Startup FAQ / Q&A

Last updated: October 16, 2025

General Questions

Q: What problem does your startup solve?

A: Great Wall, the protocol, app and startup solve the terrifyingly rising problem of **wrench attakcs** on custody of crypto assets.

Q: How is your approach different from competitors?

A: Great Wall is, the first to combine 4 crucial information security properties, namely:

1. **Devicelessness / Stateless / Kowledge-Based Authentication**: AKA **KBA**. This refers to the attainment of secret at hand, in our case, the private keys of a wallet, is not depending on any physical object except user's own brain, in contrast to PBA (**P**ossession-**B**ased **A**uthentication). Namely, by having access to a piece of information (the 'knowledge', in knowledge-based authentication) that is possible to memorize, user can attain their wallet in any device. There is no particular external device, metal plaque, object, physical address that, user depends on, therefore becoming an additional point of failure;

2. **Individual, non-Shared Custody**: This is the foundational premise of Bitcoin. Compromising on that premise negates the core valeu proposal of Bitcoin / crypto. As our saying goes "be your own bank" means you don't rely on anybody else or any company for the custody of your assets.

3. **Non-Obscurity**: This is a direct consequence of one of the most widely accepted paradigms in cryptology and protocol design, the Kerckhoffs's principle. The principle states: 'Knowledge inevitably diffuses, so a security system cannot rely on adversaries' ignorance on how it works, but on the secrecy of each instance.'. Put simply: 'A good security system is not one based on a secret trick (that one just *hopes* adversaries never get to learn), but on the secrecy of each user's keys'.

4. **Coercion Resistance**: It's the main point: to render attempts of using violence or threats thereof on custodians to compel them to give up their stashes (or any private information whatsoever). Not only that, system has to be such that an attacker cannot even threaten the integrity of the user's stash to try to obtain something else.

Q: Why are this specific combination of 4 properties so important?

A: The importance of that specific combination of properties becomes clear once we analyse the shortcomings of systems lacking or compromising each one individually, as described below:

1. **Individual, Non-Obscure, Coercion Resistant, but not KBA**: This setup describes solutions that are intensive in physical elements, such as sharding your key and distributing shards accross multiple addresses as well as the elements for physical security of each such address, like physical vaults, surveilance, home defense, alarm, geographic separation, etc. The shortcoming of that approach is the **astronomically high costs** of setup, maintenance and update. Put simply: 'Bitcoin is reduced to a glorified gold'.

2. **KBA, Non-Obscure, Coercion Resistant, but not Individual**: This setup describes solutions in which owner depends on services of shared custody. As explained above, that completely negates the foundational premise of Bitcoin. Companies offering said services, are themselves liable to state regulations, judicial subpoenas, and are themselves, additional points of failure. Put simply: 'Bitcoin is reduced to a glorified fiat currency';

3. **KBA, Individual, Coercion Resistant, but based on Obscurity**: This setup consist of relying on a critical secret trick that only works in so far adversary is ignorant about it. Many modern 'solutions' unknowingly have that shortcoming, which fundamentally violates the most widely accepted paradim in protocol design: Kerckhoffs' principle — more about that in questions about obscurity. Put simply, it is wishing on your lucky start that the bad guys willing to plan and execute an elaborate and risky endeavor as a kidnapping will just never hear about how your James Bond play works. Don't bet yours Sats (or your life) on that!

4. **KBA, Individual, Non-Obscure, but Coercion Vulnerable**: This is, basically, vanilla self-custody. Having your keys, but not doing anything in particular to defend them against wrench attacks. "We are still early", the saying goes. For purposes of going by as a needle in a haystack to avoid being attacked, not so much anymore, and, as crypto continues to grow in adoption and value, it will only get worse! So don't wait until problem reaches you (it's an *when*, not an *if*) to solve it!

Combining the aforementioned 4 properties literally means that "1) it's all in your head; 2) in nobody else's; 3) attackers are aware of that; and 4) they are unable to coerce you into giving up said knowledge." By logic, this can only be possible if said knowledge is tacit. Hence the jargon **T**acit **K**nowledge-**B**ased **A**uthentication **TKBA** to the innovation.

Q: What exactly does client pay for?

A: Client does **not** pay for ~~custody~~ in any definition. In one sentence: if clients does setup today and company disapears tomorrow, clients still has as much access to their

stash as right after setup. The catch is that running the entire process offline, though possible, is typically very inconvenient. For a small price, user can render their UX much more **convenient**.

Technical Questions

Q: How does the solution work?

A: Likewise BIP39, it's just a key derivation scheme that can run 100% offline in any device. The simple way to describe the Great Wall protocol is to say:

1. You input an initial seed equivalent in strength to 6 BIP39 words into an hours-(, days-, or even weeks-) long hash;
2. Hash specifies a seed to your 'private game', which admits a near-infinity of possible 'gameplays';
3. Bits of your 'private gameplay' of your 'private game' are added to entropy of master secret;

The catch is that the knowledge of 'gameplay' is easy to memorize but impossible to put into words. Because of that, you, alive and well, are strictly throughout the entire process, which can be set up to take hours, days, or even weeks. As a result, stealing your stash becomes at least as difficult as conducting a kidnapping that takes as long.

Q: What does such *TKBA game* consist of?

A: **Zooming in a fractal into a specific object**. Desired characteristics met by that are:

1. **Deterministic:** a fixed math formula with a given input will always yield the same fractal no matter the platform;
2. **Quantifiable:** a simple straightforward calculation ensures the given objects have exact the cryptographic strength needed;
3. **Familiarity and Ease:** everybody is familiar to the UX of zooming in Google from planetary scale into specific addresses. With 200 M distinct locations in Google maps, selecting one single location is roughly equivalent to 27 bits, which is north of 2 BIP29 words (11 bits each). That puts memorizing the equivalent strength to a conventional 12-words seedphrase at equivalent to locating 6 addresses: **easy!**;
4. **Tacitness:** Unlike the UX of zooming in to location on planet earth, zooming in into a location in a private fractal, involves using reference points that are all:
   (a) private — hence not previously known by the adversary;

(b) unlabeled — the only way to user can (attempt to) refer to them is by (attempting to) describing them;

(c) unfamiliar — yielded purely by math formulas with secret parameters, private reference points' shapes are unlike anything adversaries had seen before

(d) similar to one another — reference points differ from one another with very subtle nuances of position, shape, angle and coloring, which makes them **impossible to describe apart with words**;

5. **Culturally Neutral:** Also worh mentioning process is not linguistic, or linked to something that might be vary in cultural acceptability accross time and geography. It's purely mathematical, so no cultural barriers to adoption;

Q: How can user decide the duration of their time barrier? Wouldn't an adversary equiped with a setup 1000-X more powerful than your be able to perform your derivation in 1000-th of the time?

A: Three important concepts are the basis for it:

1. **sequences of tasks that cannot be computed in parallel**: Many types of sequential computation have the characteristic adding more machines, don't speed up the process. It's similar to how having 1000 marathonists don't allow you to run a marathon in 1/1000 of the time, only in the speed of the fastest runner.

2. **frequency scaling barrier**: continuing the analogy between compuation performance of a sequential task and running a race, the 'speed of a runner' is equivalent to the *frequency* of a CPU. For highly technical reasons, since mid 2000's we have reached a ceiling on frequency scaling Said ceiling is very directly consequence of physice — the speed of light, limiting how fast a signal can travel, or size of atoms limiting how small a transistor can be, hence,cannot be expected to change.

3. **memory intensivity**: continuing the analogy even further, there are known techniques to minimize the material advantage a top performing CPU has over an ordinary one, or in the analogy pro runner has over a casual walker. To have the sequential task at hand be intensive in memory access, would be equivalent to equip both pro runner and casual walker alike with a running parachute: the top performer would still outperform the casual one, but way less efficiently.

The combination of those three paradigms are widely used in modern technology. They are at the core of design of **offline** key derivation functions like that employed in BIP39, memory-intensive hashes, and time-lock puzzles.

Q: Wouldn't it be simpler to have the time barrier consist of blockhain block counting?

A: In that case, there would be a necessary piece of information published on the blockchain, therefore invalidating the premise of individual non-shared custody. The entire process has to be possible to do offline.

Q: There are current solutions also based on imposing a time-barrier. Why are they inadequate?

A: The catch is 'What exactly happens / becomes possible to be done after the time barrier?'. If the answer is: 'transaction from a protected address can be redirected to emergency address before n blocks enlapse', this means the system only works in so far adversary does not know how it works, namely it's an obscure system. The problem with obscurity is that not only, knowledge diffusion undermines a security system, but also it makes it backfire. In that example, an educated attacker has incentive to assassinate wrench attack victim to prevent them from performing transaction redirection. Reliance on obscurity, therefore, introduced the incentive for an educated attacker to worsen their attack — and obviously, there are no reliable ways to prevent wrench attackers form educating themelves (Kerckhoffs' principle).

Q: How can user buy improvement in convenience without compromising security properties or true self-custody? If I pay for somebody else to derive my keys, then I no longer have individual custody of them, right?

A: User doesn't pay to derive the seed to the private game. Instead, he pays to derive a one-time usage key to decrypt locally stored game seed. The outsourced material is insufficient to ... As a consequence anonymity of user becomes critically important, which consolidates first-mover lock-in (anonymity is maximized by joining marketplace with largest pool of users).

Q: Wouldn't a frustrated / distrusting / uneducated wrench-attacker, then brutalize their victims? How to solve that?

A: Likewise private security companies warning signs, users will want to acquire, use and wear branded merchandise to dissuade attackers or would-be attackers. Likewise warning signs, branded merch will educate attackers (and would-be attackers) that victim (and would-be victims) really are well protected.

Q: It seems like there is an obvious single point of failure in user forgetting either inputs to their setup. How to fix it?

A:

Q: Is the expected UX, by itself, not too tiresome?

A: We are talking about a half minute game-like UX, automatically notified and replayed

on average once a week. Different, but definitely not much to ask considering the benefits.

Q: What is so bad about obscurity, really? How to avoid it? Can you provide examples?

A:

Q: Can the system be used for other currencies?

A: Protocol is merely a key derivation scheme. Resulting secret can be converted into any secret user wants to protect against wrench attack. That includes key to any wallet, any digital secret, or singature scheme, password manager, or even physical vaults.

Business Model

Q: How this UX improvement purchase happen?

A:

Q: How will clients feel about that?

A:

Q: What exactly is the business model?

A: Company is a marketplace between anonymous clients of key derivation and providers of key derivation. Clients want to stay anonymous in order not to reveal their window of vulnerability (namely, the span of time in which their time-locked virtual vault is open or close to be opened). Another venue is to sell branded merchandise. Likewise warning signs of private security companies, branded merchandise will educated attackers and would-be attackers that their victims (and would-be victims) really are well protected, and that attempts on wrench attacking them are futile.

Q: Is there any first-mover advantage or lock-in mechanism?

A: Absolutely! Users want to maximize their anonymity. The more anonymous they are, on buying their key derivation, the less likely it is they are wrench-attacked right upon receiving their keys (which would anull their time barrier). By adhering to first mover, new users make sure they blend in the largest possible pool of other users, therefore maximizing their anonymity. This is similar to having the tendency to choose the largest social network.

Q: What are the customer acquisition strategies?

A:

Q: Are payments recurrent?

A:

Q: Are there ways to minimize user churn?

A:

Q: Is there synergy among different revenue streams?

A: