00:00

Minor.

00:02

So Jacinta and I worked on a little agenda to make sure that we.

00:06

Were able to get all of those questions.

00:16

Sounds interesting.

00:26

And then Andre has some items that.

00:28

He would also like to talk about with regards to release planning. Andre, how about we start with your.

00:47

Topic around release planning?

00:51

And basically, like.

00:56

I've got the hackathon brief up here, which I can share. And there's three deliverables that we're hoping to pull out of this.

01:06

So one is a signer demo.

01:09

So as part of today, I would.

01:11

Like to almost visualize or storyboard our.

01:15

Way through what this demo would look like. Okay. What are the user journeys that we're hoping to be able to show off? What's the experience going to look like? Bryce kind of went through a similar thing this morning with regard to clarity VM.

01:33

I think.

01:36

Being able to work backwards and only build exactly what you need.

01:42

To build to this meeting is being recorded.

01:47

To be able to pull off the.

01:49

Performance is kind of the goal of that exercise. So anyway, that's deliverable one, deliverable two and three are pretty intertwined. And so you could assume that Andre.

02:02

Is leading two, I'm leading three.

02:05

Is downstream of two. So starting with Andre thinking about okay, the go to market plan, the release.

02:13

Planning, essentially like making sure that there.

02:18

Is just total clarity on what the definition of done is for SBT 0.1 and 1.0. There's some ideas that Andre is going to ask about today to try to simplify and or get us there a little faster.

02:39

And it has implications to has legal.

02:44

Implications, it has business partnership implications, product implications. So anyway, once those things are settled, then it'll be reasonable for me to.

02:56

Start to dial into the actual Sprint.

03:01

Planning and the project planning that will go into helping us all achieve those.

03:06

Goals reflective of the product decision. So, anyway, enough of my rambling. Andre, take it away.

03:15

All right, thanks, Will no and appreciate all the context there. So, yeah, happy to go over this and try to talk about this at a high level, but kind of explain sort of how we're thinking about some of the design trade offs for the release planning. And so a key question that I'm thinking about with some others is basically, how do we derisk the release timeline while optimize the release of SPTC? SPTC mini optimizing for speed while also ensuring that user funds are safe and secure. And we're going about the rollout in the appropriate way and really just making sure that we are structuring this go to market plan in a way that's aligned with our goals for the ecosystem. We're capturing these product requirements and we're weighing the trade offs appropriately. So I have a Notes document linked here that you all can feel free to dig into more that has some more of the requirements captured from some of the apps in the ecosystem that I've been speaking to and try to kind of distill all of this into a concise summary.

04:34

And so basically, just to recap a few different options of where things are today, I guess first I'll start with the problem, and that is that Mini, as it's currently designed, is really a developer focused release that's focused on testing and user feedback for the SBDC team. And it really hasn't been designed for real capital under adversarial scenario. You know, I kind of mentioned that here around what are the security assumptions that we're making with Mini. And my understanding is that we've designed Mini with an open signer set without any economic incentives against bad behavior. And so the actual go to market for that probably, but we're planning on launching it on a main net, but strictly for the purpose of testing. And so it would probably be suited for one bitcoin or some minimal amount of capital to test in real scenarios, not like a full sort of release that is going to onboard a large amount of users.

05:59

But I think we're trying to balance that with the market pressure and the ecosystem demands and the requirements from apps basically to ship something sooner. Because if we aren't able to have a product in the next few months that is in a usable state, basically the risk is that teams will be forced to switch to options like xBTC, which could fragment liquidity and maybe search out alternatives. So basically trying to find the right trade offs between optimizing the timeline for something that's usable with some of these security guarantees. And this is why the security assumptions are really important. Because if there are ways to improve the security assumptions that we're making with this current model and make it usable so that it could potentially support millions of dollars of liquidity, it could be something that is really worth it while for us to consider because it would solve a few problems for the ecosystem.

07:11

And so there's a few different ideas that have kind of been kicked around, some of them have different trade offs. But I think there's one option which is that we can continue with this plan. We can launch SPDC mini first on a testnet and then a limited release on a main net with a small amount of capital and be very clear that this is only for testing, it's not going to be for user onboarding. This likely means that SBTC, like the actual product launch, is blocked on the Nakamoto release, which I think.

07:55

We can.

07:55

Talk about the timelines for that and how that impacts the broader timelines. I think there's another scenario that I know a few folks have been considering around. Could it make sense to do more of like a closed signer set? Is that something that could potentially allow us to have more security in the system and launch a product and then basically gradually decentralize it over time. I guess the third option would be, is there a way that we can improve the security assumptions of Mini that we currently have without having to go to a closed signer set? So, basically, these are the trade offs that we're trying to make. Just wanted to share with you all kind of some of the feedback that we have from the ecosystem and why we're trying to sort of launch something this year and hopefully start a conversation for what is the right way that we can stage out our release process between Mini V two nakamoto each of these have there's a testnet launch.

09:02

There's time for testing and bug fixes before it can go onto mainnet. There's likely a period on mainnet where it needs to have further testing before we can increase liquidity and trying to kind of condense all of these things. And I know I've asked Will to kind of help to visualize these trade offs a little bit better, but I think maybe I'll pause there again, try to just give an overview of kind of some of the trade. Offs that we're facing and basically try to plan for the next six months for what does this release look like? And how can we potentially build something that really meets the requirements that we're hearing from the ecosystem?

09:44

App Builders.

09:46

Thanks. Andre.

09:49

I'm curious if others are willing to listen a little bit more to you. I think it'd be super helpful for me to hear just like, what's the leader for you and why? Just so I get an even better sense of how you're kind of weighing the trade offs of these different concerns. Because there's, like, pros and cons in each one, obviously. So just from the product kind of mind that you have, I'm curious which is your top?

10:12

Yeah, so there isn't a clear answer, but I think that in my perspective, waiting until Nakamoto, which is likely sometime in Q One at best, we don't know for sure, represents a significant risk in my perspective, because what that would mean is that basically we lose some momentum in the ecosystem. Those builders have to go elsewhere. There's like, market pressure of more competitors launching, and I just wouldn't want the release of SBTC to be blocked on these major architectural improvements that are coming with Nakamoto. I think the full system is the end goal, right. That includes greater security, backed by the full economic guarantees of bitcoin faster blocks to provide a better user experience on DeFi. But I think that there is value in releasing something sooner so that we can get, again, like this MVP out into market, that ideally we can start actually onboarding users and build momentum leading into the Nakamoto release, as opposed to trying to start from more of a cold start.

11:31

Again, all of that is kind of dependent on, can we find something that has robust security guarantees that we can get to a place that it could support millions of dollars of liquidity? And can we do it in a way that just satisfies all of the business and legal requirements that we have outlined?

11:53

I just wanted to go to a.

11:56

Quick little set of notes that Andre and I have put together as just like a gut check when we're making these decisions.

12:02

So we just wanted to remind ourselves.

12:05

What is the value proposition of SPPC? And the statement that we came up with was it's a programmable bitcoin asset with Fast settlement, a positive developer experience, and with minimal counterparty risk?

12:18

So if you unpack that programmable, that.

12:22

Means song clarity, Fast Settlement. We know that this is underway, like, in parallel with the Nakamoto improvements and the Clarity VM improvements, positive developer experience. We know that is being forged ahead with the Docs and the SDK.

12:39

And a lot of programming.

12:41

And so really, the main crux of this all comes down to the counterparty risk. If counterparty risk is determined sorry, if the signer is what determines counterparty risk, and counterparty risk is determined by the amount of decentralization.

13:00

And decentralization is a spectrum, essentially like.

13:06

Trying to right fit.

13:10

The amount of.

13:11

Decentralization that's going to occur. How do we set this up to be able to plant our flag in the market and get people onboarded and testing, but then dial up the decentralization as we go? Because that seems to be the most complicated bit of.

13:42

That'S. A.

13:43

Thanks, Will. It's a nice little prelude for some stuff that we'll be talking about in office next week around some of the value proposition for SPDC and how all the work that we're doing and all the tasks that we have kind of aligned to that.

14:00

So maybe let's pause for a second. We can come back to this conversation. It might make even more sense or provide clarity after we talk about some of the other signer related things.

14:14

Jacinta, I wanted to first kind of put you on the spot, and let's say that on August 21, the last.

14:26

Day of the sprint, on Monday, we're doing a demo of the sign in basically the fruits of our labor in New York. Could we just hypothetically walk through what's a storyboard for that user flow? What does this look like? What are we demoing to people? And I think maybe just going through that mental exercise and hearing it from.

14:53

You will help us get a sense of what needs to be accomplished.

14:59

Sure.

14:59

So there are kind of five interacting components.

15:04

I guess we would need to have a Bitcoin node set up. We would need to set up a Sax node. We would need to ensure that Sax node, like the Stacker DB implementation is done, because that's how signers communicate with each other. We would then have a signer binary that we would run that would have a configuration file with it. And preferably we would have more than one signer running on multiple VMs so that we can show that they're not just communicating locally and are communicating via sackerdb. And then what we would need to do is open up a bridge Web UI or a CLI tool, put in a deposit or a withdrawal, and observe as signers, pick this up and process them. And if we've submitted one with an address that we consider a bad address that we configure signers to ignore, we should see it fail to go through.

15:59

If we do one that is from a valid address or a valid amount, everything good from the Web UI, SBTC Web UI, like we do a valid deposit, we should see that go through and should see SBTC appear in our wallet. Similarly, if we did a withdrawal, that's kind of how the overview of the system would look.

16:21

But in terms of like, if you.

16:24

Wanted to go into depth for the signer, you might spend a little bit more time looking over the configuration. Or if you wanted to demonstrate the SDK, you might quickly write up signer binary using it that has different validation logic for particular transactions. Those are all things you could do in a demo if we actually get it all done hypothetically during the sackathon.

16:44

Which is quite a high expectation. Sure. In terms of.

16:57

How you envision these three days going. Again, just back to visualization. You come in, set your bag down.

17:10

Tuesday morning, what's the first thing that.

17:13

You want to kind of grab people and huddle about and get working on now that we're all in a room.

17:22

Together and get the most out of.

17:25

That, where do you start this process?

17:28

I would start this process on the overview flow of the system, so the SBC docs that are getting done, as soon as they're done in advance, I would want to actually sit down and look at the overall diagram of the entire SBTC system so that people are all on the same page. Because I have had many conversations with.

17:44

People and realized their view of the.

17:46

System is not the same as mine. So I would start there, and from there, since it's signer focused, I would go deep into the signer architecture overview and have the assumptions listed for I mean, they're already going to be hopefully listed in the SBC doc. But the assumptions for the Clarity Work stream make sure that we're on the same page and that the functionality that the Clarity Working Group requires to support this design are very clear. So the Clarity guys can do what they need to do.

18:14

Same goes for the Stacker DB protocol level communications that they are also clear on.

18:21

What I'm pretty sure that's kind of independent. It just takes a chunk of binary, like of bytes, and we need to handle it. But that's kind of what I would do, is here's the architectural component, this is kind of the assumptions we've made. And I assume in that first conversation there will be many questions of how is that going to work? Because this scenario, that scenario, all these edge cases, hopefully in advance, we have identified those, but I'm hoping also at the same time having all these people in the same room, they might see.

18:47

Something we didn't see.

18:49

So that would be the first step. And from there, assuming we've come to a conclusion of yes, that's the correct architecture and we have the edge cases.

18:56

Mapped out, it should be at that.

19:00

Point a little bit easier to say, hey, you work on this component. Hey, you work on that component. And we would go from there because I'm sure there'd be then integration steps depending on how far along we get. That's kind of, I think the start to the hackathon. I'm not sure if anyone else disagrees.

19:16

But that's what I would do. That sounds great.

19:22

So, yeah, we will be starting off I don't know if everyone caught this.

19:25

But canceled a bunch of meetings for.

19:28

Next week, but added a two hour block of time so that we can have hybrid remote IRL sync.

19:37

Up sessions where people that aren't in.

19:41

New York are able to dial in, listen in, contribute, take something away, raise their hand on what they can work on, and the people in New York can learn from the insights from people that are dialing. Yeah, I mean this sounds like a really great way to get the ball rolling. Martin or anyone else? Joey, any insights or things that you.

20:12

Want to add to that? I don't have much to add.

20:20

I think it's very important that we just figure out how to deliver as much as possible in as short time as possible without inflicting damage to the quality of the code or shooting ourselves in the foot for any future efforts because there's a lot of work remaining even after the hackathon.

20:42

Yeah.

20:43

So Martin, on that note, do you want to say anything about expectations?

20:48

This came up in a call previous to this. Okay, let's say we're making good headway.

20:56

Things are happening fast. To what extent do we want this captured in real time in the documentation before getting implemented or in parallel to it being implemented?

21:10

Yeah, I mean, we can definitely implement things first, especially considering details, but if we're starting to make decisions that have severe implications for the rest of the system, then that information should naturally be relayed or communicated. If we need a lot of big changes in the clarity contracts, we should be like, of course we need to strike a balance between moving fast and also not going off on our own tangent because even though that feels productive, that's not going to help the project in the broader sense. And at the end of the day, we need to look after the big picture. We need to make things that are helping us move forward and deliver SBDC. We should not just entertain our own sense of productivity because that can also be deceiving.

22:04

No heroes. We don't want any heroes.

22:08

That was a very vague answer, but I think hopefully it shouldn't be. I think those things are helping each other. Like good documentation helps you execute, good libraries help you execute. And when you're executing, you should backport anything that has implications for the documentation and for libraries that you're relying on. Of course, not necessarily third party libraries, of course, but our internal libraries that we're maintaining that enables us to move faster.

22:35

Cool.

22:37

Jose sedzus anyone? From the clarity side of things. Thoughts on how to best integrate into.

22:53

The work happening with the signer during this little sprint next week?

23:00

Yeah, I think best case is I make it out there and just walk through Claire, walk through everything that we have in the developer release with the signer group. Outside of that, my focus from now until the hackathon is going to be focusing on integration and specifically maybe writing up like a one sheet or something that's like, these are the most critical functions for the signers and just have those summarized up so Jacinta and team can have that ready.

23:44

Joey, how about you? Any top of mind goals for next week that people should be aware of? I think it's pretty much already been.

23:57

Set at this point.

24:03

Stepan on the SDK side, anything you want to add?

24:10

Yeah, so as a prerequisite to the hackathon, I'm today working on the Commit Reveal transactions, and I think that would be in before the hackathon begins. That was kind of one of the important things to get in before it, so, yeah, I'll do my best in order to kind of support the hackathon as much as possible. We also have some types from the Blockstack Clip Library that we also need to port, but that's not a lot of work. So yeah, I think we are getting there. We are on time.

24:42

Quick question, do you have a CLI as well, or is it the library first? Sorry, is there CLI wrappers included in this right now or is it library first?

24:54

No, it's just a library. But the CLI, when everything else sits in the CLI, it's just like a.

25:00

Wrapper on top of yeah, I was just wondering if I was just super eager, getting a bit greedy.

25:07

Yeah, I saw the PR, but I.

25:11

Haven'T looked at it yet.

25:13

Yeah, no worries. When you've had that.

25:19

Cool. And then Jacinta coming back to the.

25:24

Goal, deploying a Block Producer signer, stacker signer, SBTC signer.

25:31

These were some notes I believe I.

25:33

Had started, you looked at yesterday. Anything here that you want to make sure that we touch on with the group?

25:40

A good thing? Actually, I totally forgot to mention this as part of the setting up a demo. The only other thing I didn't mention is in addition to the bitcoin node stacks, node stacker, DB, all that stuff. We are going to need a Revealer server, basically. So this is because of the commit Reveal solution. There needs to be some way for someone who's using that process to let the signers know, hey, I want to.

26:11

Peg into the system, deposit into the system.

26:14

But we don't want to have these signers expose themselves like with a public API endpoint. So instead we should have a separate binary that Bridge or anyone can interact with and say hey, here's my commit with my taproot spend script or unlock script. However, it doesn't have to be one particular type of lock script. But that needs to be able then to be read by all signers and they can then determine if they want to actually reveal that to the system. So that is a component that is a very separate and thing that someone could write during the hackathon. That'd be a very good task, actually to do quite independently, but that would be needed as well. I forgot that.

27:00

Is that something anyone feels bold enough to raise their hand to take on now?

27:07

Or do we need to wait and kind of have a more complete understanding of the requirements?

27:18

I think the most important thing is knowing the Reveal format, which I think.

27:23

Is well documented by Morten in the.

27:26

SPTC doc, so it could be started now. It's certainly not something that needs to.

27:32

Wait on the hackathon, but I think.

27:37

All of these efforts should be coordinated relating to the bigger picture. We can definitely delegate this piece. I think this is one of the simpler piece that we implement. There are, of course, a lot of different ways we can do this, but I'm not sure we need to assign this now unless that's something I mean, it's still the interface between Revealer and the rest of the design. It's one of all the moving pieces of designer. It's really important to track this on the board, but shouldn't stare ourselves blind and just jump on the first thing that we're all looking at. Right.

28:12

I think the biggest kind of point.

28:15

Is the Schema definition of what the get and post to it would be would need to be known, basically.

28:24

But in terms of the implementation, beyond that it's pretty independent. Yeah, but I guess from within the.

28:36

Signer we're going to have a very simple interface. Like we have the trait that you already defined and all of this interaction is going to be sort of isolated and quite loosely coupled with the rest of the signer. So this is work that can be done pretty independently, the whole interaction with that.

29:00

But sorry, Will, you did have the question you originally had below about what was your question? Sorry.

29:12

We had pulled these notes together. I just wanted to see if there was anything I just wanted to remind you that these are here, see if there's anything that you needed to talk through today.

29:28

If anyone has questions specifically about these points, that would be good. One of the concerns that was raised was about the handoff process and the writing to the Smart contract. So technically, any one of the signers can trigger a DKG round to determine the new SBTC wallet address. But the actual setting of this wallet address in a Smart contract, like in Alpha, you had to be the coordinator in order to have the authority. But technically, as far as I'm aware.

30:00

Any one of the signers could do.

30:01

It at any point. So they could overwrite each other, they could overwrite a bad address. For example, you should, as the new signers who are about to execute this wallet handoff, be able to verify, because you should be able to observe the results of the signing round.

30:14

So you could say, hey, that's an invalid wallet address.

30:19

We're not actually going to send it there. But that's only the signers, really, who have easy access to that.

30:23

That address set in the Smart contract.

30:25

Would be used by individuals who want to peg in to say, that's where I need to send my funds and we need to have some better, tighter control of this.

30:35

And since we're doing coordinators are determined.

30:38

Via VRF, what sort of validation do we have for the actual Smart Contract call and setting up this address? That could have some implications. Granted, we are going under the assumption for many that people behave correctly. So is it enough to just say we're not going to worry about that? Or is that something we should really bake into smart contract? I'm not sure that's kind of what that handoff discussion is about. There's definitely a way for signers to validate what's in the contract, but I just don't know in terms of validating before it even gets into the contract, how do we prevent an invalid address being set in the contract? What else? We do want to use a VRF sort of to determine the coordinator of a particular transaction of the DKG signing round. How we implement that's up to debate, but that's the approach we're taking at the moment.

31:38

I think that's it more ten. Stephan and I had a discussion about how we might maintain internal state to prevent double withdrawals. So that's good, but I don't think there's anything else to go over at this exact moment. But if people have concerns about the system, like raise it, we might have answer for we've considered that, but if you don't know that we've considered that, it means we haven't documented it well and we need to make sure that's put somewhere so that people are all on the same page. But I don't have anything else to add.

32:16

Great. Andre, I know that you're still maybe feeling blocked on this. Is there anything that we can do together here on the call to help move you forward?

32:32

I don't think so right now. I mean, I would love to hear other folks opinions on this, but also you might want to think about it. I have some of the trade offs outlined in the doc there. My goal is to have a strategy finalized by the end of this sprint. And so I'm sure the hackathon is going to be really important to kind of get on the same page about our release strategy. And so yeah, feel free to think about it. Let me know if you have any thoughts on which approach we should consider. And yeah, again, I think that whatever we decide is good as long as we're all agreed on the intent for that, who it's designed for and we can communicate it out to other folks.

33:25

But yeah, one quick question.

33:28

I have actually kind of related to this release if it was done on mainnet, given conversations that I've had with.

33:36

Other Trust Machines employees.

33:42

Is there a way for us for.

33:44

Mini to guarantee a certain number of.

33:49

Signers participating in the system? I know we mentioned there's whitelisting, but is there a means for us to say SBTC is just not going to operate unless we have this distribution of vote shares? Is that something that we would consider for the Mini release?

34:07

There's currently a limit on how much needs to be stacked in order there to be like, an active state, but I'm not sure if there's a requirement for the distribution of how many make up that million jacinta.

34:24

Yeah, so I know there isn't one, but is it something we would consider potentially adding to prevent a single signer dictating the system, basically, like to prevent.

34:36

That ever even being an option.

34:41

I think for the first developer release, I kind of like that feature being in, but yeah, I could definitely know how to have that hoisted out by the second release.

34:53

That's the only comment I had.

35:10

Cool.

35:15

Nothing else comes to mind that needs to be discussed. At least I don't have anything identified. Happy to give people some time back.

35:30

Let me know if you want to keep the call open. All right.

35:39

Talk to you soon and.

35:42

Everyone see some of you soon. Thanks. Y'all take care. Thanks.

35:49

Bye.

35:54

The recording.