sebis

TUM

Design and Evaluation of Secure Multi-Party Computation Approaches for Non-Custodial Crypto Wallets with a Focus on User Experience and Security

Lucas Kissling

03.06.2024, Master’s Thesis Final Presentation

Chair of Software Engineering for Business Information Systems (sebis)
Department of Computer Science
School of Computation, Information and Technology (CIT)
Technical University of Munich (TUM)
wwwmatthes.in.tum.de

Outline

1. Motivation and Introduction
2. Problem Statement
3. Research Questions & Results
4. Live Demo
5. Evaluation & Future Work

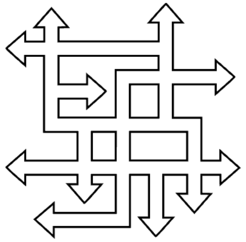
Motivation - Security and usability challenges of crypto asset self-custody

Digital assets such as cryptocurrencies have revolutionized financial transactions

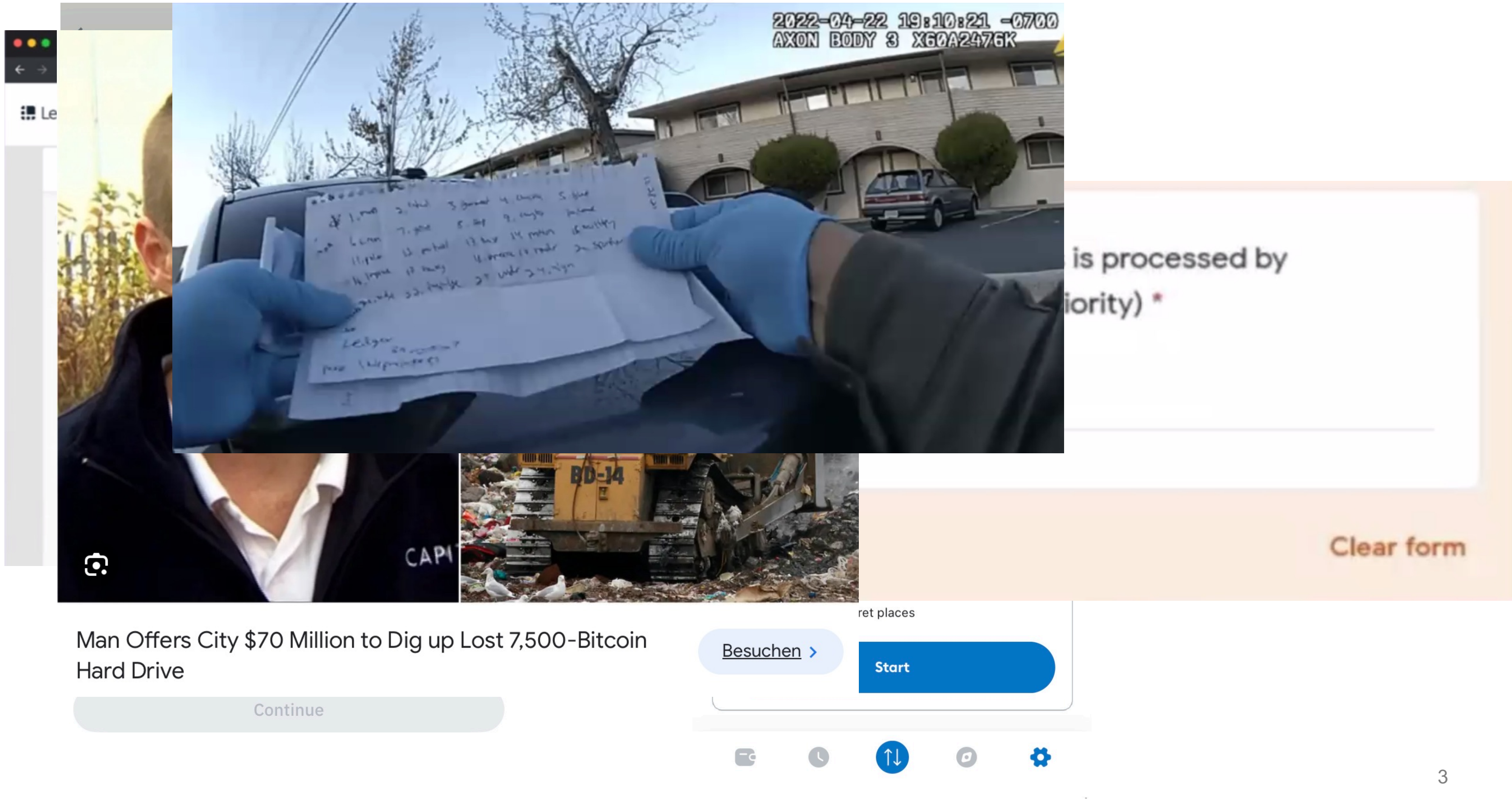
→ Surge in the development of mobile wallets for these assets

These crypto assets enable independence from centralized institutions like banks (and should prevent bank runs)

But …



High complexity and many pitfalls of crypto asset self-custody for average user



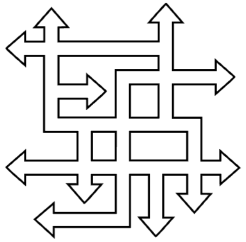
Motivation - Security and usability challenges of crypto asset self-custody

Digital assets such as cryptocurrencies have revolutionized financial transactions

→ Surge in the development of mobile wallets for these assets

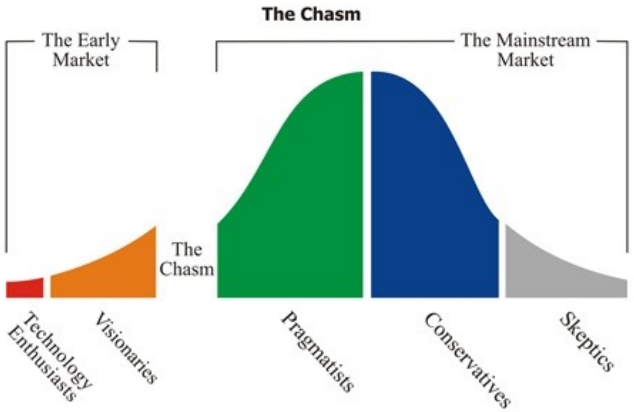
These crypto assets enable independence from centralized institutions like banks (and should prevent bank runs)

But …



High complexity and many pitfalls of crypto asset self-custody for average user

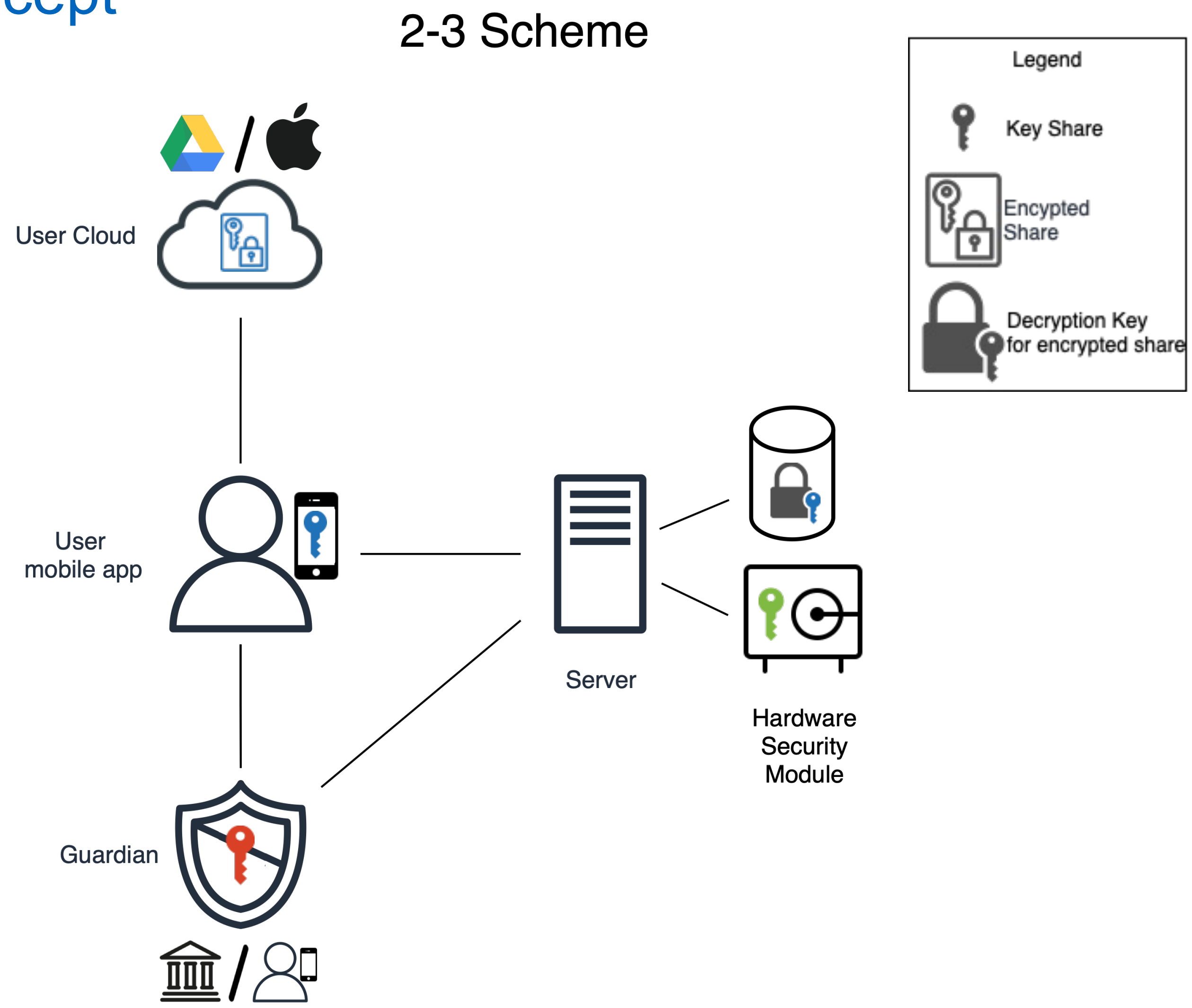
Contrary to the blockchain ethos, users leave assets on centralized exchanges.



Barrier for mass adoption

Basic signature scheme and recovery concept

- Multi Party Computation: Each party generates a share of a private key together with the other parties off-chain
- User co-signs transactions with service provider
- In case of censorship/bankruptcy of service provider or switching the mobile platform, the user can regain access to the funds through a guardian



Outline

1. Motivation and Introduction
2. Problem Statement
3. Research Questions & Results
4. Live Demo
5. Evaluation & Future Work

Problem Statement - Goal

- Wallet without need to write down private key mnemonics
- No single point of failure (private key)
- Further bring user experience closer to a custodial solution like on a bank account or an exchange (with functionalities like transaction limits, inheritence, ...)

Goal: Design of a secure and user error-free crypto asset management platform that is truly non-custodial and ensures asset recoverability in any scenario

Problem Statement

- Positive impact of MPC on security has been shown in the literature
  - But the impact on user experience and its interplay with security has not been explored

- Various possible setups of the signature scheme and recovery architecture with different implications on security and user experience
  - But an optimal one has not jet emerged
  - Room for improvement

Outline

Research Questions

RQ 1

How can inherent security and usability challenges in crypto wallets be technologically addressed and what design requirements, principles and features emerge for enhancing wallet solutions?

a) What challenges in digital asset management and transaction security are addressed by Multi-Party Computation (MPC) and Account Abstraction technologies?

b) How can we leverage MPC techniques to implement new features in crypto wallets, such as recoverability, transaction limits or inheritance of assets, while maintaining security and useability?

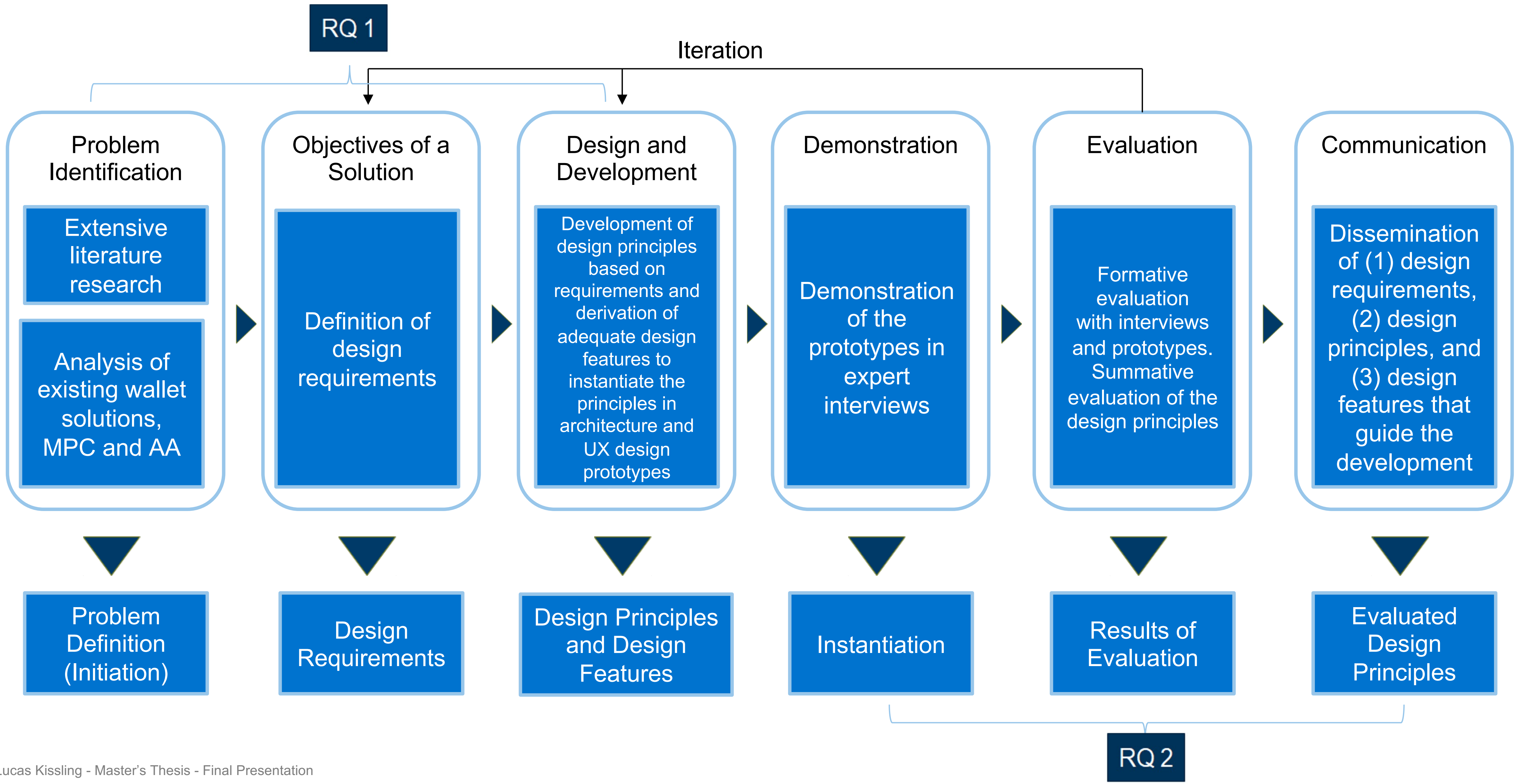
RQ 2

How can the application of Multi-Party Computation (MPC) in non-custodial mobile cryptocurrency wallets improve their security and user experience, thus enabling mass adoption of digital assets?

a) How do different recovery mechanisms and their associated threshold signature schemes (2-2 and 2-3) affect the security and user experience?

b) How is the security and user experience perceived compared to other non-custodial and custodial solutions

Design Science Research Approach based on Peffers et al.



Research Question 1

RQ 1

How can inherent security and usability challenges in crypto wallets be technologically addressed and what design requirements, principles and features emerge for enhancing wallet solutions?

a) What challenges in digital asset management and transaction security are addressed by Multi-Party Computation (MPC) and Account Abstraction technologies?

b) How can we leverage MPC techniques to implement new features in crypto wallets, such as recoverability, transaction limits or inheritance of assets, while maintaining security and useability?

RQ1: Initial Functional Requirements

- Based on
  - extensive literature review
  - user survey with 109 participants

FR-1	No seed phrase backups
FR-2	All standard functions of self-custodial wallets must be supported
FR-3	Damage containment
FR-4	Assets must not be lost if user passes away
FR-5	Integration of different crypto use cases (Storage and Payment)
FR-6	Payment in retail stores with crypto assets
FR-7	User can switch to another wallet without transacting from each address

RQ1: Initial Non-Functional Requirements

Usability:

NFR-1	Must be easy to navigate and understand features without getting stuck in the user flow
NFR-2	User must not get stuck during onboarding

Security:

NFR-3	Private key does not exist at any place at any time
NFR-4	No one else than the user shall be able to access the assets
NFR-5	Assets not censorable
NFR-6	Protection against theft of shares
NFR-7	Protection against spoofed addresses
NFR-8	Protection against fraudulent recovery attempt
NFR-9	Protection against collusion

Reliability and Availability:

NFR-10	Device can be lost
NFR-11	Recoverability if service provider not available
NFR-12	User can switch to other device and OS

RQ1: Key feature differences of SMPC and AA based wallets

	SMPC	Account Abstraction
Chain agnostic	Yes	No
Transaction costs	One transaction (lowest as with any EOA)	Costs per signature required + additional fee for contract execution
Time locks, limits, firewalls, ...	Enforced on application layer	Enforced by contract
Account creation costs	Free	High, depending on network
Signer anonymity	Yes	Guardians would be exposed as potential attack vector
Allowance	No	Yes
Send assets via link	No	Yes
Gas fee abstraction	No	Yes
Automatic recurring/continuous payments	No	Yes

SMPC as base layer

+

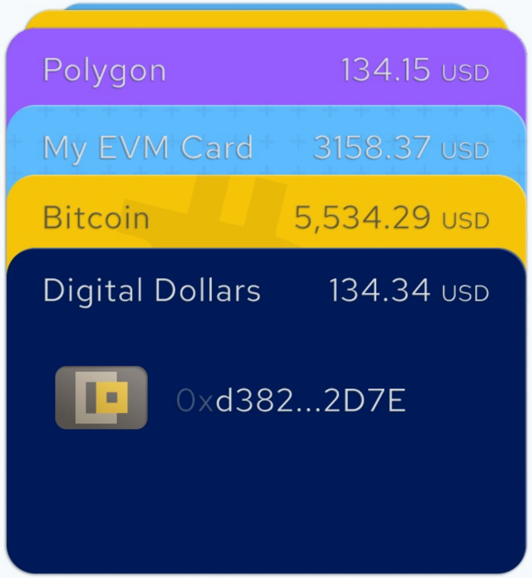
AA on top of SMPC for payment use case

RQ1: Design Features and Principles (excerpt)

Design Principles:

- UX like a custodial solution or banking app
- No single point of failure & redundancy
- For Non-Crypto-Natives familiar wallet look
- Payments: Practicability in daily life use cases and seamless as Apple Pay

Design Features:



Card stack incorporating different use cases and a combination of SMPC with AA

Guardian system for social recovery and independence from service provider

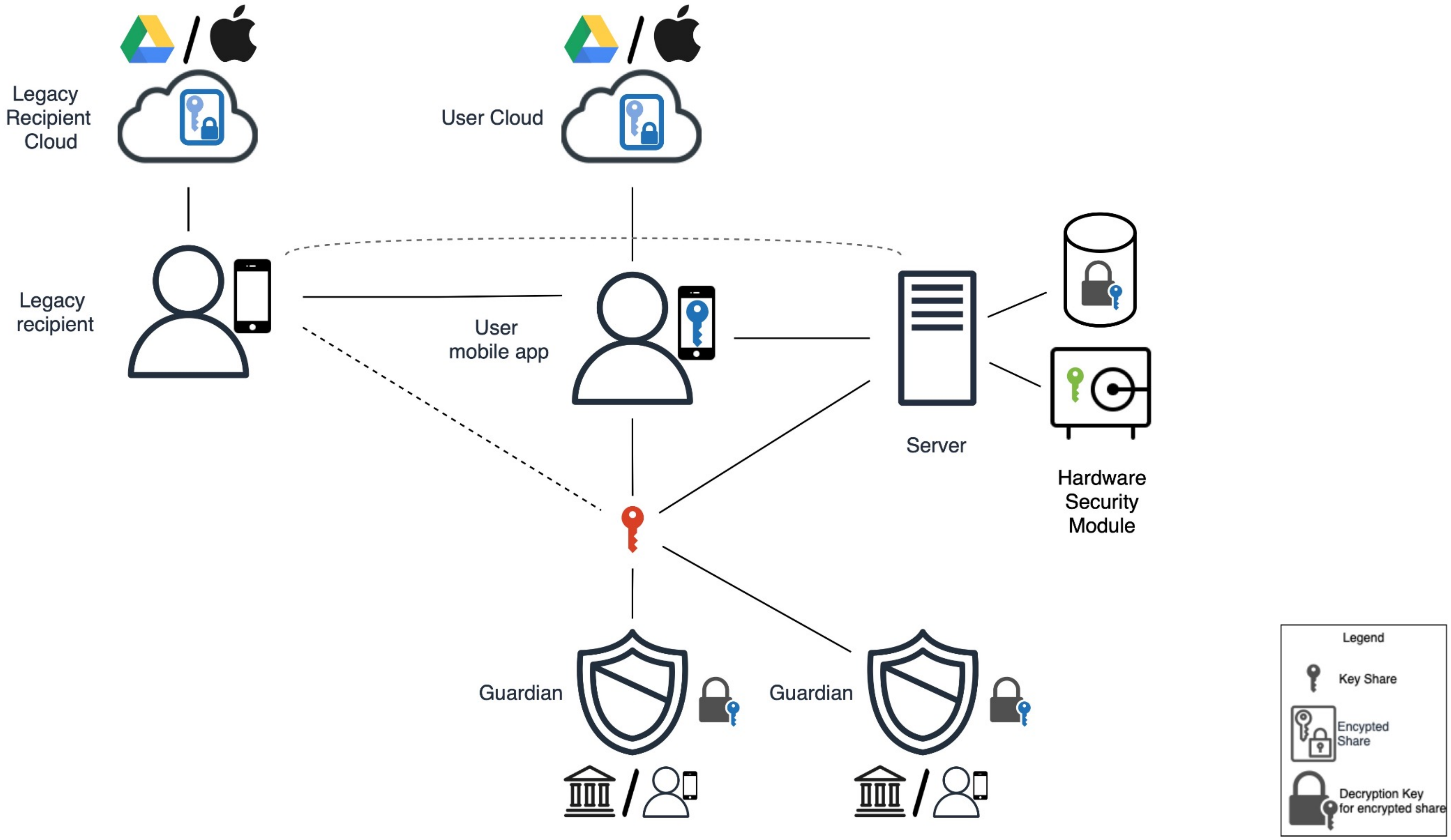
TSS and authentication at co-signing service provider with email and device ID instead of seed phrase/private key

Transaction limits

Wallet inheritance (Legacy Transfer)

Instant merchant payment system

RQ1: Initial 2-of-3 TSS and Recovery Architecture with Inheritance



Research Question 2

RQ 2

How can the application of Multi-Party Computation (MPC) in non-custodial mobile cryptocurrency wallets improve their security and user experience, thus enabling mass adoption of digital assets?

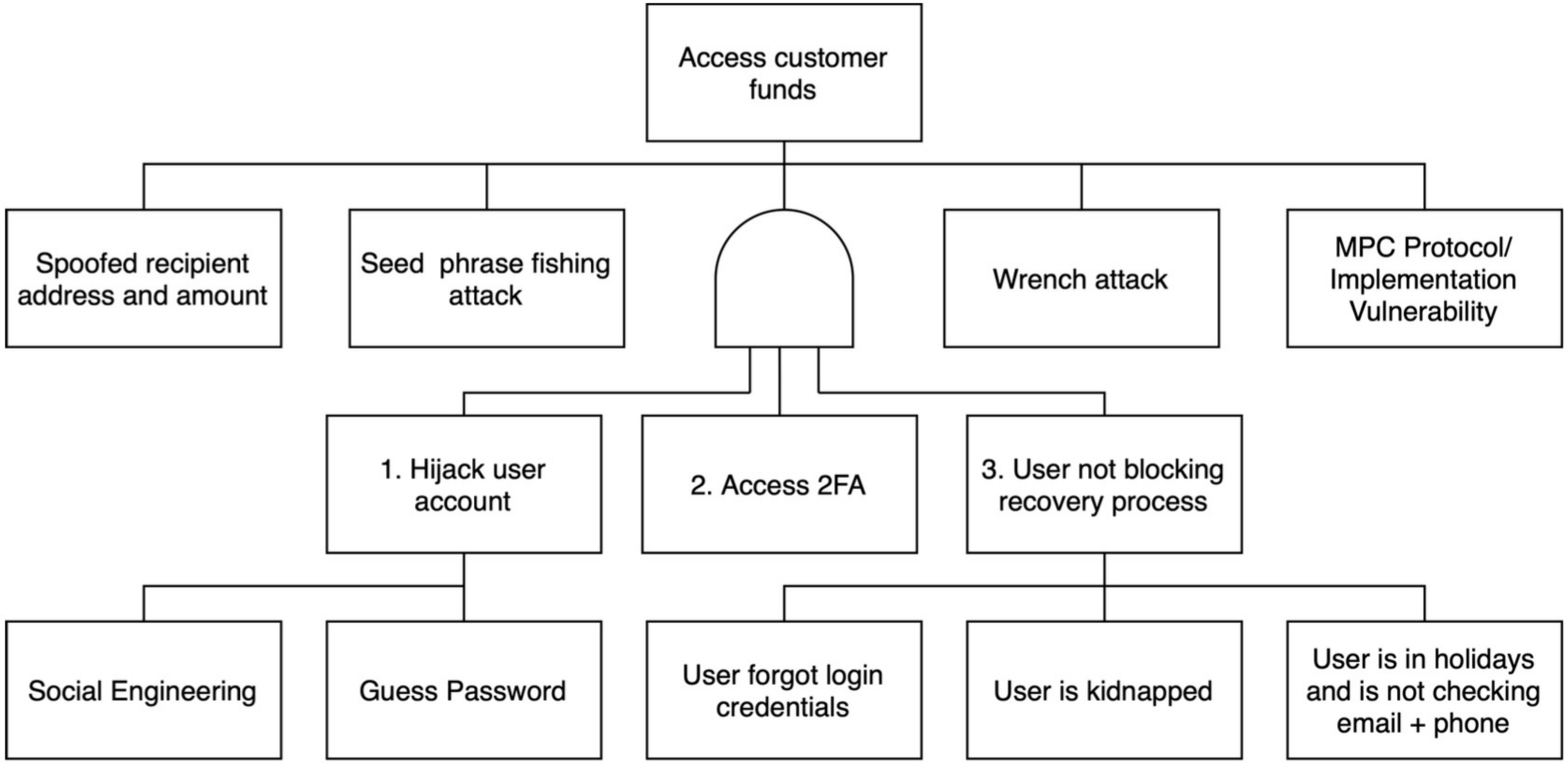
a) How do different recovery mechanisms and their associated threshold signature schemes (2-2 and 2-3) affect the security and user experience?

b) How is the security and user experience perceived compared to other non-custodial and custodial solutions

RQ2: Expert Interviews

- Demonstration of the prototypes in expert interviews
- We conducted 5 semi-structured expert interviews
- Attack and fault tree analysis:

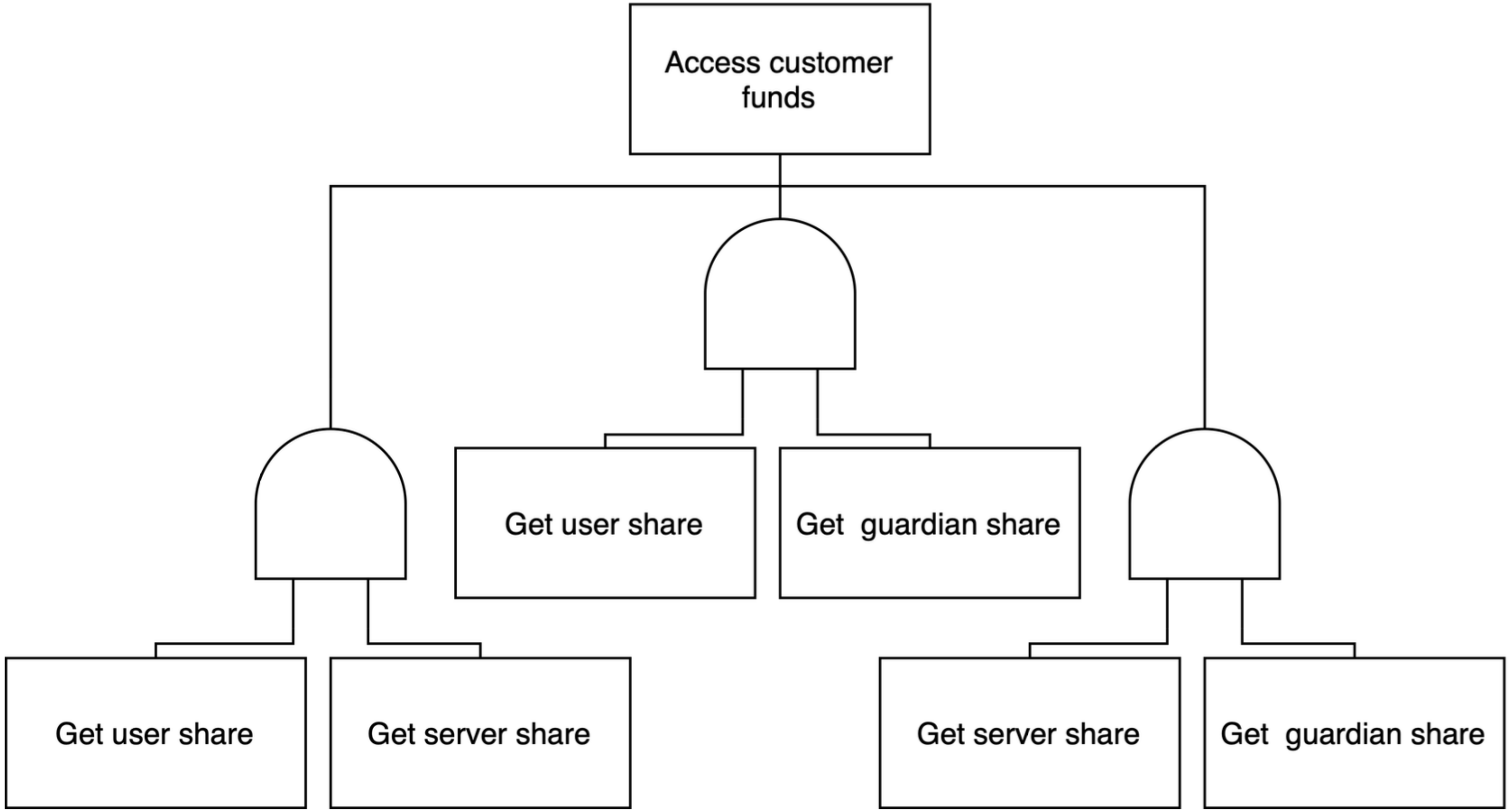
Attack tree (1/2)



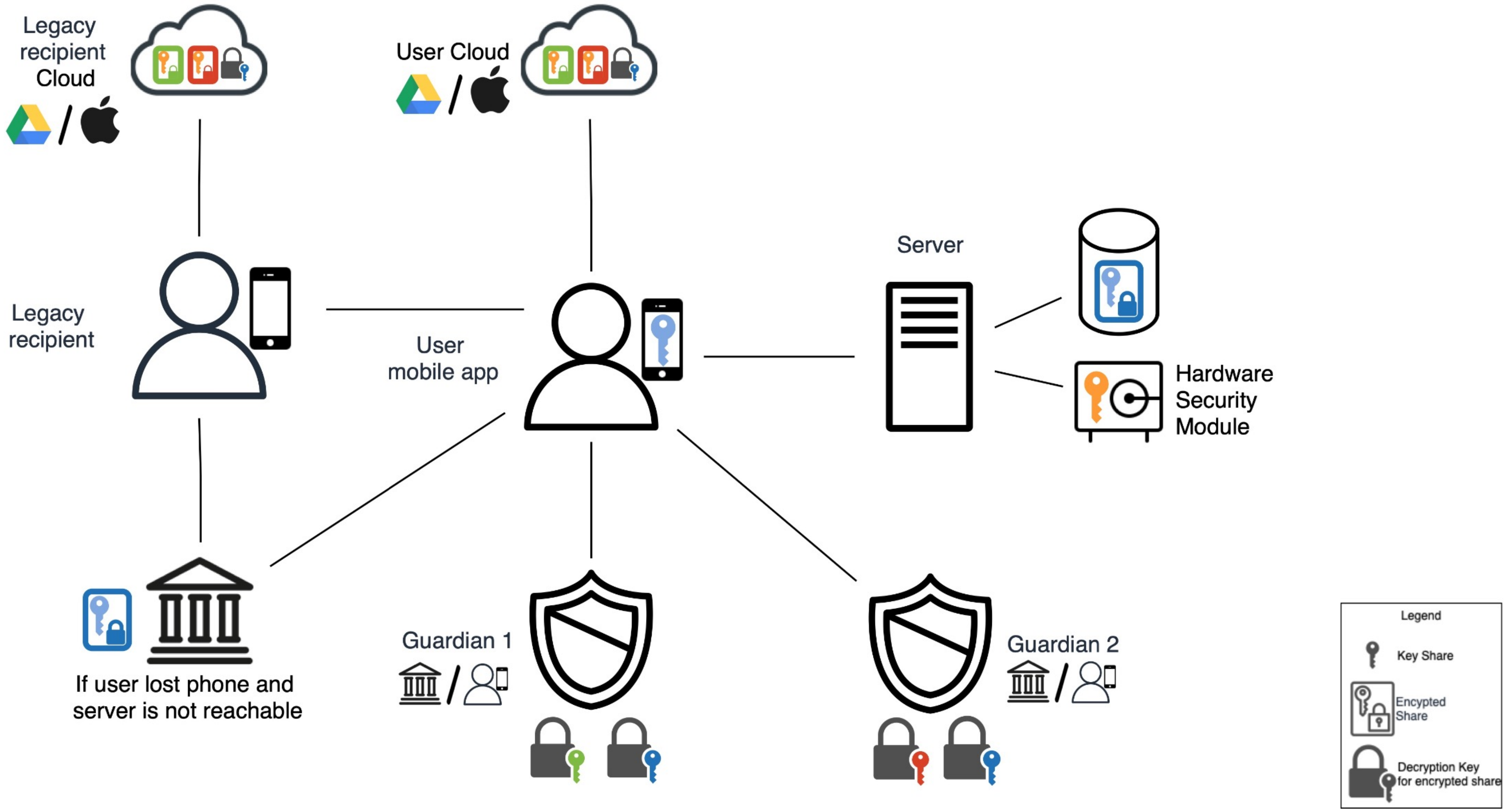
RQ2: Expert Interviews

- Use of proactive SMPC protocols necessary → Additional Design Feature for NFR-6

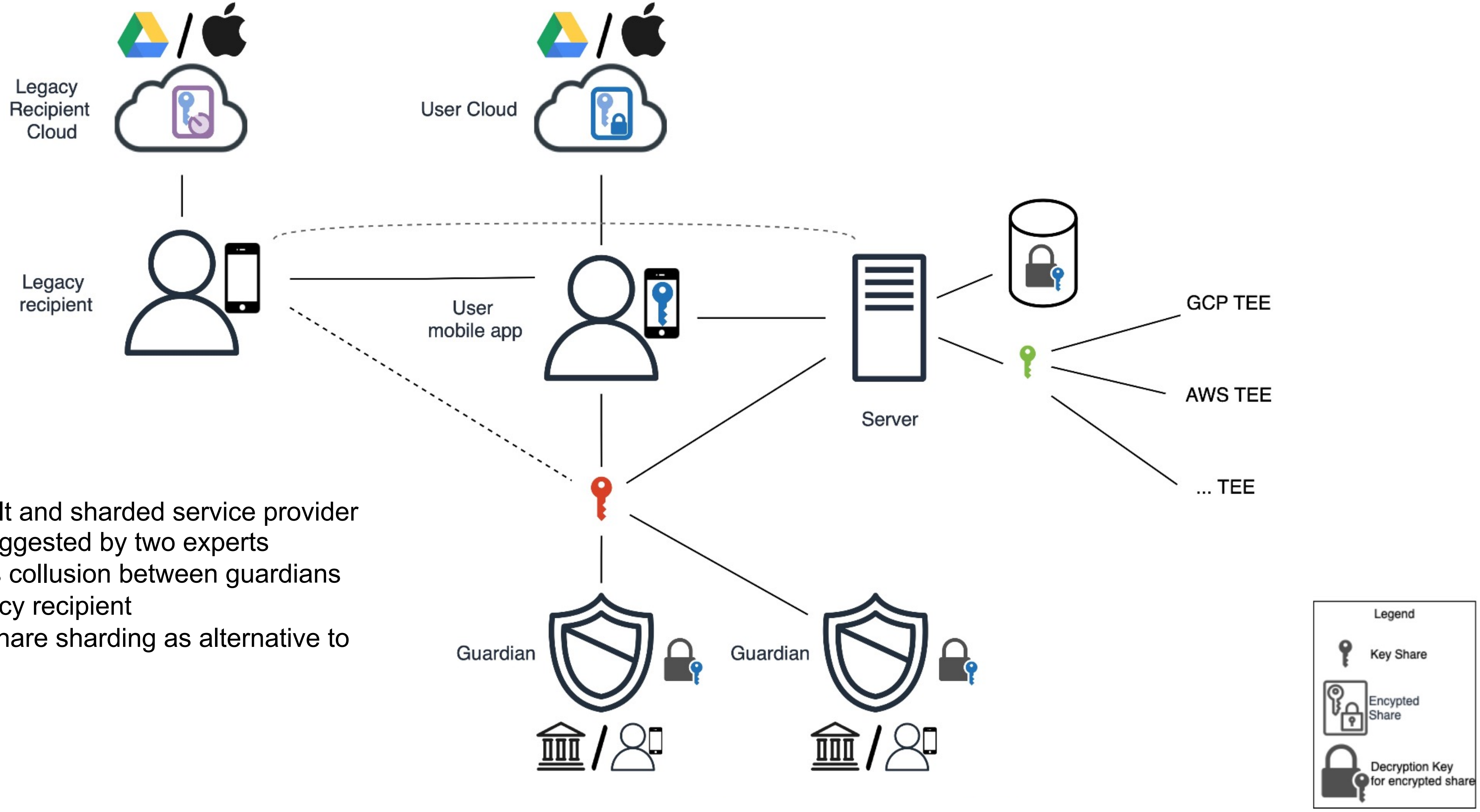
Attack tree (2/2)



RQ2: 2-of-2 TSS and Recovery Architecture with Inheritance



RQ2: 2-of-3 Architecture with Timevault and Sharded Service Provider Share



- Timevault and sharded service provider share suggested by two experts
- Prevents collusion between guardians and legacy recipient
- Server share sharding as alternative to HSM

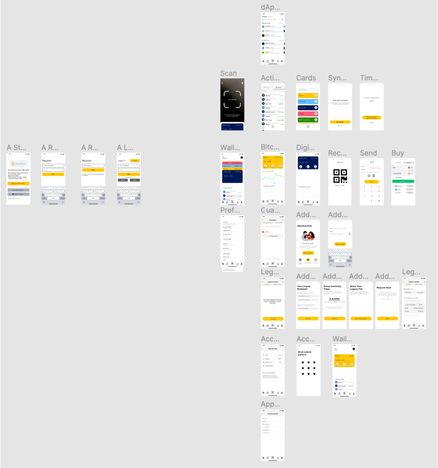
RQ2: User Interviews

- Semi-structured user interviews
- Questions of interview guide following:
  - A. General Information
  - B. Initial Reactions
  - C. User Experience
  - D. Security Perception
  - E. Optimal Balance of Ease of Use and Security Perception

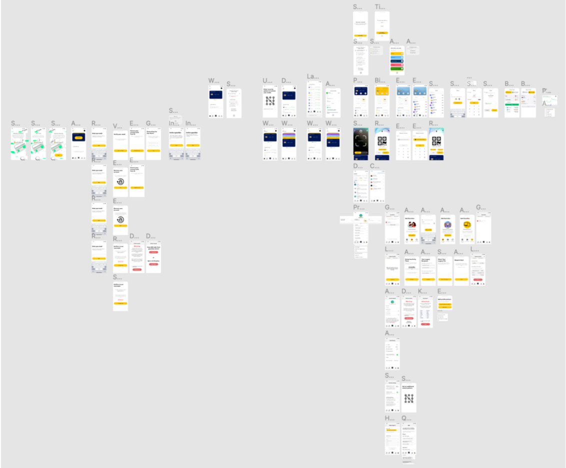
- All users of custodial, self-custodial hot and cold wallets were convinced by the superiour combination of security and ease of use
- Total newcomers highlighted the „intuitive design“ and quickly navigated to all functions

Iteration 1
(9 participants)

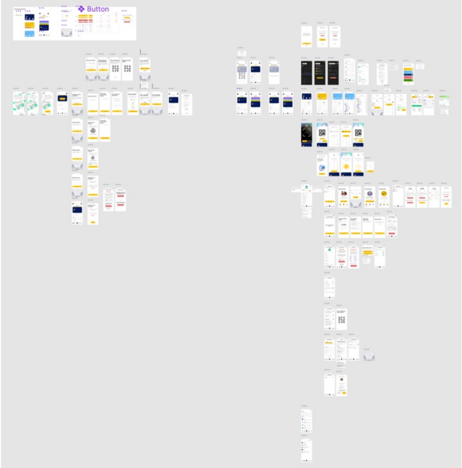
Iteration 2
(10 participants)



Artifact 1



Artifact 2



Artifact 3

RQ2: System Usability Scale

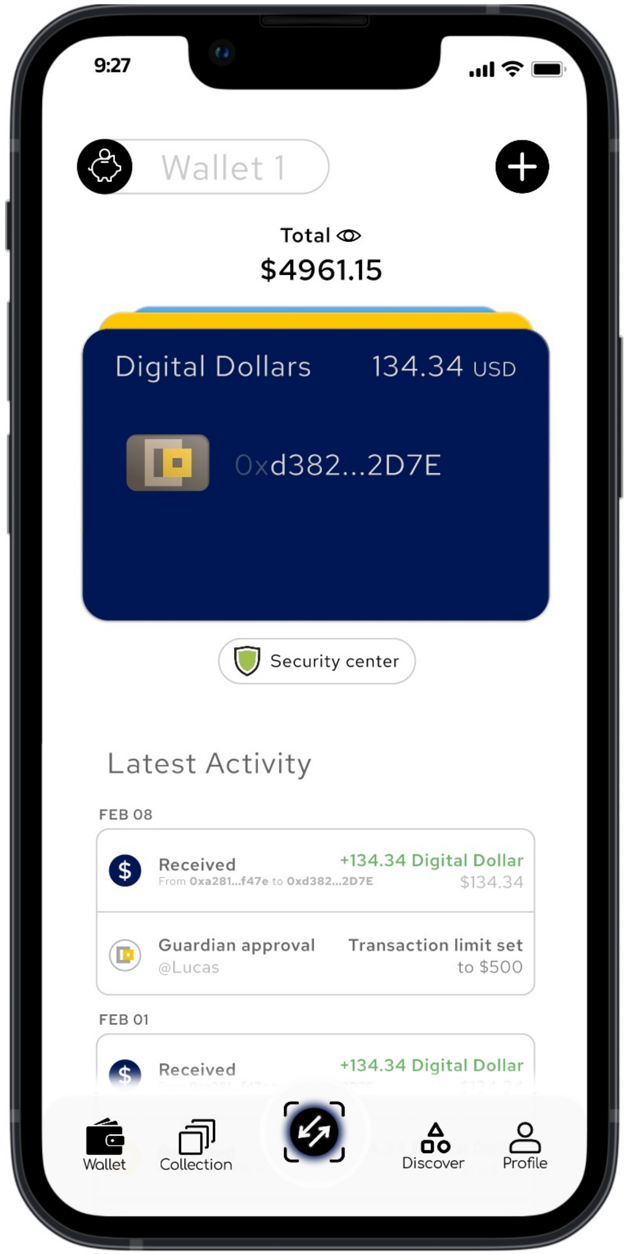
- SUS Score: 81.5
- Users with
  - No experience: 29%
  - Some experience: 41%
  - Experienced: 29%

Questionnaire

- I think that I would like to use this system frequently.
- I found the system unnecessarily complex.
- I thought the system was easy to use.
- I think that I would need the support of a technical person to be able to use this system.
- I found the various functions in this system were well integrated.
- I thought there was too much inconsistency in this system.
- I would imagine that most people would learn to use this system very quickly.
- I found the system very cumbersome to use.
- I felt very confident using the system.
- I needed to learn a lot of things before I could get going with this system.

Outline

Live Demo


9:27
Wallet 1
Total
$4961.15
Digital Dollars
134.34 USD
0xd382...2D7E
Security center
Latest Activity
FEB 08
Received
From 0xa281...f47e to 0xd382...2D7E
+134.34 Digital Dollar
$134.34
Guardian approval
@Lucas
Transaction limit set to $500
FEB 01
Received
+134.34 Digital Dollar
Wallet
Collection
Discover
Profile

Outline

1. Motivation and Introduction
2. Problem Statement
3. Research Questions & Results
4. Live Demo
5. Evaluation & Future Work

Evaluation and Future Work

Evaluation:

-All requirements fulfilled + 2 additional added

*The 2-of-2 architecture theoretically centralizes the private key at a single location, but in form of encrypted shares

*With the 2-of-3 architecture and less than three guardians a guardian could gain access to the user’s funds by stealing the encrypted share from the user cloud by physically accessing the users devices

-All user groups are very interested and convinced once they understood the concept

Limitations & Constraints:

-Due to a lack of SUS assessments of other wallet types, we could not compare our solution quantitatively with them

-Practical boundaries of available MPC protocols

Future Work:

-Assessment of other wallet solutions using SUS to compare them to our solution

-Development and extensive testing of the individual components, such as inheritance with Timevault or the instant merchant payment system

Lucas Kissling
lucas.kissling@tum.de

Technische Universität München
Faculty of Informatics
Chair of Software Engineering for Business
Information Systems

Boltzmannstraße 3
85748 Garching bei München

Tel +49.89.289.17132
Fax +49.89.289.17136

matthes@in.tum.de
wwwmatthes.in.tum.de

Backup

Functional Design Requirements, Principles and Features

	Design Requirement	Design Principle	Design Feature
FR-1	No seed phrase backups	Familiar UX to custodial solutions like a crypto exchange or banking account	TSS and authentication at co-signing service provider with email and device ID instead of seed phrase/private key
FR-2	All standard functions of self-custodial wallets must be supported		Send/receive crypto assets via QR-code, Buy, Swap, …
FR-3	Damage containment (Damage of loss of assets must be contained in case of unauthorised access or user error)		Transaction Limits
FR-4	Assets must not be lost if user passes away		Legacy recipient system
FR-5	Integration of different crypto use cases (Stable coins payments as well as investment in Bitcoin or EVM coins/tokens		Tokens clustered under addresses and cards representing the address
		Practicability in daily life use cases and seamless as Apple Pay	Smart contract based payment card (Gas fee abstraction, send assets via link, recurring payments, …)
FR-6	Payment in retail stores with crypto assets		Instant Tap to Pay
FR-7	User can switch to another wallet without transacting from each address	1. No platform lock-in 2. No single point of failure	Private key derivation without return
FR-8 (From attack tree analysis)	Plausible deniability of assets	To be not bound to unlock via pattern instead if FaceID or finger print, additional plausible options to deny actual wallet must accompany a deniability feature	Dummy wallet
			Unlock Pattern
			Hide wallets
			Hide balance before revealing the wallet

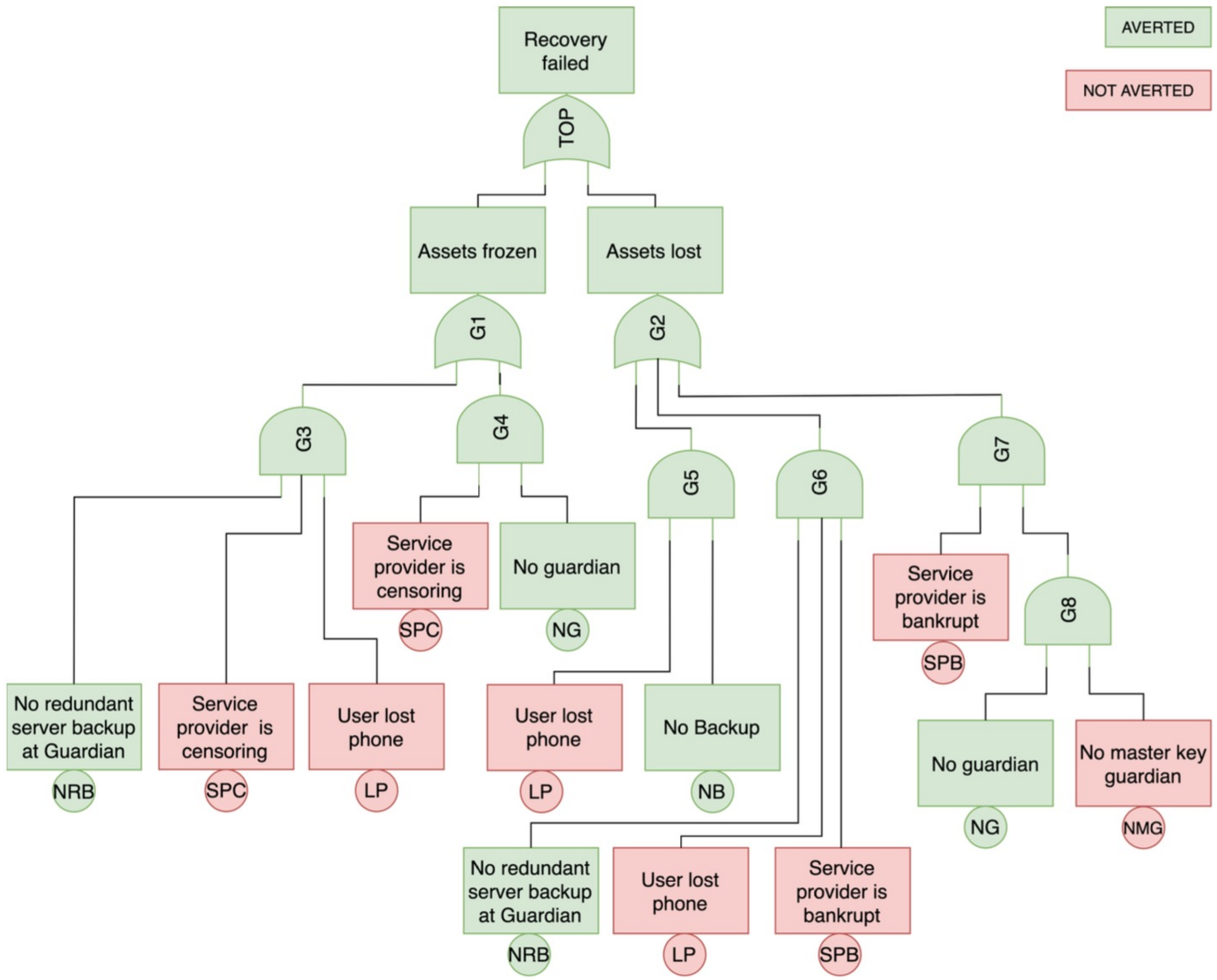
Non-Functional Design Requirements, Principles and Features (1/2)

	Design Requirement	Design Principle	Design Feature
	Usability:		
NFR-1	Must be easy to navigate and understand features without getting stuck in the user flow	For non-crypto natives familiar wallet look	Design based on credit cards and Apple wallet
NFR-2	User must not get stuck during onboarding	Explaining, short, engaging by outlining what lies ahead	Wallet preview and explanations
	Security:		
NFR-3	Private key does not exist at any place at any time	1. No single point of failure (including in future post-quantum scenarios) 2. Security by Design	2-of-2 or 2-of-3 TSS
NFR-4	No one else than the user shall be able to access the assets (non-custodial)	1. non-custodial, while at the same time time technical knowledge is not required to avoid user error 2. Security by Design	
NFR-5	Assets not censorable		Guardians
NFR-6	Protection against theft of shares	Security by Design	Proactive SMPC protocols
			Hardware Security Module for server share
		1. No single point of failure 2. Redundancy 3. Security by Design	Server Share Sharding
			Encrypted backups with decryption key stored at another party
NFR-7	Protection against spoofed addresses	Security by Design	2FA Email notification with address shown to allow co-signing
			Warning of unknown/unused addresses
NFR-8	Protection against fraudulent recovery attempt		2FA, time lock period, user cloud backup
NFR-9	Protection against collusion		2-of-2 TSS
			Independent guardians
			Legacy file with timevault for 2-of-3 TSS

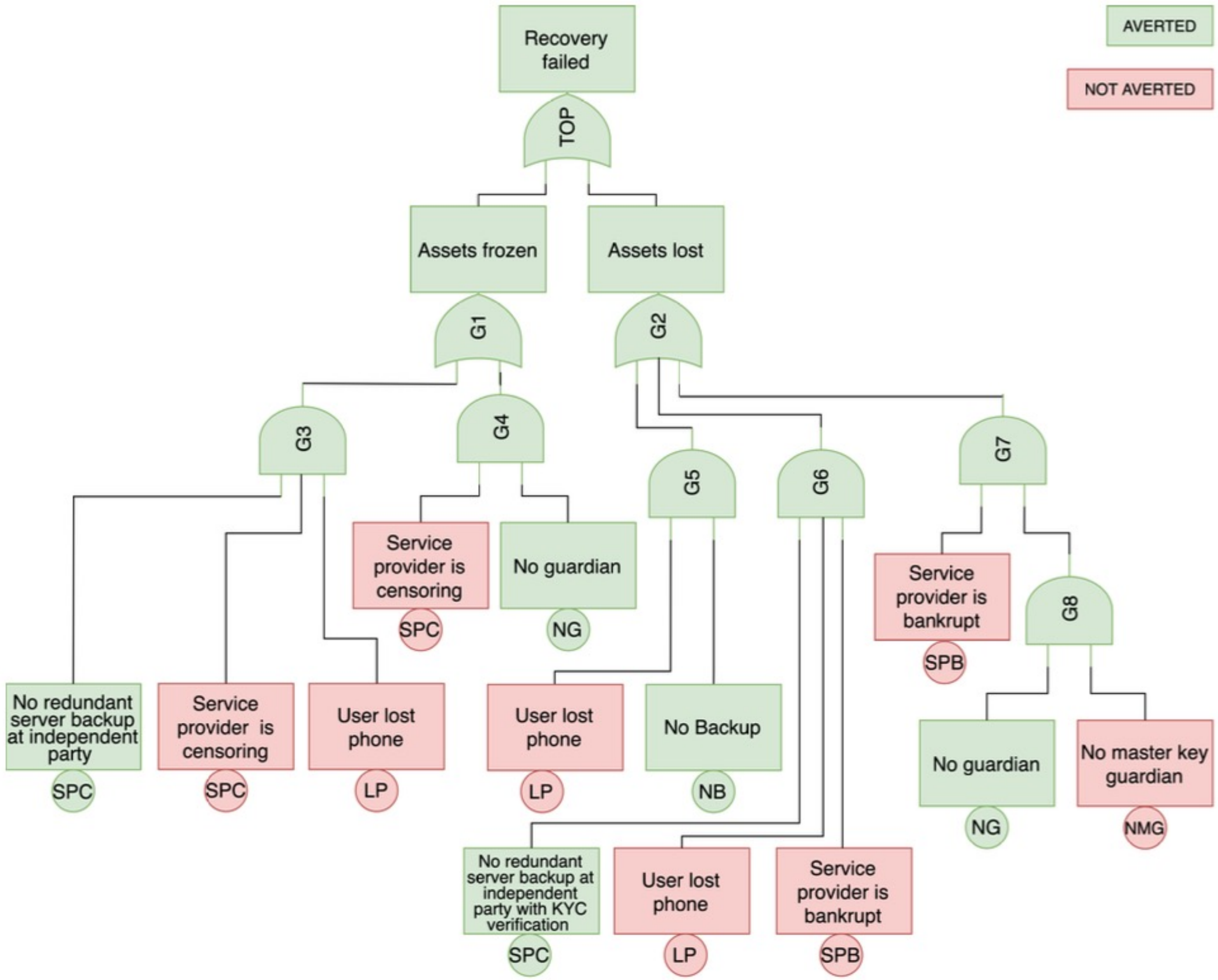
Non-Functional Design Requirements, Principles and Features (2/2)

	Design Requirement	Design Principle	Design Feature
	Reliability and Availability:		
NFR-10	Device can be lost	1. No single point of failure 2. Redundancy	Backup at user cloud with complement stored server side
NFR-11	Recoverability if service provider not available		Backup at user cloud complemented with guardians for 2-of-2 TSS or guardian share in 2-of-3 TSS
NFR-12	User can switch to other device and OS		Server side backup complemented with guardians for 2-of-2 TSS or guardian share in 2-of-3 TSS
NFR-13 (From fault tree analysis)	Device can be lost when service provider is not available at the same time		Backup complemented with guardians is stored at an institutional custodian requiring KYC for 2-of-2 TSS or guardian with redundant server backup for 2-of-3 TSS

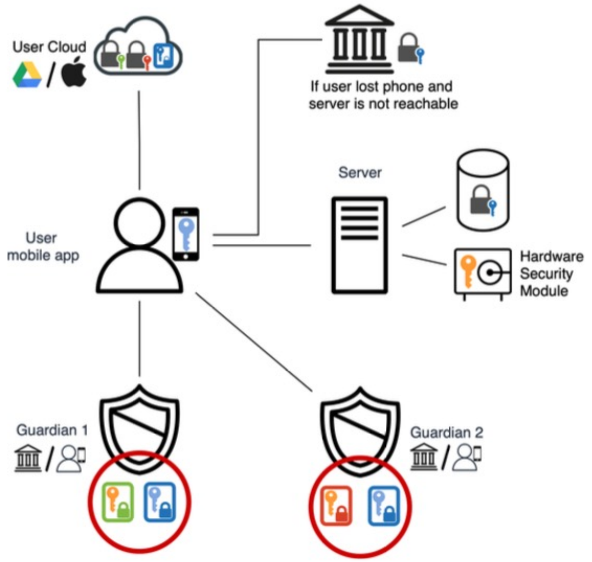
2-of-3 Recovery Architecture Fault Tree



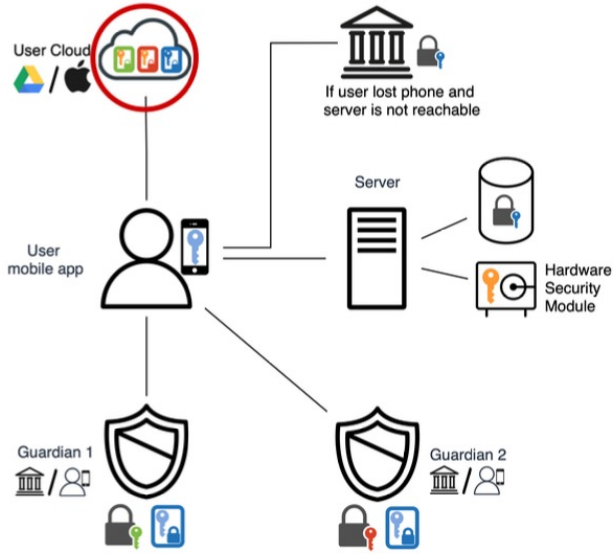
2-of-2 Recovery Architecture Fault Tree



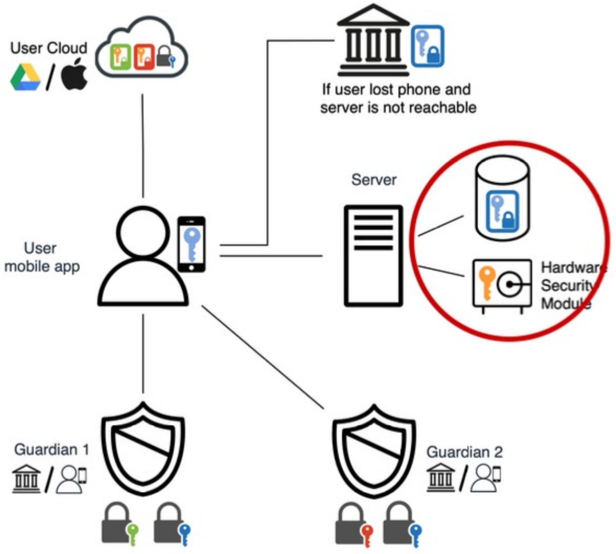
Recovery Architecture Backup Distributions



(a) All Shares on Guardian Devices.



(b) All Shares on User Cloud.



(c) Shares Separated by Service Provider.

User Interview Perceived UX and Security Taxonomy

<table>
<tr><th>Custody</th><th>Account</th><th colspan="2">Authorisation</th><th>Retail Wallet Exapmple</th><th>Security</th><th>Useability</th></tr>
<tr><td rowspan="7">A) Self-Custodial Wallet</td><td rowspan="6">A.1.) EOA</td><td colspan="2">A.1.1.) MPC TSS</td><td>Our wallet solution</td><td>+ Secure and innovative
+guardian system
+inheritance
+physical theft protection</td><td>+Intuitive UI
+Easy Setup
+Sccessible for beginners
+Seamless mobile experience</td></tr>
<tr><td rowspan="5">A.1.2.) Private Key</td><td rowspan="4">Hot Wallet</td><td>Trust Wallet</td><td rowspan="4">+ Full controll
- Single point of failure</td><td rowspan="4">+ Intuitive UI
+ Defi suited</td></tr>
<tr><td>MetaMask</td></tr>
<tr><td>Rabby</td></tr>
<tr><td>GME Wallet</td></tr>
<tr><td>Cold Wallet</td><td>Ledger</td><td>+ Highly secure</td><td>- Complicated UI</td></tr>
<tr><td colspan="3">A.2.) SCA</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td colspan="4" rowspan="5">B) Custodial Wallet</td><td>Binance</td><td rowspan="5">- Centralization
- Collaps/ Hack</td><td rowspan="5">- Cluttered</td></tr>
<tr><td>Coinbase</td></tr>
<tr><td>Bison</td></tr>
<tr><td>Bittrex</td></tr>
<tr><td>CoinSwitch</td></tr>
</table>

Onboarding & Key Generation