: Bitcoins, Bitcoin wallet, Biometric External USB Sensor.

1. INTRODUCTION

Digital currency is nothing but a virtual currency and it generally called as Bitcoins where a Bitcoin is decentralized peer to peer payment network that is powered by its users with no central authority or Middleman. Bitcoin network is controlled by all Bitcoin users around the world. In user perspective, Bitcoin is nothing more than a mobile application or computer program that provides a personal Bitcoin wallet and allows a user to send and receive Bitcoins with them.

Bitcoins are stored in Bitcoin wallet. It can be application on mobile or software on computer. A "digital wallet" refers to a mobile application for smartphones that allows users to replace their wallet with smartphone technology, by giving options like making payments and performing monetary transactions solely using the technology inside of the device, as well as providing valid forms of identification. Perhaps a better way to define a wallet is something "that stores the digital credentials for client's bitcoin holdings" and allows him to access and spend them. Bitcoin uses public-key cryptography, in which two cryptographic keys, one public and one private, are generated. The public key can be thought of as an account number and the private key, ownership credentials. In contrast, wallet is a collection of these keys.

If client wants to receive bitcoins, he needs to have a Bitcoin address. In order to generate an address, client's wallet first generates a private key. Next, wallet converts that private key to a Bitcoin address using a well-known function. If anyone knows private key, they could easily convert it to a Bitcoin address, too. Moreover, Wallet keeps track of private keys, usually by storing them in an encrypted wallet file, either on hard drive, on a server on the Internet, or elsewhere. If the private key to an address is lost (for example, in a hard drive crash, fire or other natural disaster), any associated Bitcoins are effectively lost forever.

All users related secret information will be stored in a file called **Wallet.dat**. But this file will be stored in a mobile itself and it is less secured. So our main objective is to propose a Biometric electronic wallet for digital currency. Our Biometric Electronic Wallet for Digital Currency uses external Biometric USB sensor module to provide a high security wallet that uses advanced encryption and unique access control. In proposed system we will save that **Wallet.dat** file in an external USB sensor which is explained in detail in proposed systems. In this paper we discuss about an android application called Bitcoin wallet which already exists and we need to integrate external USB sensor module to that already existing application to make it more secured. We use AES256 standard algorithm to encrypt the contents of **Wallet.dat** file before

sending it to the external USB after each transaction and then we decrypt the contents of Wallet.dat file before moving the **Wallet.dat** file from external USB to the mobile device. We will discuss this procedure in detail in proposed system chapter. In following chapters we briefly go through basic concepts like Bitcoin Distribution, Bitcoin Transactions, Bitcoin Addresses, Authorizing Transactions, Existing Systems, Proposed Systems, Results and discussions and finally Conclusion.

2. RELATED WORK

There are different kinds of Bitcoin Clients [3]. Similarly there are different kinds of Bitcoin wallets exists [4]. But these wallets can be categorized into three main types which are explained below:

A **desktop wallet** is an application client installs on Windows, MacOS or Linux. Example : Electrum [4] and Multibit [4] client's private keys are stored locally, in a file somewhere on client's hard drive such as wallet.dat, and the security of his bitcoins is only as good as his ability to protect that file from data loss and theft.

There are also **web wallets** such as Coinbase [4] or Blockchain info's MyWallet [4] service. When using a web wallet, client's private keys are stored – usually encrypted – on the website's servers instead of his own hard drive. Web wallets allow you to use Bitcoin on any browser or mobile and often offer additional services.

A **mobile wallet** is an app you install on a smartphone or tablet. One notable exception is Bitcoin Wallet for Android, which stores private keys directly on client's mobile device. BitCoin Wallet [4] Android application is another example.

We will understand the major disadvantage when we go through a practical example of Bitcoin transaction [6]. Now a days Certified Coins [9] are used to provide more security.

Existing solutions in the market use simple encryption mechanism to access sensitive information in electronic wallets. In most cases simple "Login ID and password" is used and in some cases 2-factor authentication is used. Some wallets encrypt information whereas most wallets don't encrypt sensitive data [1]. Due to this there is a high chance of security breach where hackers can steal sensitive information or electronic cash. But in all the above applications and software there is less security for client side file wallet.dat which contains all the secret information about the client like private keys. So if a hacker gets the access to this file then he can get all the Bitcoins collected by the client [3]. So our main objective is to make a Bitcoin better currency [7] by providing high level of security.

There is a need for a high security wallet that uses advanced encryption (AES) [2] and unique access control (like finger print or facial recognition technology) as there is a high chance of security breach where hackers can steal sensitive information or electronic cash. Where Advanced Encryption Standard (AES) [2] is one of the most frequently used and most secure encryption algorithms available today. AES remains the preferred encryption standard for governments, banks and high security systems around the world. Our solution which includes a hardware sensor and an app will full this specific need in the market.

3. BITCOIN DISTRIBUTION

There is no central server to keep track of everyone's bitcoins. But that doesn't mean there are no servers keeping track of bitcoins. But in fact, there are, in fact, hundreds of thousands of servers keeping track of bitcoins. Each server in the Bitcoin network is called a **node**.

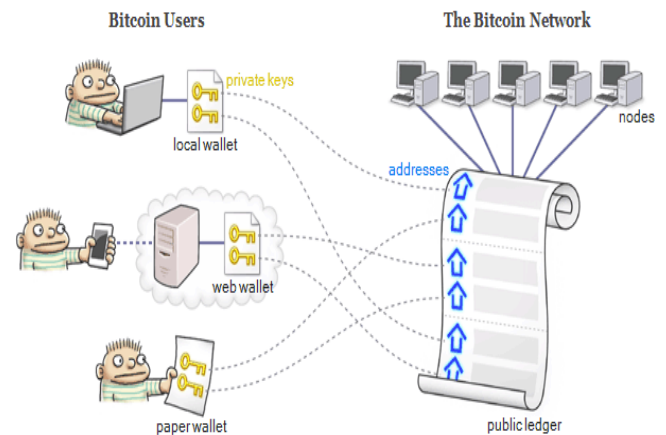



Fig -1: Bitcoin distribution

A Bitcoin node is an electronic bookkeeper, and anybody in the world can set up and run a node. Each node has a complete copy of the **public ledger** – that is a record of every Bitcoin transaction that ever happened, in history, all the way back to the very beginning of Bitcoin [8].

5JrFqG1rMLy6SkoWcZktr3HGqTSaXj63VAvQJNrrj78Yhb1FtB 

To actually use bitcoins, client needs some kind of device which functions as a **wallet**. Bitcoins are stored in Bitcoin wallet. It can be an application running on computer, a mobile app, a service offered by a website or something else entirely. Perhaps a better way to define a wallet is something "that stores the digital credentials for client's bitcoin holdings" and allows him to access and spend them. Wallet can add a transaction to the public ledger by informing a single node on the Bitcoin network. That node will relay the transaction to other nodes, which will relay it to others, and so on – similar to

the way BitTorrent works. It only takes about 7 seconds for a transaction to propagate across the entire Bitcoin network [8].

4. BITCOIN TRANSACTION

By now, it should be apparent that when client “send” bitcoins to another person, he aren’t really sending anything directly to that person. Instead, his wallet reassigns those bitcoins, from one owner to another, by adding a transaction to the public ledger.

We define an electronic coin [1] as a chain of digital signatures. Each owner transfers the coin to the next by digitally signing a hash of the previous transaction and the public key of the next owner and adding these to the end of the coin. A payee can verify the signatures to verify the chain of ownership.

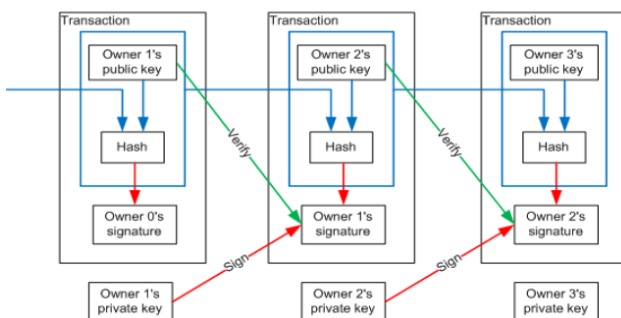


Fig -2: Bitcoin transactions

The problem of course is the payee can't verify that one of the owners did not double-spend the coin. A common solution is to introduce a trusted central authority, or mint, that checks every transaction for double spending. After each transaction, the coin must be returned to the mint to issue a new coin, and only coins issued directly from the mint are trusted not to be double-spent. The problem with this solution is that the fate of the entire money system depends on the company running the mint, with every transaction having to go through them, just like a bank. We need a way for the payee to know that the previous owners did not sign any earlier transactions. For our purposes, the earliest transaction is the one that counts, so we don't care about later attempts to double-spend. The only way to confirm the absence of a transaction is to be aware of all transactions. In the mint based model, the mint was aware of all transactions and decided which arrived first [1].

To accomplish this without a trusted party, transactions must be publicly announced, and we need a system for participants to agree on a single history of the order in which they were received. The payee needs proof that at the time of each transaction, the majority of nodes agreed it was the first received. For this purpose we use **Public ledger**- that is a

record of every Bitcoin transaction that ever happened, in history, all the way back to the very beginning of Bitcoin. We can also have Bitcoin transaction network [10] which shows how transactions are done in Bitcoin network.

5. BITCOIN ADDRESSES

If client wants to receive bitcoins, client needs to have a Bitcoin address. Client's wallet can generate addresses for client and these are stored in **Wallet.dat** file.

In order to generate an address, client's wallet first generates a **private key** [3]. A private key is nothing but a large random number roughly between 1 and 2^{256} . To make such numbers shorter to write, it's customary to encode them as sequence of numbers and letters as shown below.

Next, client's wallet converts that private key to a Bitcoin address using a well-known function. This function is very straightforward for a computer to perform. If anyone knows client's private key, they could easily convert it to a Bitcoin address, too. In fact, many Bitcoin wallets have a feature allowing client to import private keys [3].



On the other hand, it's extremely difficult to go the other way. If someone knows only client's Bitcoin address, it's virtually impossible to figure out what the private key was.



It's perfectly safe to give client's Bitcoin addresses to other people, but extremely important to keep his private keys secret. Most of the time, as a Bitcoin user, client will never even see his own private keys. Typically, his wallet keeps track of his private keys for him, usually by storing them in an encrypted wallet file, either on his hard drive, on a server on the Internet, or elsewhere.

6. AUTHORIZING TRANSACTIONS

That brings us to why it's important to keep client's private keys secret: his private keys give him the ability to spend the bitcoins he have received. To see how, take a closer look at the second transaction in the above listing, (*b6f4ec453a021ac561...*). This transaction spend the Bitcoins from previous output (*e14768c1d648b98a52...:0*). When we examine that previous output, we see that those Bitcoins were previously sent to the address (*1NqUajrFeStshjad1bhrEFFzWSQw6JHbqv*). It stands to reason that this transaction should be authorized by whoever generated that

address in the first place. That’s where the **digital signature** [5] comes in. In Bitcoin transaction we use Elliptic curve Digital Signature [5]. In Bitcoin, a valid digital signature serves as proof that the transaction was authorized by the address’s owner. Here’s what makes it safe: Just as a private key was required to generate that address, the same private key is required, once again, to generate a valid digital signature.

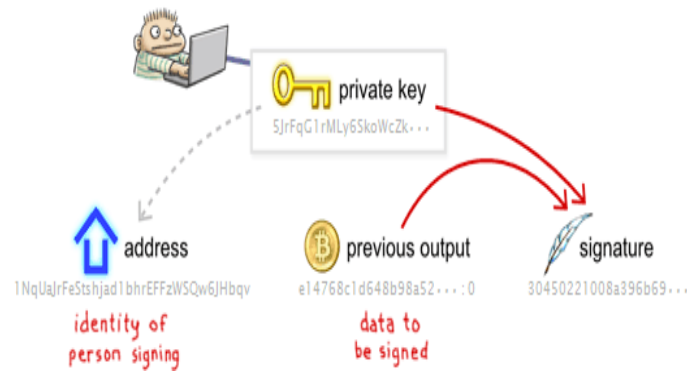


Fig -3: Digital signatures for transactions

A digital signature is only valid if a specific equation is satisfied by the address, the previous output and the signature. As client had expected, every time a Bitcoin node receives a new transaction, it checks to make sure each digital signature is valid. The node has no idea which private key was used to generate each signature, but that’s OK, because it doesn’t need to know. It only needs to verify that the equation is satisfied.



Fig -4: Authorizing Transactions

The concept of digital signatures is based on old idea known as public-key cryptography. Bitcoin is not the first digital currency to secure transactions using such cryptography, but it is the first to do so without relying on a single, centralized server. That’s the breakthrough at the heart of Bitcoin [5].

In further chapters we will discuss about the Existing system and proposed systems where there is an android application called Bitcoin wallet and we need to integrate external USB sensor module to that application in order to protect client’s side file called **Wallet.dat**.

7. EXISTING SYSTEM

The original Bitcoin client wallet file is named wallet.dat and contains:

- key pairs for each of your addresses
- transactions done from/to your addresses
- user preferences
- default key
- reserve keys
- accounts
- a version number
- Key pool

Wallet keeps track of private keys, usually by storing them in an encrypted wallet file, either on hard drive, on a server on the Internet, or elsewhere. If the private key to an address is lost (for example, in a hard drive crash, fire or other natural disaster), any associated Bitcoins are effectively lost forever.

Existing solutions in the market use simple encryption mechanism to access sensitive information in electronic wallets. In most cases simple “Login ID and password” is used and in some cases 2-factor authentication is used. Some wallets encrypt information whereas most wallets don’t encrypt sensitive data. Due to this there is a high chance of security breach where hackers can steal sensitive information or electronic cash.

There is a need for a high security wallet that uses advanced encryption (AES) and unique access control (like finger print or facial recognition technology) as there is a high chance of security breach where hackers can steal sensitive information or electronic cash. Where Advanced Encryption Standard (AES) [2] is one of the most frequently used and most secure encryption algorithms available today. AES remains the preferred encryption standard for governments, banks and high security systems around the world. Our solution which includes a hardware sensor and an app will full this specific need in the market.

8. PROPOSED SYSTEM

Our Biometric Electronic Wallet for Digital Currency uses external Biometric USB sensor and Bar code/QR code module to provide a high security wallet that uses advanced encryption and unique access control.

Here we propose a system for android phones which can be further enhanced for other systems. The main objective of the propose system is to protect the **Wallet.dat** file. An android application called Bitcoin wallet already exists and we need to integrate USB module to this application.

Bitcoin wallet application has following features:

- No registration, web service or cloud needed! This wallet is de-centralized and peer to peer.

- Display of Bitcoin amount in BTC, mBTC and μ BTC.
- Conversion to and from national currencies.
- Sending and receiving of Bitcoin via NFC, QR-codes or Bitcoin URLs.
- Address book for regularly used Bitcoin addresses.
- When you're offline, you can still pay via Bluetooth.
- System notification for received coins.
- App widget for Bitcoin balance

In a proposed system we integrate external USB memory stick with fingerprint scanner. This prototype Biometric sensor has already been developed and it is available in market.

8.1 Steps to be followed after each Transaction

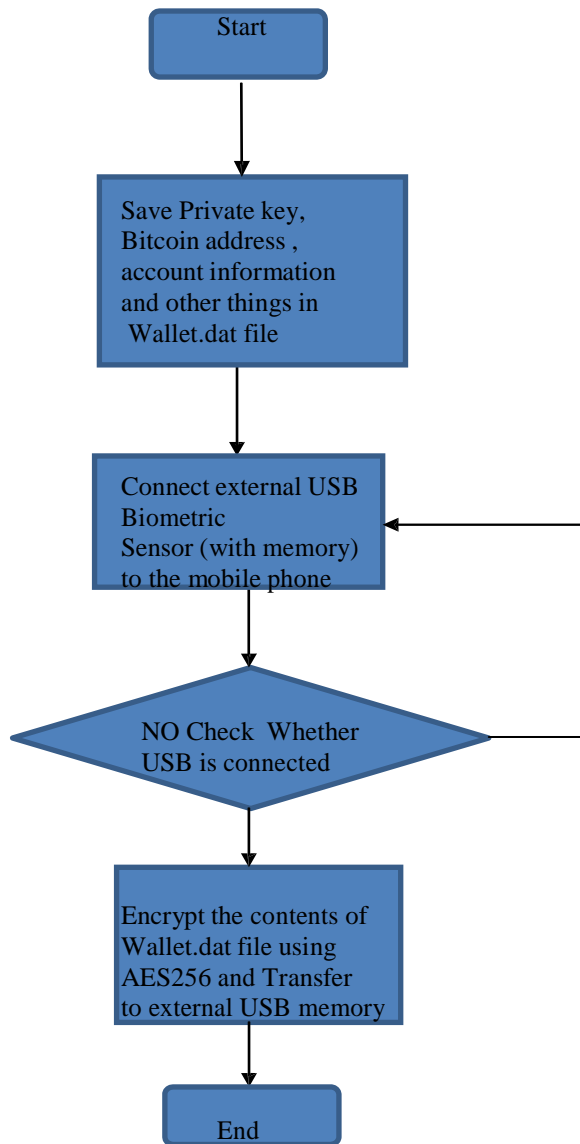


Fig -5: Flow chart of operations to be done after each transaction

Flow chart is explained as below:

- Generate private key for particular transaction and save it in **Wallet.dat** file.
- Convert the private key into Bitcoin address of the user.
- **Wallet.dat** file now has all secret information about the user and his transactions.
- Now use an external USB memory stick along with fingerprint scanner.
- Encrypt the contents of **Wallet.dat** file using AES256 algorithm and then transfer it to the external USB memory.

Now, **Wallet.dat** file is saved in an external USB memory stick rather than on Mobile device. So, after each transaction follow the above steps.

8.2 Steps to be followed before each transaction

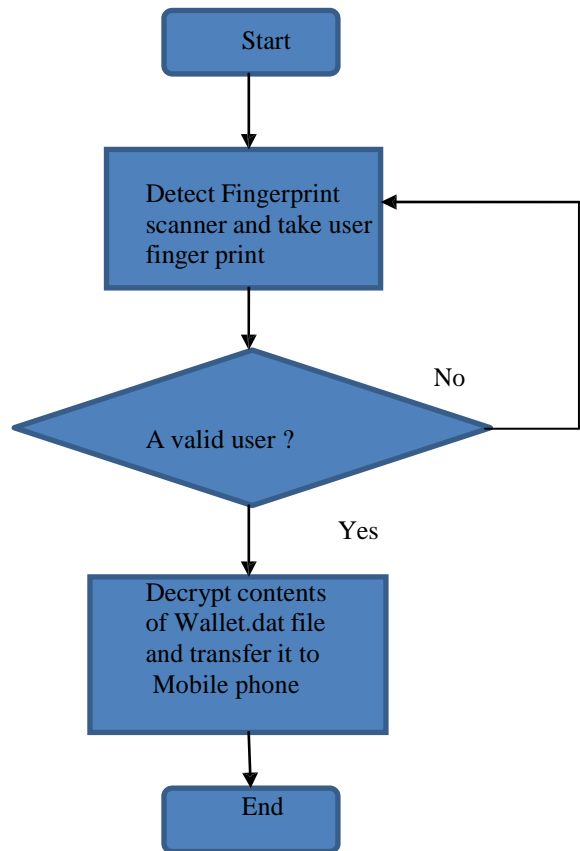


Fig -6: Flow chart of operations to be done before each transaction

Flow chart is explained as below:

- Using fingerprint-scanner get the access to the external memory stick
- Decrypt the contents of **Wallet.dat** present in external memory stick.

- Transfer that decrypted file to Mobile phone.
- Do transactions with the private keys present in Wallet.dat file.

So each time before the transaction these steps are followed.

9. RESULTS AND DISCUSSION

We have to integrate the External USB sensor module to the application which currently exists in android market called Bitcoin wallet. Here we will explain few screenshots of already existing Bitcoin wallet android application. To this application we need to integrate external USB sensor module. The following screenshots shows the different functionalities that can be performed using this application:

The first screenshot shows the home screen of the app where a user’s Bitcoin address and its corresponding QR code are present. There are also options like “Send Bitcoins”, “Peer monitor” etc., which have their own functionalities.

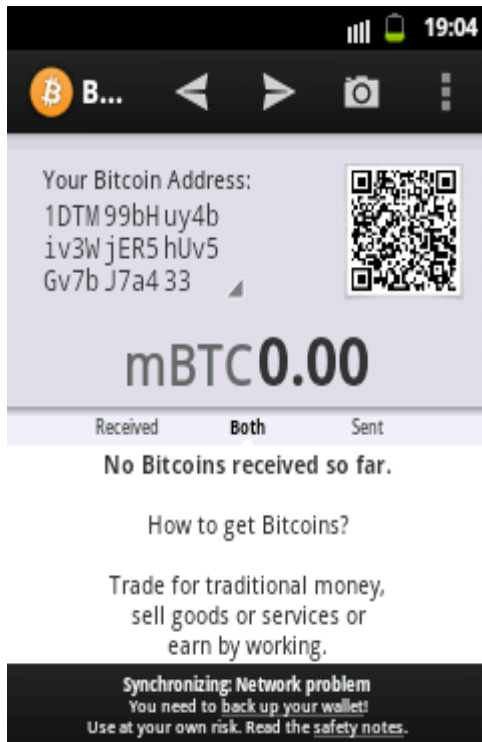


Fig -7: Home screen

In above screen shot as we have not yet got the Bitcoins it is showing balance as mBTC 0.00.

Next screenshot shows address book of the user where he has different Bitcoin addresses (alphanumeric characters of length 27 to 34). Here user has saved many addresses to which he has performed different transactions.

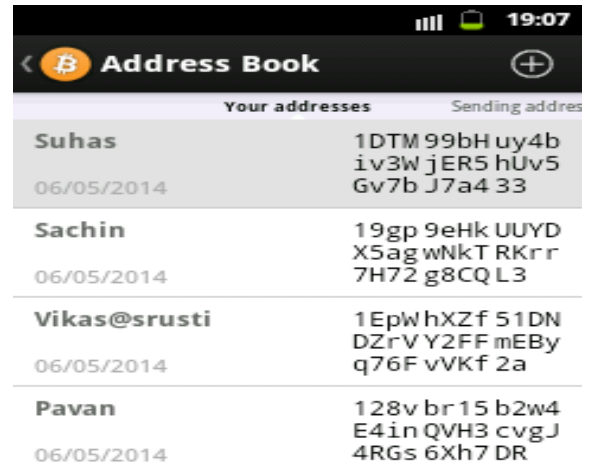


Fig -8: Address Book of User

Next two screenshots shows how to request and how to send Bitcoins to other users. While sending Bitcoins client needs to know the address of other user to whom client wishes to send. Then client has to enter the amount that he wishes to send. Here client should have enough balance to send Bitcoins. After client sends Bitcoin that transaction will be updated in public ledger and also in address book of the client.

While requesting for Bitcoin client has to enter amount. He can request for Bitcoins from any of the users addresses client having in his address book. Other users will get our address or QR code along with the request. Then they can send coins to the mentioned address.

If client does not have enough Bitcoins then in Send Bitcoin screen it will pop up a message saying that not enough balance. Clients can send Bitcoin address or QR code. When they send QR codes other users should scan that QR code to get the actual Bitcoin address.

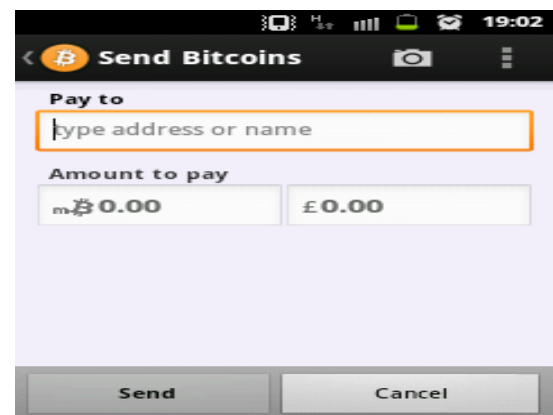


Fig -9: Sending Bitcoins

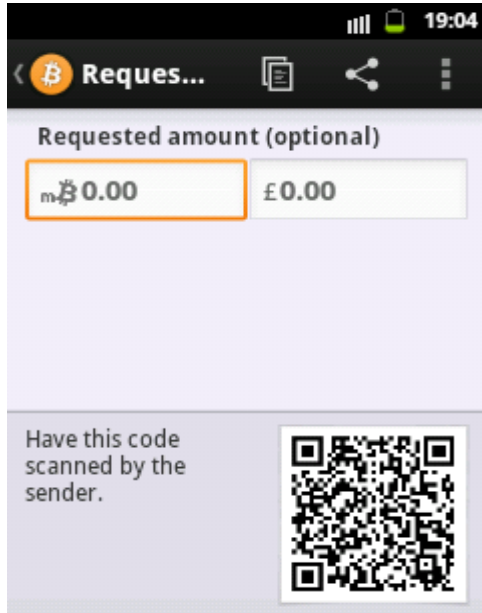


Fig -10: Requesting Bitcoins

All the above screen shots are from already developed application called Bitcoin Wallet present in android market. For that application we have to integrate USB module and hardware required for that is as shown in below image. (This USB sensor can be used only with android devices which supports USB host mode that is devices having API-12 and above or android version 3.1 and above)



Fig -11: Biometric External USB Sensor (Memory)

The above external USB sensor is a device which has in built fingerprint scanner and also memory. The **Wallet.dat** file in mobile should be transferred to this device and should be saved in an encrypted form after each transaction and same **Wallet.dat** file should be decrypted and should be transferred back to mobile before doing next transactions. This device has in built memory and fingerprint scanner to access that memory. For encryption and decryption its better to use AES256 standard algorithm

10. CONCLUSIONS

In this paper we have discussed basics of Bitcoin, Bitcoin addresses and Bitcoin transaction. Here we have also discussed the currently available Bitcoin wallets and their disadvantages. Then in proposed systems we have used an external USB sensor which has in built fingerprint scanner. We also discussed different function of Bitcoin wallet application which is present in android market. Here we have not discussed about USB module which is required to connect the external USB sensor to the mobile phone. But it has been already coded and it will be integrated to the already existing application. (This USB sensor can be used only with android devices which supports USB host mode that is devices having API-12 and above or android version 3.1 and above) Here we have explained about protecting client's side file **Wallet.dat** (which has secret information about client) of a particular mobile wallet called Bitcoin wallet. This same method can be enhanced and used to other type of wallets also like web wallets and desktop wallets.

ACKNOWLEDGEMENTS

We are thankful to Mr. Raj Shetgar for providing us all the required documents and helping us to prepare this paper. We are also thankful to Mr. Manjit (Developer of external USB module) for providing us information's regarding External USB Sensor module.

REFERENCES

- [1]. Satoshi Nakamoto - "Bitcoin: A Peer-to-Peer Electronic Cash System" – technical report 2008.
- [2]. William Stallings -"Advanced Encryption standard" -5th chapter in Cryptography and network Security 3rd edition book.
- [3]. Rostislav Skudnov -"Bitcoin Clients" Bachelor's Thesis Turku University of Applied Science
- [4]. Azzief Khaliq -"10 Best Bitcoin Wallets For Secure Bitcoin Storage" [Online] <http://www.hongkiat.com/blog/bitcoin-wallets/>
- [5]. Oleg Andreev -"Blind signatures for Bitcoin transactions" - Second draft February 22, 2014
- [6]. Tobias Bamert, Christian Decker, Lennart Elsen, Roger Wattenhofery, Samuel Welten -" Have a Snack, Pay with Bitcoins", 13-th IEEE International Conference on Peer-to-Peer Computing.
- [7]. S. Barber, X. Boyen, E. Shi, and E. Uzun, "Bitter to better– how to make bitcoin a better currency," in Financial Cryptography 2012, vol. 7397 of LNCS, 2012, pp. 399–414.
- [8]. D. Ron and A. Shamir, "Quantitative Analysis of the Full Bitcoin Transaction Graph," Cryptology ePrint Archive, Report 2012/584, 2012
- [9]. Giuseppe Ateniese, Antonio Faonio, Bernardo Magri, and Breno de Medeiros, "Certified Bitcoins"- Research paper, Sapienza - University of Rome, Italy; Johns Hopkins University, USA.

[10]. F. Reid and M. Harrigan, "An analysis of anonymity in the Bitcoin system," in Privacy, security, risk and trust (PASSAT), 2011 IEEE Third Internatiojn Conference on Social Computing (SOCIALCOM). IEEE, 2011, pp. 1318–1326.

BIOGRAPHIES



SUHAS M S, Student of JSS Academy of Technical Education, Bangalore, Karnataka, India. Currently pursuing 8th semester in Computer Science and Engineering



ABHILASH C B, Assistant Professor in Department of CSE, JSS Academy of Technical Education, Bangalore, Karnataka, India.



VIKAS K C, Student of JSS Academy of Technical Education, Bangalore, Karnataka, India. Currently pursuing 8th semester in Computer Science and Engineering



AMIT PAREEK, Student of JSS Academy of Technical Education, Bangalore, Karnataka, India. Currently pursuing 8th semester in Computer Science and Engineering