A3. Architectural Enhancement

CISC 322/326

Assignment 3 Report

Tuesday, April 11, 2023

Group 6: ArcProject
Chuyang Li (19cl98@queensu.ca)
Qintao Zhang (18qz28@queensu.ca)
Anthony Zhou (18zz172@queensu.ca)
Chang Xu (18cx19@queensu.ca)
Wenchu Xiao (19wx25@queensu.ca)
Xiaoran Zhang(19xz64@queensu.ca)

Table of Contents

1. Abstract

After an in-depth examination and analysis of the specific architecture of Bitcoin Core, we have summarized the 10 core components and their respective dependencies. While the current architecture is robust, there is still potential to introduce new features to enhance its overall capabilities. The focus of this report is on the addition of a new feature to the wallet component and the necessary modifications to other components for a successful implementation.

2. Introduction and Overview

The booming world of digital currencies and their increasing global adoption has highlighted the importance of solid privacy and security features in cryptocurrency wallets. Consequently, the Bitcoin Core wallet needs to keep evolving and enhancing its offerings to stay relevant in this ever-changing landscape. In this report, we present a meticulously researched proposal to boost the Bitcoin Core wallet by introducing a "Privacy Mode" feature, specifically tailored to strengthen user privacy and security during Bitcoin transactions. We'll explore the ins and outs of this proposed functionality, including an in-depth look at the core technologies, such as Coin Control, Automatic Coin Mixing, and Stealth Addresses, which collectively form a complete privacy solution. Moreover, we'll assess the implications of incorporating Privacy Mode into the existing Bitcoin Core architecture and lay out the necessary changes for smooth and successful integration. This comprehensive analysis paves the way for informed discussions, further research, and possible implementation, ultimately striving to create a safer, more secure, and private digital currency experience for all Bitcoin users. By tackling the mounting concerns about privacy in the cryptocurrency realm, our proposal aims to reinforce user trust in the Bitcoin network and encourage its ongoing growth and adoption in the global financial system.

3. Overview of the New Feature

The Privacy Mode feature offers a way to obscure transaction details and provides additional privacy-centric functionality by enabling features such as Coin Control, Automatic Coin Mixing, and Stealth Addresses. Much like privacy-focused tools like Tor, Privacy Mode allows users to better manage their transaction privacy and minimize the chances of third-party tracking.

Coin Control gives users the power to selectively use specific unspent transaction outputs (UTXOs) in their transactions, making it harder for trackers to spot transaction patterns. Automatic Coin Mixing presents an optional feature that blends users' coins with others', further complicating transaction history analysis.

Stealth Addresses generate single-use addresses for each transaction, ensuring a user's primary Bitcoin address remains unlinked to their transaction history.

Capitalizing on the growing adoption of Bitcoin and the increasing number of Bitcoin Core wallet users, the Privacy Mode feature can help create a safer and more private digital currency experience. As more users adopt Privacy Mode, the overall privacy of the Bitcoin ecosystem will improve, making it increasingly difficult for third parties to track and analyze transaction details.

In case of potential risks or vulnerabilities, such as security, maintainability, or performance issues, Privacy Mode will be designed to alert users, allowing them to make informed decisions about their privacy preferences. This cross-communication mechanism will empower the Bitcoin Core wallet to proactively address privacy concerns and adapt to evolving user needs and expectations.

By incorporating the Privacy Mode feature into the Bitcoin Core wallet, our goal is to offer users a convenient, effective, and robust way to enhance their privacy and security in the digital currency world.

3.1 Approach 1

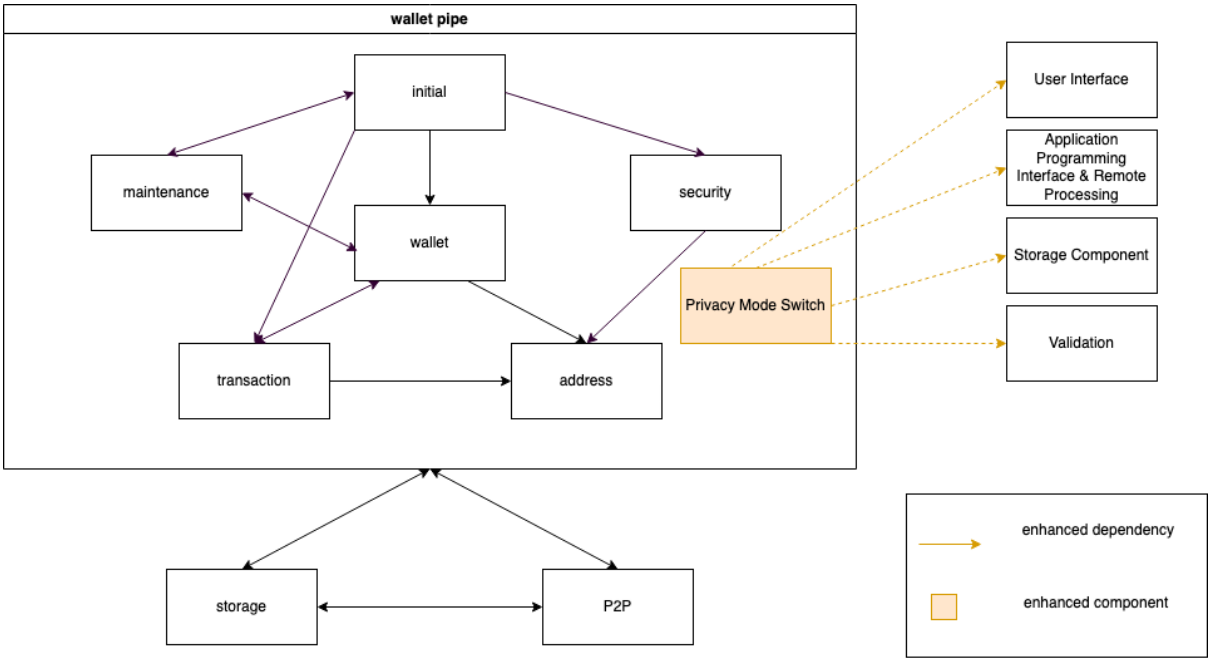

Figure 1. Approach 1 Architecture diagram

This implementation is to add a new function within the wallet subsystem without changing the original dependency. This new function called ‘Privacy Mode Switch’ which can enable privacy mode in order to hide the user’s IP address is intended to improve user privacy and security.

3.2 Approach 2

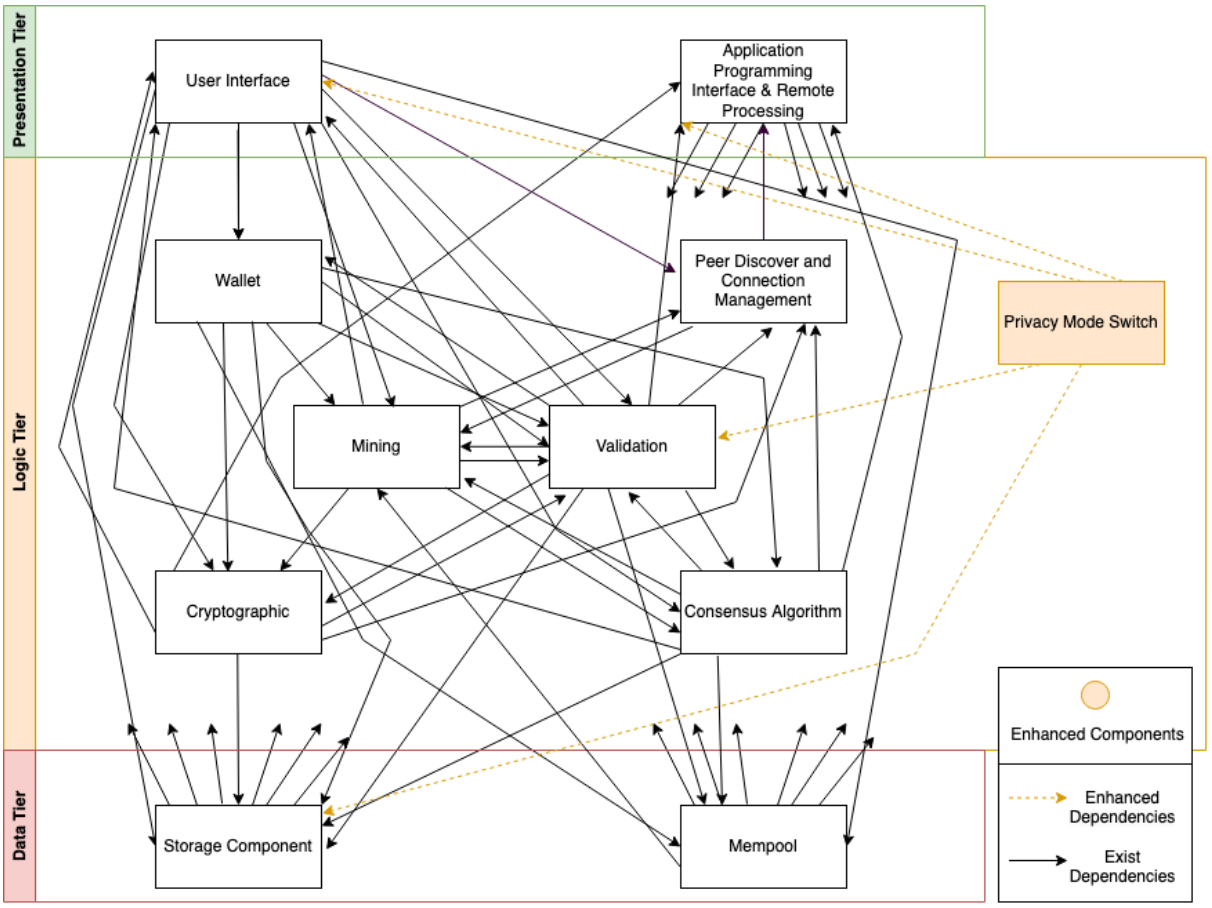

Figure 2. Approach 2 Architecture Diagram

4. SAAM Analysis

Stakeholder	Non-Functional Requirements (NFRs)
User	**Security**: Advanced encryption technology is used to enhance security and safeguard user data and transaction information from illegal access. **Usability**: A user-friendly experience is ensured using a new intuitive interface that makes privacy-enhancing features available to all users. **Privacy**: Using incognito addresses, coin blending and other privacy-enhancing technologies, the Privacy Mode feature helps users maintain their anonymity and prevent third parties from tracking their transaction history or personal information.
Developer	**Maintainability**: Follow modular design principles that allow developers to easily update and modify the software without causing disruption or interference with existing functionality. **Scalability**: The design should be able to efficiently handle the increased workload and ensure that the performance of the Bitcoin wallet remains stable and reliable. **Interoperability**: Compatible with other software and systems for seamless communication and interaction, providing a cohesive ecosystem.

Investor	**Stability**: Improve stability, attract more users, reduce potential vulnerabilities, and ensure that the software remains reliable and well-maintained. **Performance**: Provide a fast and responsive user experience and reduce delays or bottlenecks that may affect the value of an investor's holdings. **Transparency**: Provide investors with enough transparency to monitor bitcoin transactions and holdings to be able to make informed investment decisions.

Table 1. NFR Evaluation

To find a better implementation, we generate 5 major NFRs to compare the advantage and disadvantages of the two approaches (Table 2.).

NFRs	Approach 1	Approach 2
Modularity	Maintains the separate structure of the wallet component, but it can make the component more complex and harder to maintain.	Promote modularity by isolating privacy model functionality in a separate subsystem, making it easier to update and maintain without affecting other components.
Integration	Tighter integration with wallet components for a more seamless user experience.	Additional effort may be required to ensure smooth communication and interaction between the new subsystem and existing components.
Flexibility	May limit the flexibility of the privacy model functionality due to the close connection to the designs and constraints of the wallet component	Standalone subsystems provide greater flexibility, allowing for the addition of new privacy-enhancing features or integration with other privacy-focused services.
Performance	May result in increased resource usage and may affect wallet performance.	Minimize the impact on wallet component performance, resulting in better resource allocation and management.
Security	New vulnerabilities or complexities may be introduced that may affect the security of the wallet.	Helps isolate any potential security risks or vulnerabilities and mitigate the impact on other components within the Bitcoin Core architecture.

Table 2. Comparison of Two Approach

After carefully evaluating the non-functional requirements (NFRs) and comparing the two approaches, our team has decided to adopt Approach 2 for implementing the privacy mode in our Bitcoin wallet. This decision is based on several crucial factors, such as modularity, integration, flexibility, performance, and security. Approach 2 encourages modularity by separating the privacy mode

functionality into its own subsystem, which makes it easier to update and maintain without interfering with other components. Although this approach might require extra effort for integration, it offers more flexibility, allowing for the inclusion of new privacy-enhancing features or the integration with other privacy-centric services. In terms of performance, Approach 2 lessens the impact on the wallet component, leading to improved resource allocation and management. Additionally, it assists in isolating potential security risks or vulnerabilities, thereby reducing their impact on other components within the Bitcoin Core architecture. Taking these factors into account, we believe that Approach 2 is the optimal choice for enhancing the privacy features of our Bitcoin wallet.

5. The New Features Impact on Subsystems

5.1 Impact On High-Level Architecture

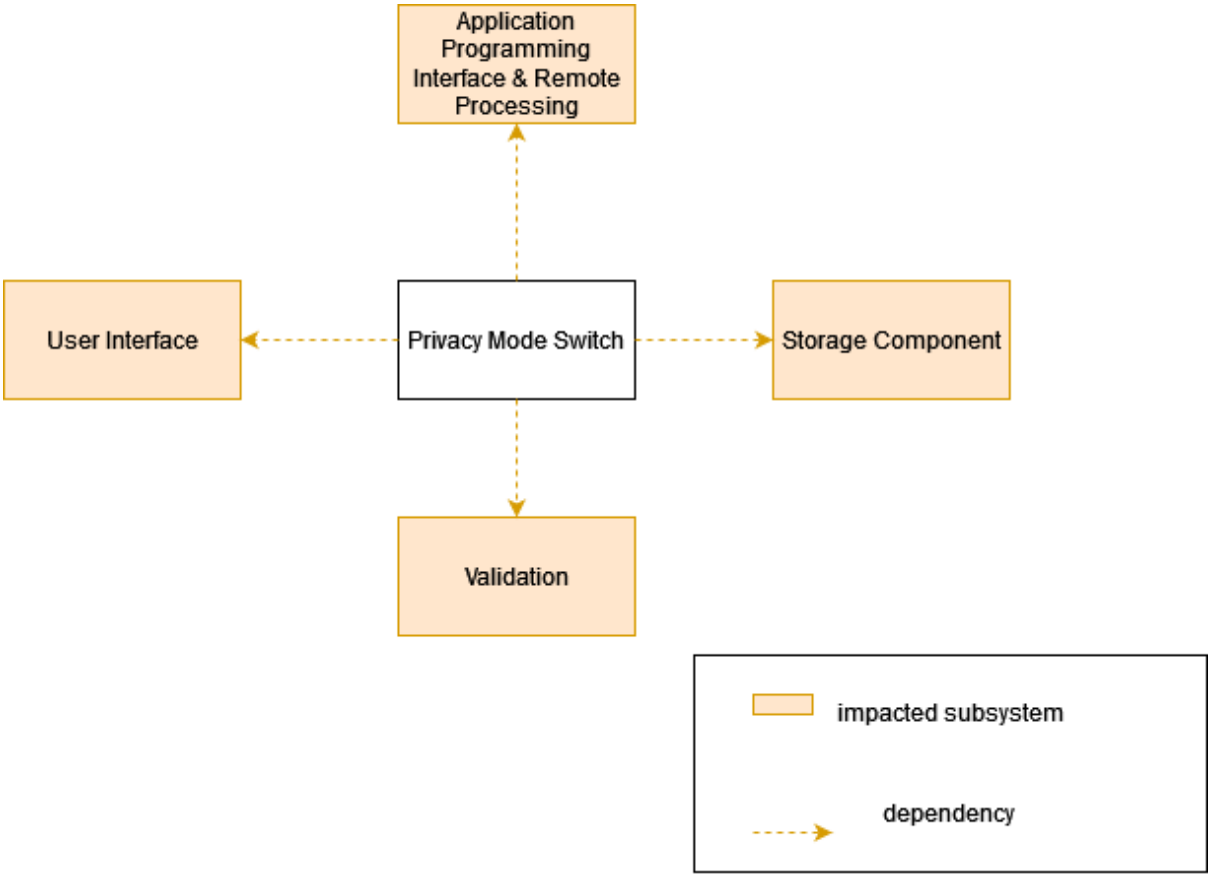

Figure 3. High-Level Architecture

5.2 Impact On Low-Level Architecture

5.2.1 User Interface

WalletTx(Struct):

Need to add a switch function that allows users to switch between privacy mode and normal mode.

WalletAddress(Struct):

Need to add an API to the new component that can retrieve encrypted IP addresses back and allow users to know the IP address is changed.

ipc:

Need to add a new function that sends control flows to the privacy switch subsystem in order to enable privacy mode.

5.2.2 API

In order to implement the new feature to the architecture, we need to make change to the source file:

src/wallet/wallet_rpc_server.cpp and src/wallet/wallet_rpc_server.h:

Add new RPC command related to the Privacy Mode.

src/rpc/server.cpp and src/rpc/server.h:

Needed change to accommodate and new privacy-related RPC calls or to modify existing ones.

src/rpc/protocol.cpp and src/rpc/protocol.h:

Include new privacy-related methods, parameters or error codes.

5.2.3 Storage Component

The storage subsystem is impacted by the data stored locally on the devices. If we add the privacy mode component, the files in the storage subsystem need to change, such as src/txdb.cpp and src/txdb.h. Both of them are responsible for the transaction database. The modifications might be needed to handle storing and retrieving privacy-related data.

5.2.4 Validation

The Validation component will also be affected by add the PMS component. Because PMS will make obtaining representative test data more difficult, potentially skewing performance measurements and necessitating additional testing and validation methods to assure compliance with privacy standards. Also, these files in the validation subsystem need to change: src/validation.cpp and src/validation.h. Since they are responsible transaction validation.

6. Sequence Diagrams

6.1 Transaction

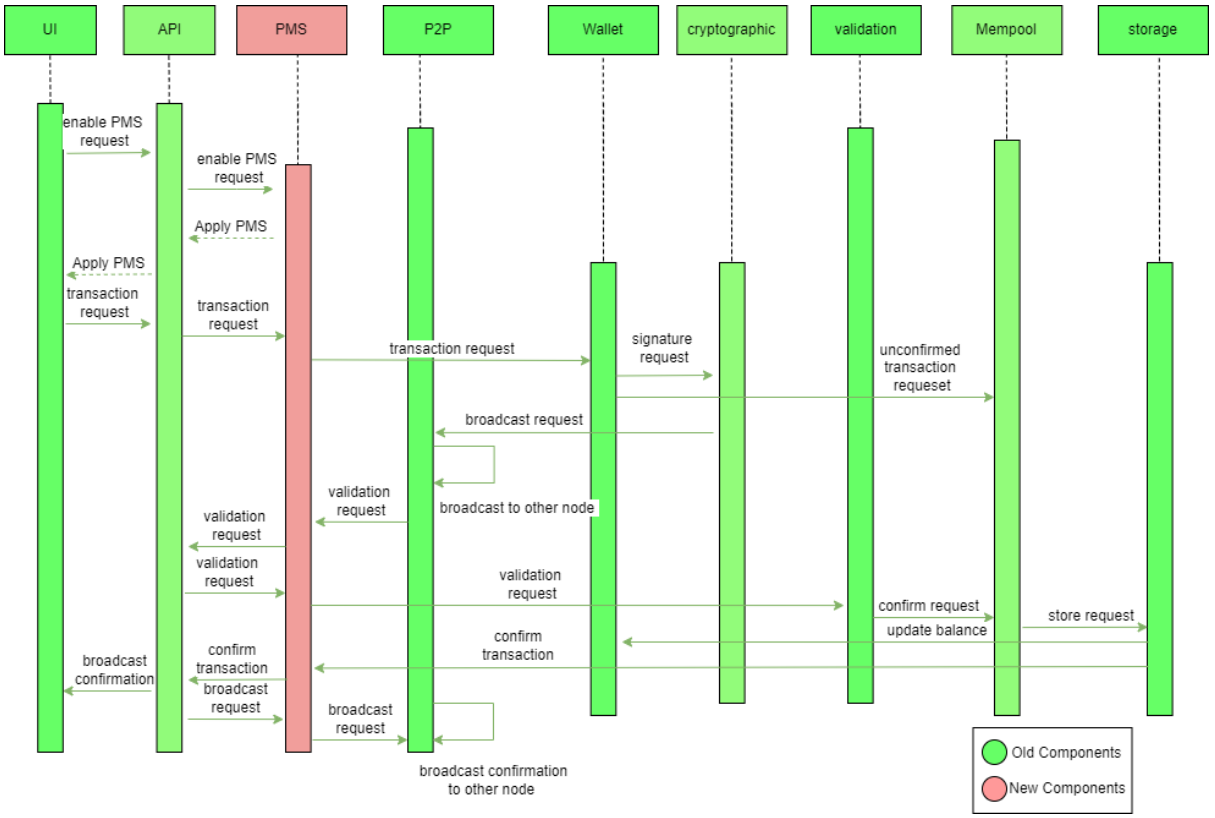

Figure 4. Transaction Sequence Diagram

The first use case shows a transaction scenario (Figure 4) in which the user first enables the PMS through the UI, then the user can send a transaction request to the API, which then forwards the request to the wallet component through PMS. At this point, the wallet component will do two things: first, it will make a signature request through the cryptographic component, and second, it will send the unverified transaction request to the mempool. After the signature request is done, the cryptographic component will send a broadcast request to the P2P network, where the P2P will broadcast the message to other nodes, and then send a validation request to the API through PMS, which will forward the request to the validation component, which will send the confirmation message to the mempool after a successful validation, so that the mempool will update the previous unconfirmed request. and save it to the storage component. After saving, the balance of the wallet will be updated and the completion message will be sent to the API through PMS, which will return the broadcast confirmation message to the UI for the user to check, and

send the broadcast request to the P2P network through PMS to broadcast it to other nodes. The above is the complete flow of this scenario.

6.2 Mining

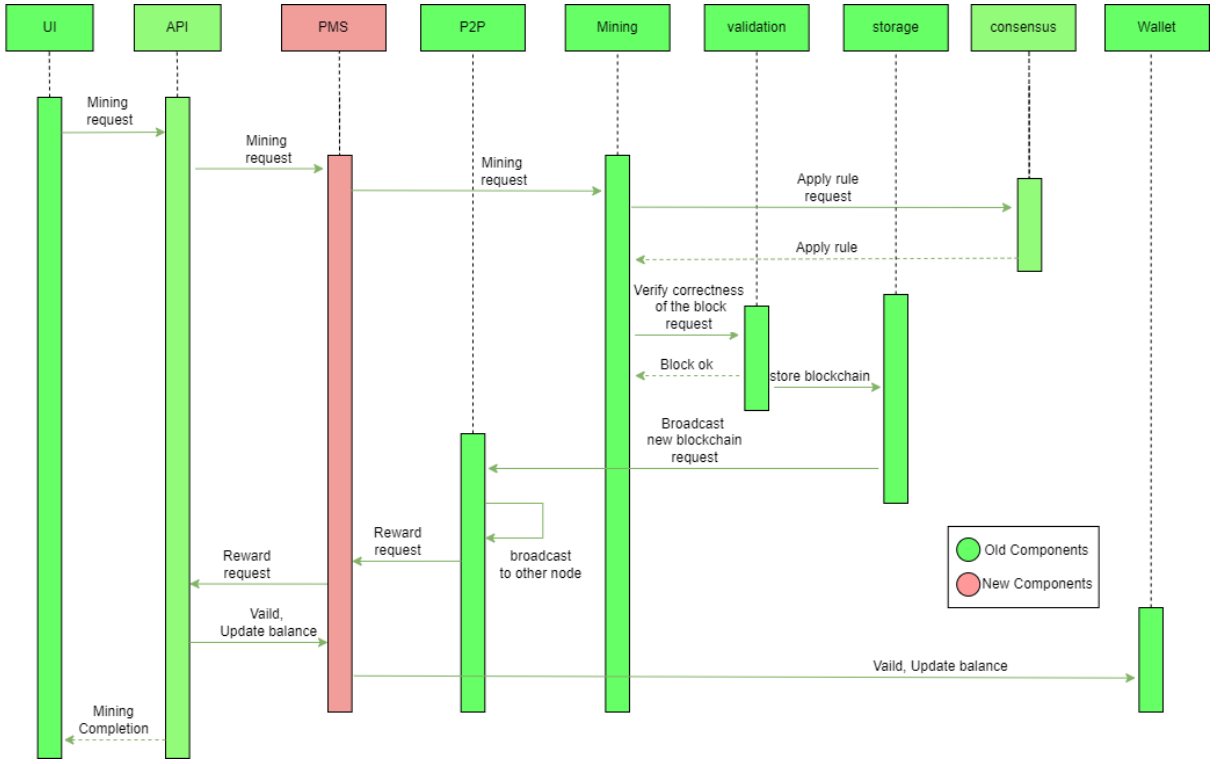

Figure 5. Mining Sequencce Diagram

The second use case presents the solo-mining scenario (Figure 5). The user can perform the mining operation through the UI. The PMS component is responsible for hiding the user IP in order to protect the security. First, API will connect to PMS in order to anonymize the IP address. Then the mining request is dispatched to Mining through PMS. The consensus includes the mining rules and some algorithms inside. When mining is running, Mining will request these regulations from Consensus, and load the terms. After Mining creates a new block, it will request Validation to verify the validity of transactions and blocks. For example, the electronic signature is correct, the transaction is correctly formatted and the block is correctly constructed and linked.

After successful verification, the new blockchain and other data are stored inside Storage and then notified to other users via the P2P network. At the same time, the P2P network will also request a reward from the API through PMS because a new block is created, then the API allows Wallet to update the balance through PMS.

7. Testing

Adding a privacy mode switch as a new subsystem to the Bitcoin Core architectural design would necessitate careful planning and execution. Unit testing, integration testing, and system testing would all be part of the process. To ensure the functionality and dependability of each component of the privacy mode subsystem, unit testing would be used. Testing the interaction and integration of the privacy mode subsystem with other already-existing subsystems and components of the Bitcoin Core software would be the main focus of integration testing. To find any potential problems or conflicts with other subsystems, system testing would entail thorough testing of the entire Bitcoin Core system, including the privacy mode switch, in a controlled environment. Security testing, in addition to functional testing, would be necessary to identify and address any potential vulnerabilities or risks associated with the privacy mode switch. Penetration testing, vulnerability assessments, and code audits would be performed to ensure that the privacy mode does not introduce any security flaws that attackers could exploit. The testing procedure should also take into account how the privacy mode affects the network's transparency, potential risks of criminal activity, and regulatory compliance. To determine the impact of the privacy mode on the overall integrity and transparency of the Bitcoin network, extensive testing of transaction flows, edge cases, and potential attack vectors may be required. The testing procedure should also consider how the privacy mode affects network transparency, potential criminal activity risks, and regulatory compliance. Extensive testing of transaction flows, edge cases, and potential attack vectors may be required to determine the impact of the privacy mode on the overall integrity and transparency of the Bitcoin network.

8. Potential Risk

If a privacy mode switch is added to Bitcoin Architecture, it may pose a number of risks to the network's overall integrity and transparency. One of Bitcoin's defining characteristics is its decentralized nature and transparency, which may be harmed by the addition of a privacy mode. A switch to privacy mode, for example, may make it difficult for regulators to track transactions and ensure compliance with AML and KYC regulations. Furthermore, it may serve as a cover for illegal activities such as money laundering, drug trafficking, and terrorist financing. A hard fork risk exists as well, as users may refuse to use the privacy mode, while others may switch to the new version. This could result in a network split, with two separate and incompatible versions of Bitcoin. Adding a privacy mode switch would also increase the Bitcoin protocol's complexity, potentially introducing new bugs or security vulnerabilities. Finally, the addition of a privacy mode switch may raise regulatory concerns, leading to additional restrictions on the use of cryptocurrencies.

9. Lesson Learned

Our team thoroughly enjoyed the process of developing a privacy mode feature for the Bitcoin Core wallet. We quickly grasped the importance of meticulously examining different implementation strategies and fostering open

dialogue throughout the entire process. Additionally, we realized that even minor design tweaks could lead to remarkable outcomes, which is why we invested time in thoroughly exploring all options and selecting the optimal privacy enhancement solution.

Another invaluable lesson we learned was the significance of encouraging transparent communication within our team. Embracing this approach not only facilitated the successful development of this feature, but it also set the stage for our team to collaboratively tackle future projects and enhancements. By adhering to these principles, we can persistently refine the Bitcoin Core wallet, address user needs, and stay at the cutting edge of cryptocurrency solutions. Our team is more dedicated than ever to ensuring our users experience top-tier privacy and security on our platform.

10. Conclusion

In conclusion, this report presents a comprehensive proposal for enhancing the Bitcoin Core wallet by introducing a new Privacy Mode feature designed to bolster user privacy and security. Through a thorough evaluation of two different approaches, we have determined that Approach 2, which implements Privacy Mode as a separate subsystem, best meets the non-functional requirements of modularity, integration, flexibility, performance, and security. The proposed Privacy Mode feature combines Coin Control, Automatic Coin Mixing, and Stealth Addresses to provide users with a convenient, effective, and robust means to enhance their privacy and security in the world of digital currencies. As the adoption of this feature increases, the overall privacy of the Bitcoin ecosystem will improve, making it more challenging for third parties to track and analyze transaction details, ultimately strengthening user trust in the Bitcoin network and promoting its continued growth and adoption in the global financial ecosystem. By addressing the growing concerns around privacy in the cryptocurrency space, this enhancement report sets the stage for informed discussions, further research, and potential implementation of the Privacy Mode feature, aiming to contribute to a safer, more secure, and private digital currency experience for all Bitcoin users.

11. Glossary

Bitcoin Core Wallet: The official Bitcoin wallet software that is a reference implementation of the Bitcoin protocol, allowing users to send, receive, and manage their Bitcoin transactions.

Privacy Mode: A proposed feature for the Bitcoin Core wallet designed to enhance user privacy by seamlessly integrating a combination of native privacy features and external privacy services within the wallet interface.

Coin control: A wallet feature that allows users to select specific unspent transaction outputs (UTXOs) for their transactions, providing better control over the privacy of their transactions.

Coin mixing: A privacy technique that mixes a user's coins with other users' coins to obfuscate transaction history, making it more difficult for third parties to track.

Stealth addresses: One-time use addresses are generated for each transaction to ensure that a user's main Bitcoin address is not directly linked to their transaction history.

SEI SAAM: Software Engineering Institute's Software Architecture Analysis Method, a technique for evaluating architectural alternatives based on their ability to meet stakeholder requirements.

UTXOs: Unspent transaction outputs, the indivisible units of Bitcoin that can be spent in a transaction.

Public key: A cryptographic key that is publicly shared and used to verify digital signatures, allowing others to confirm the authenticity of a transaction.

Private key: A secret cryptographic key that is kept private by its owner and used to sign digital transactions, proving ownership of the associated public key and the ability to spend the corresponding funds.

12. Reference

[1] Bitcoin. (n.d.). *Bitcoin/Bitcoin: Bitcoin Core Integration/Staging tree*. GitHub. Retrieved March 23, 2023, from https://github.com/bitcoin/bitcoin

[2] *P2P network*. Bitcoin. (n.d.). Retrieved March 23, 2023, from https://developer.bitcoin.org/devguide/p2p_network.html