Hybrid Custody using Layer 2 Smart Vaults for Cold Storage

Dr. Praveen Baratam
CoinVault.tech
Updated 28th Aug, 2024

Summary:

Hybrid Custody using **Layer 2 Smart Vaults** can make Bitcoin and similar Altcoins unstealable, unlosable, and unconfiscatable for all practical purposes.

Smart Vaults are Layer 2 constructs on top of n-of-n **MultiSig** transactions allowing us to spend from the Smart Vault using any number of the private-keys (1-to-n of n) i.e. we can spend from the Smart Vault even with just 1 private-key or m private-keys (m ≤ n) as needed. Over and above that, spending from Smart Vaults is a two-step process with unlock and spend steps allowing some private-keys and combinations of private-keys to override unlock-and-spend attempts by other private-keys and combinations of private-keys with lower priority as per the Smart Vault's configuration.

Eg. a typical Smart Vault locked with two private-keys (*a and b where a+b > a > b*) and unlocked using one stolen private-key (*b*) can be recovered using *a* alone (*a > b*) or both *a and b* private-keys (*a+b > b*).

Smart Vaults achieve this flexibility using **Zero Knowledge Proofs** mediated by a set of **Partially Signed [Bitcoin] Transactions (PSBTs)** cleverly distributed among the Smart Vault participants during setup.

To make things even better, **Hybrid Custody,** a service framework built using Smart Vaults, combines the best of self-custody and managed-custody for everyone.

With 2 participants, The Owner and the Hybrid Custody provider, the Hybrid Custody framework creates a Smart Vault with respective private keys (*o and h where o+h > o > h)* to protect the Owner's tokens even when his private-key is lost or stolen. Here the private-key priorities are set such that the Owner can spend/transfer his tokens using just his private-key (*o*) similar to self-custody. Moreover, when the Owner's private-key (*o*) is lost, the Owner can summon the Hybrid Custody provider to sign a recovery transaction using just the Hybrid Custody provider's private-key (*h*) and recover his tokens. Finally, if the Owner's private-key (*o*) is stolen and used to unlock the Smart Vault, the Owner can cosign a recovery transaction with the Hybrid Custody provider and recover his tokens (*o+h > o*).

Background:

Cryptocurrencies in general are secured with public-key cryptography where a public-key or a derivative of it serves as a wallet/address containing tokens in the underlying

blockchain/ledger and the owner has to prove to the corresponding network and community that he owns the respective private-key with a cryptographic signature to transfer/spend his tokens.

From an end user's perspective, it boils down to keeping his private-keys safe and away from prying eyes to secure his tokens. If one misplaces/loses his private-keys, his tokens are lost forever. Similarly, if anyone can steal his private-keys, his tokens can also be stolen without any chance of recovery.

While securing one's private-keys might seem trivial at first, ensuring safety i.e. preventing data loss (lost private-key) along with absolute security i.e. preventing data theft (stolen private-key) is very difficult if not impossible. It is a balancing act and one has to sacrifice a bit of security to increase safety and vice versa. Eg. Increasing the number of places where you store copies of your private-keys to prevent accidental loss increases safety but decreases security as they can now be stolen from more places than before.

Several methods and devices were proposed and implemented to alleviate this problem at least partially but none are as flexible and secure as we would like them to be. Furthermore, the devices, platforms, and frameworks used to generate, store, and sign transactions with private keys are susceptible to both known and unknown vulnerabilities. Relying solely on these contraptions to secure our tokens can be risky, to say the least.

Comparison with MultiSig:

MultiSig (m-of-n where m < n) Vaults can tolerate the loss and theft of a subset of private-keys used to create them and are often touted as a solution to the lost/stolen private-keys problem. But, MultiSig Vaults come with the burden of managing multiple private-keys when most crypto holders are shying away from setting up and managing a single private-key due to the complexities involved. We can delegate the responsibility of additional private-keys to near and dear or third parties to simplify private-key management and to prevent ourselves from becoming a single point of failure, but they can then restrict our ability to spend and transfer our tokens as we see fit or can conspire and steal our coins!

In the end, we either need to dilute our control over our tokens and risk insider fraud or accept ourselves as a single point of failure for our tokens with MultiSig Vaults - which is a choice we don't need to make with Hybrid Custody using Smart Vaults. Moreover, there is no way to implement any kind of override, recovery, or clawback using MultiSig constructs making Smart Vaults exponentially better than MultiSig.

Comparison with MPC:

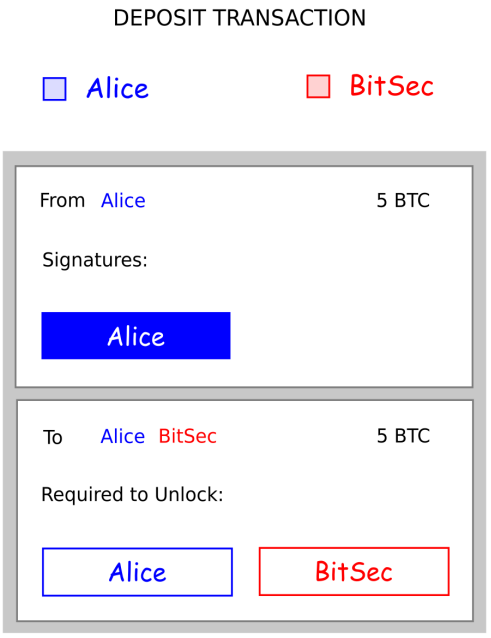
MPC Vaults are similar to MultiSig Vaults but execute the quorum logic off-chain and in our opinion are worse than MultiSig for most use cases. Importantly, MPC Vaults give plausible deniability to participants signing malicious transactions as we cannot trace which fragments were used to sign a spending transaction once fully signed.

Solution:

The following describes Hybrid Custody using Smart Vaults, in its simplest form, between the First Party and Second Party participating in a cryptocurrency network/system to drastically reduce the probability of loss and theft of the First Party's tokens. Here the Second Party is acting as a Hybrid Custody provider to the First Party. This arrangement and method can be similarly and analogously extended to even more parties as may be necessary.

The method presumes that time-locks for transaction outputs are available for the crypto-currency system of interest. Relative time-locks (CheckSequenceVerify) similar to the one described in Bitcoin Improvement Proposal 112 are more desirable than absolute time-locks (CheckLockTimeVerify) similar to the one described in Bitcoin Improvement Proposal 65. The subsequent discussion assumes relative time-locks are available for the cryptocurrency of interest even though similar functionality can be devised using absolute time-locks.

Figure 1



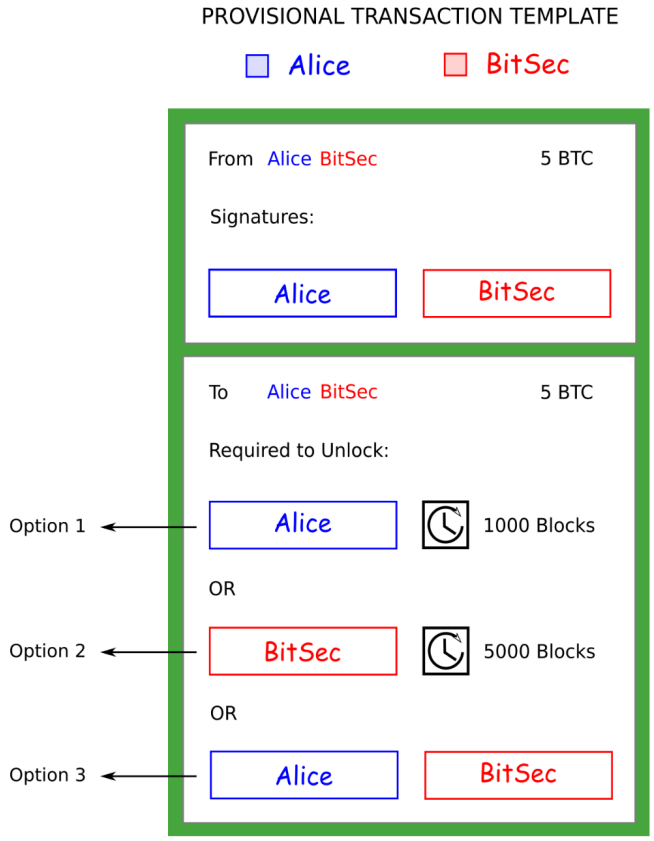
Setup:

1. At inception, the First Party creates a transaction similar to the one depicted in **Figure 1** (Alice as the First Party and BitSec as the Second Party)**,** hereafter called the Deposit Transaction, in which the First Party transfers an arbitrary sum of tokens it owns to a multi-signature address but does not yet sign or broadcast it. The multi-signature address in the Deposit Transaction requires the following signatures to authenticate and spend/transfer from it:

a. First Party's Private Key generated Signature
b. Second Party's Private Key generated Signature

2. Then, the First Party creates a second transaction, hereafter referred to as Provisional Transaction, as depicted in **Figure 2** (Alice as the First Party and BitSec as the Second Party), spending all the tokens sent to the multi-signature address in the Deposit Transaction, and sends a copy of the unsigned Provisional Transaction to the Second Party. Please note that the Provisional transaction is spending from an unconfirmed Deposit Transaction.

Figure 2



3. Then, the Second Party adds its Private Key generated signature to the unsigned Provisional Transaction received from the First Party and sends the partially signed Provisional Transaction back to the First Party.

4. Parallelly, the First Party also adds its Private Key generated signature to the unsigned copy of the Provisional Transaction it created and sends the partially signed Provisional Transaction to the Second Party.

   Note: The signatures used in this scheme sign the transactions similar to SIGHASH_ALL or SIGHASH_SINGLE in Bitcoin protocol.

5. At this point, the First Party has the partially signed Provisional Transaction with the Second Party's Private Key generated signature already added to it and the Second Party has the partially signed Provisional Transaction with the First Party's Private Key generated signature already added to it.

6. Then, the First Party signs and broadcasts the Deposit Transaction it created to the cryptocurrency network/system completing the setup process. The whole process is outlined in **Figure 3** (Alice as the First Party and BitSec as the Second Party).

7. Once the Deposit Transaction is confirmed, both First Party and Second Party start monitoring the Cryptocurrency network directly and/or indirectly (using third-party services) for transactions referencing the Multi-Signature output address of the Deposit Transaction to detect any security breach and foul play.

8. Subsequently, the First Party, at its discretion, can add its Private Key generated signature to the partially signed Provisional Transaction with the Second Party's Private Key generated signature already added to it and broadcast a fully signed and valid Provisional Transaction to the cryptocurrency network/system when necessary.

9. Similarly, the Second Party can add its Private Key generated signature to the partially signed Provisional Transaction with the First Party's Private Key generated signature already added to it and broadcast a fully signed and valid Provisional Transaction to the cryptocurrency network/system when necessary.

10. To sum it up, either party can add missing signatures to the partially signed Provisional Transaction in their possession, broadcast the same when necessary, and unlock the Smart Vault.

11. Whenever the First Party or the Second Party wants to terminate this arrangement and transfer the tokens from the Smart Vault created above, it can sign (add the missing signatures) the partially signed provisional transaction with it and broadcast a fully signed and valid provisional transaction to the network and unlock the Smart Vault. Either party can also ask the other party to do the same if its private-key is lost.

12. Once the provisional transaction is confirmed, the First Party or the Second Party, either unilaterally or in coordination with the other if they suspect foul play, can create and broadcast another transaction transferring the tokens from the Provisional Transaction to a desired address using the respective options of the Provisional Transaction.

Figure 3

Alice creates a **Deposit Transaction** transferring Bitcoin to a Multisig Address requiring Signatures generated by Alice's Private Key and BitSec's Private Key to spend from it

t_0

Alice creates a **Provisional Transaction** as per Smart Vaults **Provisional Transaction Template** spending all the Bitcoin from the Deposit Transaction and transmits a blank copy of the Provisional Transaction without any signatures to BitSec

t_1

BitSec receives a blank copy of **Provisional Transaction** without any signatures from Alice

Alice adds the **Signature** generated using her Private Key to the unsigned Provisional Transaction and transmits it to BitSec

t_2

BitSec adds the **Signatures** generated using its Private Key to the unsigned Provisional Transaction and transmits it back to Alice

Alice receives **Partially Signed Provisional Transaction** from BitSec

t_3

BitSec receives **Partially Signed Provisional Transaction** from Alice

Alice broadcasts the Signed **Deposit Transaction** to **Bitcoin Network**

t_4

Discussion

Common attacks on data safety and security include ransomware, trojans, man-in-the-middle, etc. Most attacks either compromise **data safety** by denying/destroying access to one's private-keys/secrets (ransomware, computer virus, etc.) or **data security** by gaining access to one's private-keys through whatever means (trojans, man-in-the-middle, cloud backdoors, etc.) or both.

The above discussed scheme/protocol with deposit and partially signed provisional transactions addresses many contingencies such as loss or theft of either party's private-keys while drastically reducing the risk of the First Party losing its tokens. We refer to this arrangement as **"Hybrid Custody"** using Smart Vaults, as it combines the best of self-custody and managed custody i.e. the First Party has absolute control of its private-keys and tokens just like self-custody and can recover its token even when all its private keys are lost or stolen as if it opted for fully managed custody!

Whenever the First Party or the Second Party realizes that it has lost its private-key, it can request the other party to add the missing signatures to the partly signed Provisional Transaction with it and broadcast the fully signed and valid Provisional Transaction to unlock the Smart Vault and initiate recovery. Once the Provisional Transaction is confirmed, the other party can complete the recovery by signing and broadcasting another transaction transferring the respective tokens to another secure address using Option 1 of the Provisional Transaction (if the Second Party's private-key is lost and the First Party is managing recovery) or Option 2 of the Provisional Transaction (if the First Party's private key is lost and the Second Party is managing recovery).

Also, whenever there is any security breach, the adversary needs one of the partially signed provisional transactions along with one of the private-keys to steal the funds. The adversary would then add the missing signatures using the stolen private key and broadcast the provisional transaction, hoping that neither party takes action before he can transfer the tokens to an address he controls. However, once the provisional transaction is broadcasted, cryptocurrency network monitoring systems—whether internal or external—will alert the First Party and Second Party to initiate recovery measures, as outlined below.

For instance, when the First Party's private-key is stolen and is used to sign and broadcast the Provisional Transaction to unlock and spend from the Smart Vault, its tokens remain safe and can be recovered as below:

1. The First Party creates a Recovery Transaction transferring the respective tokens to a new secure address using Option 3 of the Provisional Transaction requiring signatures from both the First Party and the Second Party.
2. The First Party sends the unsigned Recovery Transaction to the Second Party
3. The Second Party signs the unsigned Recovery Transaction and sends the partially signed Recovery Transaction back to the First Party.
4. As soon as the Provisional Transaction is confirmed, The First Party signs the partially signed Recovery Transaction with the Second Party's signature already

added to it and broadcasts the fully signed and valid Recovery Transaction to the cryptocurrency network before 1000 Blocks are created on the respective blockchain after the block confirming the Provisional Transaction.

5. Recovery is complete as soon as the Recovery Transaction is confirmed.

Similarly, when the Second Party is completely compromised and its private-keys are stolen, the First Party and other parties with similar arrangements and relationships with the Second Party can recover their tokens all by themselves using Option 1 or in coordination with the Second Party using Option 3 of the Provisional Transaction before 5000 blocks are created on the respective blockchain after the block confirming the respective Provisional Transaction as long as their private-keys are safe and secure.

The **Confusion Matrix** in **Figure 4** (Alice as First Party and BitSec as Second Party) enumerates the options available and outcomes of situations where private-keys of First Party and/or Second Party are compromised or stolen. It also enumerates situations where the First Party's private-key is lost but not when the Second Party's private key is lost.

Figure 4

Confusion Matrix

		BitSec's Private Key	
		Secure	Breached
Alice's Private Key	Secure	- Option 3	0-5000 Option 3
	Breached	0-1000 Option 3	#
	Lost	>5000 *Option 2	x

***** Can be recovered if private-key is lost but not stolen

Can be recovered if Alice and BitSec react immediately

x Slim chances of recovery

Since the Second Party is most likely an organized entity that can employ data-safety measures such as multi-vault secure backups, etc. this method does not explicitly specify the process and enumerate options available when the Second Party's private-keys are lost for simplicity and brevity. This method can be analogously extended to this scenario and more or simplified if desired by reordering/adding/removing options in the Provisional Transaction accordingly when planning and accounting for certain contingencies are deemed necessary or unnecessary. Also, the timelocks mentioned in the Provisional Transaction are one of the many possible combinations exemplifying a particular order and can be adjusted as required to suit any given arrangement.

It should be noted here that in certain situations where the First Party has lost its private-key, the Second Party can steal the First Party's tokens using just its private-keys but will not do so because such unilateral actions will result in loss of trust/business from other parties who will immediately withdraw their tokens from the Smart Vaults created with Second Party using just their private-keys as well as legal proceedings by the First Party. There is no scope for plausible deniability too as failure to initiate recovery and corrective measures confirms maleficence. Hence, the incentive and motive for the Secind Party to cheat the First Party of its tokens is non-existent.