EPISODE 548

[INTRODUCTION]

[0:00:00.3] JM: Soumya Basu is a PhD student at Cornell, where he studies distributed systems problems associated with cryptocurrencies. Soumya is advised by Emin Gun Sirer, a Cornell professor who previously appeared on the show to discuss smart contract security.

Soumya joins the show today to talk about a variety of issues in the cryptocurrency space. We first explored the degree to which Bitcoin and Ethereum are actually decentralized. They are probably less decentralized than you think. This is because of the centralization of mining pools. Much of the transaction processing is centralized in Bitcoin and Ethereum.

After talking about decentralization, we got into Soumya's research focus; cryptocurrency networking and bloc propagation. Bitcoin transactions are collected into blocks. When a Bitcoin full node solves the cryptographic puzzle associated with the block of transaction, that full node broadcasts the new block to all of the other nodes in the network.

It's important for that blocked broadcast to be fast and efficient so that the other full nodes in the network can be made aware of the new block as soon as possible. Then they can start working from the updated chain. The problem of making all nodes in the network aware of a new block is known as block propagation. Block propagation can be accelerated through the use of relay nodes.

A relay node is a node that is dedicated to communicating these new blocks throughout the blockchain. Soumya is working on a relay node architecture called Falcon. In this episode, we talk about what Falcon is.

If you're looking for all 700 episodes of Software Engineering Daily, you can check out our apps on the iOS or Android app store. We've got tons of episodes on blockchains and distributed systems and business, tons of other topics. If you want to become a paid subscriber to Software Engineering Daily, you can hear all of our episodes without ads. You can subscribe at softwaredaily.com.

Also, all of the code for our apps is open source. We've got a burgeoning open source community. If you're looking for an open source community to be a part of, please come check us out. We would love to have you involved. Just go to github.com/softwareengineeringdaily, join our Slack, come hang out with us and hack on some code together.

Also, meetups for Software Engineering Daily are being planned. If you're interested in coming to our meetup in New York or Boston, or LA, you can sign up at softwareengineeringdaily.com/meetup. I'm sure we'll have more in the future, so even if you don't live in those places, please do sign up.

With that, let's get on with this episode.

[SPONSOR MESSAGE]

[0:02:49.6] JM: Users have come to expect real-time. They crave alerts that their payment is received. They crave little cars zooming around on the map. They crave locking their doors at home when they're not at home. There is no need to reinvent the wheel when it comes to making your app real-time.

PubNub makes it simple, enabling you to build immersive and interactive experiences on the web, on mobile phones, embedded into hardware and any other device connected to the internet. With powerful APIs and a robust global infrastructure, you can stream geo-location data. You can send chat messages, you can turn on sprinklers, or you can rock your baby's crib when they start crying. PubNub literally powers IoT cribs. 70 SDKs for web, mobile, IoT and more means that you can start streaming data in real-time without a ton of compatibility headaches. No need to build your own SDKs from scratch.

Lastly, PubNub includes a ton of other real-time features beyond real-time messaging, like presence for online or offline detection and access manager to thwart trolls and hackers. Go to pubnub.com/sedaily to get started. They offer a generous Sandbox to you that's free forever until your app takes off that is. Pubnub.com/sedaily, that's P-U-B-N-U-B.com/sedaily.

Thank you PubNub for being a sponsor of Software Engineering Daily.

[INTERVIEW]

[0:04:32.5] JM: Soumya Basu, you are a PhD student at Cornell. Welcome to Software Engineering Daily.

[0:04:37.2] SB: Thank you. I'm glad to be here.

[0:04:39.6] JM: You're doing your PhD at Cornell. What does your research focus on?

[0:04:45.0] SB: My research focuses on different aspects of cryptocurrencies. In particular, I look at things like the network layer and how to make the network layer more efficient so that permissionless cryptocurrencies can run even faster. I also look at things like the core consensus engines behind permissioned blockchains and trying to make those faster and more efficient, have better security and privacy guarantees as well.

Then I also have a little bit of work looking into the economics behind a transaction fees and things like that. Really it's dabbling in various aspects of cryptocurrencies.

[0:05:24.1] JM: That term permissions blockchains, is that what a lot of big corporations are looking at, like to mediate supply chain and stuff like that?

[0:05:33.2] SB: Yes. I think in the settings where you know every player that is important in the consensus engine, I think a permission blockchain makes sense. In a corporation, or in an industry you know who the players are for the most parts, so you can actually just construct a permission to blockchain and trust the consortium of individuals, instead of leaving it completely open.

[0:05:59.8] JM: For those areas with the permission blockchain and you have some trust around people, can you use algorithms that resemble the ones that we have been using in trusted environments for years? Like for example, what Amazon uses to have DynamoDB reach consensus with its different nodes, or – Yeah, I mean, just basically the kinds of algorithms that perform consensus across cloud-based environments. Can you use those things?

[0:06:29.9] SB: No. This is actually a very new consensus, which is actually been studied for 20 somewhat years at least. The main difference between Amazon running a highly available cloud service versus a consortium blockchain is this notion of trust. When Amazon is looking at failures, what it's really trying to prevent is some node failing in a weird way.

It's not really looking for nodes that are getting hacked, or can do arbitrarily malicious things on nodes that are actively trying to take down the system. Whereas, in blockchain you're really looking at nodes that are actively trying to attack the system and it's a much more like adversarial environment. You're the classical algorithms from the literature for something like DynamoDB as paxos and for the Byzantine case it's PBFT, a Practical Byzantine Fault Tolerance.

[0:07:28.3] JM: Right. What I was wondering is, because I totally understand that in the environments like Bitcoin, you are looking for a solution to Byzantine failures. That is failures that can be totally random, totally malicious, or they could be very non-random and malicious. Basically the worst-case scenario that very intelligent Byzantine general would invoke upon your perfect economic system.

It seems like if you have a consortium of banks for example, that are just trying to create a faster payment network and they already have a good amount of security around their nodes, it seems like you could have something that is more like paxos, where you have higher expectations for the kindness of the different actors that are going to be in play. I guess, as you said, you could have nodes still be hacked so maybe you still do want to treat them as Byzantine.

[0:08:21.0] SB: Yeah. There is really what I think Byzantine fault tolerance try to solve is this notion of trust. The question is if say Wells Fargo had the chance to do something malicious, to try and hurt its competitors, would it do that and will it always be nice from now and into the future?

It's hard to tell and it's hard to put trust in a particular organization. I'm just picking on the Wells Fargo as an example of a big bank. Really this works in any industry. You can look at things like you eluded to, which is security and these different organizations have different levels of security.

Some organizations, may be more prone to getting hacked than others. They may have different failure models, different levels of security expertise and so on. Even if you're not looking at the malicious case where everyone is trying to subvert the system for their own profit-seeking ends, it may also just be more prudent to remove this sort of layer of trust.

Really there's a lot of trust that we see in these places. One example I've used in the past is with Amazon. When you conduct a transaction on Amazon, you trust Amazon to do the right thing. To make sure that your package gets delivered and to make sure that the payment gets received on the other end.

Amazon charges for a premium for that trust. Wherever these trust premium arrives it becomes a use case for blockchain to eliminate that trust. It's really about trust more than fault tolerance in the traditional sense. That's how I look at the space and the promise behind the space.

[0:10:14.5] JM: I see. I'm really not even look – I'm not even thinking about the right access of tradeoffs. I would love to talk to you about consortium blockchains, but I did not prepare at all for that topic, so maybe we can save that for a future show.

Let's talk a little bit about Cornell, because Cornell seems to be pushing itself towards establishing itself as a university that is open to teaching about cryptocurrencies and taking them seriously, which is the exception rather than the rule. I think that's at this point, probably largely because there is just not a lot of academic brain power that has been spent on cryptocurrencies. There's not a lot of universities that have an expertise in cryptocurrencies.

Probably we'll see more and more universities develop an expertise. Cornell has really been at the leading edge. IC3 is a division of Cornell that's dedicated to cryptocurrencies. What are the goals of IC3?

[0:11:16.4] SB: It really is to pursue the science of blockchains. To figure out what it means – well what the use cases are, what are the problems that people are facing now and will face in the future. It really is about putting a very serious academic brain power behind blockchains and pushing the science of – looking at all aspects of blockchain, to really both permissioned and permissionless and looking at these different models and things that go on.

[0:11:46.4] JM: You work under Emin Gun Sirer. I think you're doing your PhD under him. What's it like working with him? For people who don't know, he's a well-established, maybe I've even use the word famous blockchain researcher, professor who has really put a lot of effort into researching cryptocurrencies.

[0:12:06.9] SB: Yeah. I really like working with Gun. He's an expert who's been in the blockchain space for a very long time. As in one of the earlier efforts of selfish mining was by him. He's obviously expanded into many other areas since then. I think the main benefit I get I think working with someone like Gun is that he sees what the key problems are in the blockchain space and he can anticipate a little bit well what the technical problems are coming up in the blockchain space and what things sort of work spending your time on.

Also, he knows the right people to talk to so that the work that you're doing has the impact and it reaches the right people who can actually make use of it and turn it into something that helps them, because at the end of the day actually – that's part of why I do research.

[0:13:03.8] JM: In some of the earlier shows of Software Engineering Daily, one of the questions that I explored a lot was what is the difference between academic research and industry research? Because around this time I was reading a lot of these papers, like the Amazon-Dynamo paper or Google MapReduce. You read these things and you're like, "Well, these seems like fundamental computer science research. If industry is putting out this kind of research, what do we need in academia for?"

Then as I have dived into cryptocurrencies a little bit, there are some academics that are saying things that probably are not necessarily incentivized for corporations to say, like talking to – my good friend Hasib interviewed Gun on this show and also there was an interview I did with Joseph Bonneau who is an academic in this space. Who is also like Gun in the same way, where he's very moderate and he's basically interested in knowledge and teaching other people.

It is that fundamental desire to just teach and to know more about the space, rather than to profit in the short-term, or to become some sort of ideal log. That notion seems very rooted in academia and it's given me a refreshed respect for the tenets of academia.

[0:14:24.9] SB: Yeah. I think what you hit on is one of the core differences, I think behind academic and industrial research. Industrial research tends to be a little more profit-seeking for their company. DynamoDB a great piece of work, but at the end of the day it was for service that Amazon is providing. It really is to help their internal services and pursue a profit-seeking goal.

Yeah, I think in academia you are encourage to explore this. I think the core difference of teaching is really – because a PhD student who's at some point is thinking about the next steps. I think that teaching aspect is really one of the key differences behind academia and industry. Because as an academic, you teach classes and there is this whole education aspect of the job that isn't there as much in industry.

Obviously, the branching out in terms of academic labs and industry labs, these are very broad brushes and you can find academic labs that operate a little more like industry labs and solve the problems that you would expect to see solutions and coming from industry, and also vice versa.

You've seen some great industry labs too contributing things that don't really serve a profit-seeking purpose for their respective corporation, but contribute to the fundamentals of computer science. I think this aspect of teaching is one key difference between industry labs and research labs.

[0:15:54.8] JM: My sense is that Cornell in particular is renowned for its distributed systems, research and its expertise among professors. Has there been a shift in the mindset of the distributed systems researchers around Cornell that – is everybody interested in cryptocurrencies, or are there people who are distributed systems experts and are still don't really care?

[0:16:21.1] SB: I think cryptocurrencies are interesting to a lot of distributed systems with people, just because it is very much like – when you're talking about Byzantine consensus and the early works done in that space, both consensus and yes in the Byzantine sense and in the normal fault tolerance I guess you could say with access, a lot of that work was pioneered here at Cornell.

Bringing that distributed systems expertise to blockchains is something that that definitely I see a lot of systems professors here at Cornell, even if they're not – not every project they're doing is in blockchain. A lot of systems professors here have a couple of projects going on in the cryptocurrency space and in the blockchain space. Just because it is a very good and motivating application to look at for the next innovations in this space.

[SPONSOR MESSAGE]

[0:17:23.6] JM: If you love Software Engineering Daily, I think you'll also love the Google Cloud Platform Podcast. It's a podcast about Google Cloud Products, how they're built and how you can use them. Really it's all about the changes that are going on in software engineering, as told from the point of view of Google engineers.

You can find the Google Cloud Platform podcast at gcppodcast.com. You'll hear from Googlers like Vint Cerf and all about Tensorflow and Firebase and BigQuery, from the high-level use cases to the low-level implementation. The GCP Podcast has all of that covered.

Find the Google Cloud Platform Podcast at gcppodcast.com and subscribe to it wherever you get your podcasts.

[INTERVIEW CONTINUED]

[0:18:14.7] JM: By the way, if you ever see Lorenzo Alvisi, or if anybody listening ever sees him, you can pass on my gratitude for him passing me in college when I was on the verge of failing distributed systems and may not have graduated. Thanks to Professor Alvisi.

[0:18:32.9] SB: I'll let him know. He's a good friend.

[0:18:34.8] JM: Okay. You know. That's good. He's a brilliant. Very difficult to take a class from, at least for me. Some people didn't have a hard time. I want to talk about some of your writing and you research and I want to get to Falcon eventually which is a Bitcoin relay network.

First to ease into that, because I think that's a little bit more of a complex topic, there's a blogpost that you wrote where you were auditing the decentralization of Bitcoin and Ethereum, because people take it for granted that these are decentralized blockchains with fairness in decentralization.

You were eating away at some of the things that we take for granted. The first idea that you explored in this blogpost is the idea that Bitcoin underutilizes its network. Bitcoin is a bunch of computers that are networked together. They're consenting on transactions. How could these computers be used more efficiently?

[0:19:37.6] SB: I guess one thing in terms of that decentralization work, it's really looking at different aspects of decentralization and trying to see how we can quantify them. I think, what you said is correct, but we're really looking at trying to quantify these different aspects of decentralization, arguments that people have made.

Now in particular for Bitcoin and utilizing its network, what we did was a comparative study where we said – where we looked at the bandwidth in 2016 and we looked at the bandwidth in 2017. The bandwidth requirements of the protocol have remained the same. Whereas, if we look at the actual nodes in the system, the bandwidth that they have allocated to provisioning for Bitcoin has actually increased. You can increase the block size.

Really what we are trying to advocate for here is let's take these measurements and decide our protocol parameters based on these measurements. If we see that the Bitcoin – that the bandwidth of the Bitcoin, of the average Bitcoin node has gone up by a factor of two, we should increase the protocol bandwidth requirements by a factor of two, then we can now make these evidence-based decisions on what the protocol parameters are based on who is using the protocol actually.

[0:21:00.5] JM: It's like Moore's and its corelet are continuing, computers are getting better, they're getting higher bandwidth. Why wouldn't we increase the block size for example to impose more strain on the network, because we have better hardware?

[0:21:18.8] SB: Yeah, exactly. It's that question. The thing is we've seen historically that the Bitcoin bandwidth has gotten better. There is no reason to believe that it always will get better. It maybe that at some point, we see the average Bitcoin nodes bandwidth going down, in which case at that point the argument for block size decrease become much more compelling.
This can happen even with computers getting better, because it depends on what computers are people using to run the Bitcoin nodes. Are you designing the Bitcoin protocol for your cellphone, or are you designing the Bitcoin protocol for well provisioned nodes that are run by major corporations? Which type of node is using your system, will affect the design decisions that you make much more deeply than I think people have been looking at.

[0:22:11.6] JM: Another point that you explore in that blogpost is that Ethereum is more decentralized than Bitcoin. But that's not saying much, because neither of them are actually very decentralized. Are Ethereum and Bitcoin mining nodes, are these mostly just in large corporate mining farms? Is this basically an AWS type of situation?

[0:22:36.1] SB: Yeah. That's what it seems like. It's either mining farms, or mining pools, where you may not have all the mining located in one central location with where the cheap power, it may just be that there are many people who are buying mining rigs and then just pointing them to a mining pool instead.

Yes, I think the right way to look at these miners is looking at them as large professional organizations, rather than your Mom and Pop store that just thought of Bitcoin miner and is mining by themselves like an excavation.

[0:23:19.4] JM: Well, maybe codec will democratize that and make it more accessible to the Mom and Pops.

[0:23:24.4] SB: I haven't been following that too closely, so I'm not – I see a lot of these – news articles popup on Twitter and everything. I'm really focused on one of these hind aspect of this.

[0:23:38.7] JM: I don't blame you.

[0:23:39.5] SB: Yeah. Well, like I read some of them because it's amazing, but I don't follow everything super closely.

[0:23:45.4] JM: I don't blame you. I think what codec announced was – they announced that they were going to issue a token and that they were going to rent out mining rigs to people. If you wanted to borrow a mining rig from codec, you can. Let's not talk about that. Let's just leave it to other podcast.

Ethereum is more decentralized though. Like you said, most of the mining power is in mining pools and mining farms, but Ethereum according to your studies is more decentralized. Why is Ethereum more decentrealized?

[0:24:18.1] SB: The thing is with decentralization, it's a tricky topic, because you can't put a number on it. You can't say Ethereum is five decentralized and Bitcoin and seven decentralized. Therefore, Bitcoin is better than Ethereum, or Ethereum is better than Bitcoin.

In some aspects, so if we're looking at full nodes that are running and validating the Ethereum and Bitcoin blockchain respectively, Ethereum is better distributed than Bitcoin. If you're looking at the physical decentralization of full nodes, Ethereum is more decentralized in Bitcoin. If you're looking at the mining decentralization, it actually looks like Ethereum is a little more centralized than Bitcoin.

Again, when we're looking at Ethereum versus Bitcoin especially with mining, we're comparing shades of gray here. It very much is they're both very centralized and Ethereum is slightly more than Bitcoin, but neither one are really doing all that well in that metric.

[0:25:22.7] JM: What are the dangers of the downsides of having these miners and these mining pools centralized?

[0:25:30.2] SB: The biggest danger I see is transaction sensorship. If you have a transaction that you need to get through and some miner doesn't like it, if miners have large amounts of hash power they can actually delay your transaction from getting through. This also opens up Bitcoin to different kinds of sensorship too.

If you can contact I think less than 20 miners and get over 90% of the hash power, well any government can do that, now can say, "All right, well I want to sensor this transaction and don't confirm it, or we'll send the authorities on these 20 people." Now you're in the same situation that you are already in, but in a more roundabout way. I think sensorship is really the biggest issue I see with these mining, or with mining centralization.

[0:26:24.6] JM: Tell me if this is the way that sensorship would carry out if it were to carry out. If you had a situation where, for example 90% of the hash power, the problem solving power of the Bitcoin network was dominated by a single power, a single mining pool for example, the way that transactions are processed is all the pending transactions, like if I try to send you X Bitcoin for a cup of coffee, then my transaction has to be processed by the Bitcoin network.

The way it gets processed is my transactions makes its way to a Bitcoin full node and it sits in the mempool, the mempool is the set of pending transactions. Then the Bitcoin miners get to choose which transactions they are going to put up for a candidate block. They get to choose. They can choose whichever set of transactions they want, and if they choose only the ones from people who are abiding by Chinese laws for example, like if that's what they wanted to do, if they wanted to basically narrow the set of transactions that they were willing to accept, they could totally do that, and they could gain control of the financial system in that fashion. They could decide to only solve Bitcoin bock puzzles for sets of transactions that are kosher according to them. Is that the way that sensorship would play out?

[0:27:52.3] SB: That is one of the ways. It's hard to tell if this is the attack as I haven't really thought too much about it, but that definitely is a very plausible way for sensorship to happen. Essentially, best case what that would do is that would delay these non-kosher transactions from getting confirmed in a timely fashion.

Worst case, you can even think of if a miner has a 90% of the hash power, they could just refuse to build on a block that contains this non-kosher transaction. They can actually start putting other miners out of business. That's a little more overt and a little easier to detect than these covert sensorship attack. Either one is not really great for a decentralized permissionless blockchains.

[0:28:42.9] JM: Do you think that this discussion is equivalent to a critique of proof of work and that this is an indication that maybe proof of stake is going to be a good solution to this kind of potential sensorship and centralization?

[0:28:59.0] SB: Yeah. I think what this work shows is that we very much are in the early stages. It's not saying that Bitcoin or Ethereum, or that permissionless blockchains have failed. It's just these are literally some of the first systems that have been deployed and are using this technology. Of course, there is going to be some issues.

I think these more drastic redesigns, like proof of stake that you mentioned, are the things that need to be seriously considered and they may help with this mining centralization problem, they may not. The way I look at this is let's try these things and see how they go. Let's measure the metrics that we care about and then let's make technical decisions based on those metrics and go from there.

[0:29:50.9] JM: By the way, this is a good place to bring up the fact that at Cornell, you actually have a Bitcoin network simulator. You've got a bunch of computers that are setup to resemble the Bitcoin network. Can you explain what that is?

[0:30:04.2] SB: Yeah. This was done by a senior student in our group. Essentially, what my understanding is I wasn't super intimately involved with this project. What my understanding is that you have a bunch of VMs that are running in a cluster downstairs and their inter-node latencies and bandwidth are what we measured from the actual Bitcoin network. It's a one-to-one simulation of the actual Bitcoin network.

[0:30:34.7] JM: Okay. Well, maybe I'll have to do a show with your fellow researcher. That sounds like a pretty interesting topic. To get into the work you're doing with Falcon, this is a research proposal, I guess is probably the right way to say it for a different way of networking in Bitcoin, or not necessarily a different way, but an augmentation to the Bitcoin network to accelerate block propagation.

I think we should just start maybe refreshing people on the simple process of a transaction, so a simple Bitcoin transaction. I send Bitcoin to you to pay for a cup of coffee. I want to write that transaction to the Bitcoin ledger. Explain how that works today and maybe some of the frictions that go along in that process.

[0:31:28.8] SB: Yeah, of course. First to that transaction gets broadcast to everyone in the network and it goes into mempool like you said before on the show. Then after that, the miners will take a group of transactions that they care about and want to put in the next block, and they will then solve that block.

Now all of this is fine as is and Falcon doesn't play a very active role in this process. However, once the miner has solved a block, there now is a danger of another miner solving a block. Just think about how block solving works is a good way of seeing why this is a problem.

How block solving works is you guess a number and you hash that number, and you basically are trying to figure out a number who's hash is very small. Because these hashes are cryptographically secure, doing this is very, very hard. That's what essentially proof of work is. If you can show me that you found a number who's hash is very small, I assume you tried many, many times to find such a number.

Now the thing is when a block is mined, it means that miner has found this magic number. The risk is if that block propagation takes a very long time, so now imagine that block propagation takes two minutes. What that means is in those two minutes some other miner could also have guessed and solve another block at the same height. Now this creates a conflict, which basically means one of those miners is out of luck.

One of their blocks will make it on the chain and the other block will not. Essentially what this does is it makes the chain less secure, because one of the miners who have essentially wasted their time and wasted their work. This does a lot of bad things for both the miners and for the security of the chain as a whole.

How block propagation works in the traditional peer-to-peer model is when the miner mines this block, they send it to their peers who then will validate every transaction in the block. Then they'll send it on to their neighbors. Then eventually you hope that the block will get propagated to enough people, so that no one else solves the block at the same time that you do. That process is very slow.

That actually in classical networking, that's the store and forward model, except here the store and the forwarding process takes a very, very long time, because you have to check all these signatures on these transactions.

[0:34:17.6] JM: Just to recap what you said. You've got these different full nodes that are competing on what is essentially a linear search for a solution and once they find a solution – you've got these nodes that are competing on a linear search and there are multiple solutions that could be found. There are alternative solutions. It's not like there is just one solution.

I mean, they could be setting, solving for different sets of transactions, or maybe there are multiple [inaudible 0:34:45.7] that end up hashing to the same, or end up also hashing to different solutions. You can end up with essentially a race condition where two nodes come across the solution at the same time, and then it's a race to see which node can propagate their new block faster and that just leads to a lot of wasted work. It can lead to just conflicts in the chain. How often does that happen? Does that happen on a regular basis where you have two nodes, or two mining pools that find competing alternate solutions?

[0:35:21.1] SB: I don't recall what the number was for Bitcoin before relay networks came on the scene. In Bitcoin from my measurement study, it happens less than half a percent of the time. On Ethereum though which does not have this relay network, but they also run their network at a much higher – they push their network much more than Bitcoin does, so the numbers also aren't exactly comparable. In Ethereum, that number is about 6% and it can actually go up. It can get pretty bad.

[0:35:51.2] JM: Because they have a lower block time, right?

[0:35:53.3] SB: Yes. They have a lower block time. They're producing blocks – they have a lower block time, which also probably plays a role in the high number. In Bitcoin I think it was a couple percent, if I remember correctly. Yeah, I'm not quite sure.

[SPONSOR MESSAGE]

[0:36:17.5] JM: Software Engineering Daily is brought to you by ConsenSys. Do you think blockchain technology is only used for cryptocurrency? Think again. ConsenSys develops tools and infrastructure to enable a decentralized future built on Ethereum, the most advanced blockchain development platform.

ConsenSys has hundreds of web3 developers that are building decentralized applications, focusing on world-changing ideas like creating a system for self-sovereign identity, managing supply chains, developing a more efficient electricity provider and much more.

Listeners, why continue to build the internet of today when you can build the internet of the future on the blockchain? ConsenSys is actively hiring talented software developers to help build the decentralized web.

Learn more about consensus projects and open source jobs at consensys.net/sedaily. That's C-O-N-S-E-N-S-Y-S.net/sedaily. Consensys.net/sedaily. Thanks again, ConsenSys.

[INTERVIEW CONTINUED]

[0:37:34.4] JM: This Falcon network that we're going to get to, so this is not just work that's relegated to Bitcoin. This is more like in any proof of work system this could accelerate and improve the security of the network.

[0:37:48.4] SB: I think that actually say something stronger. I think this actually will help any consensus algorithm, because in any Byzantine consensus algorithm, the latency of the network plays a critical role in performance and time it takes to converge and decide on a value. In

Bitcoin, we're talking about this wasted work in everything. These are bad things that happen in proof of work.

In other consensus protocols, there are other bad things that happen, but again it's a different set of bad things depending on the consensus protocol. A faster network is good in general. This isn't just applicable to proof of work. It's applicable to everything, to all coming to consensus protocols.

Let's say, we're talking about Ethereum, because I think that's like you said, that's a better real-world case. If 6% of the time you have competing change that develop and that's essentially – that means that two nodes, or two mining pools are competing to lay claim to the reward, the block reward for verifying a set of transactions. That means you essentially have competition and even more wasted work, because now there is going to be – 6% of the time there is a – is that a soft fork, is that what you would call it?

[0:39:09.9] SB: No. It's not a soft fork. It's a pruned block. Or a uniblock –

[0:39:13.2] JM: Pruned block. Okay.

[0:39:15.1] SB: Or in Ethereum it's called an uncle block. A soft fork is when the features of the underlying block changing. Soft forks and hard forks are different from these, but from this problem.

[0:39:28.2] JM: Right. Okay. Listeners, keep in mind my error there, we're not talking about forks here. We're talking about, so pruned blocks. We're talking about two rivalling versions of the transaction history one of those is going to lose and that version of history is going to get pruned, is that what you would say? It's a block that's going to get pruned?

[0:39:51.3] SB: Yeah, exactly.

[0:39:52.5] JM: Okay. Basically, one of the problems here, I think in some sense you're not presenting a solution to rivalling blocks being created. It's still going to be a feature of your proof of work system with Falcon within it, but what you are trying to suggest is a faster way of block propagation. Somebody finds a solution to a block, you want to notify the rest of the network faster so that that delta between person A finding a solution to a block and person B finding a competing solution, you want to reduce that delta, because if you can propagate that block to person B faster then person B is going to give up on looking for a different solution. This is, you are trying to accelerate block propagation, is that correct?

[0:40:46.0] SB: Yes. It's not as competitive as I would say. I think miners generally what they really want to do is they want to mine at the end of the longest chain that they know of, because that's how this protocol makes progress. They're more likely to get paid more if they mine on the chain with the most work, which is the longest chain.

Yeah, if person B hears about the block, then they'll start mining on top of it instead of mining at the previous block, which will produce a competing block, which can produce a competing – a block if something gets print.

[0:41:22.6] JM: One strategy for accelerating the block propagation is relaying, so you can have nodes in the network that are specifically devoted to pushing out updates to the blockchain, their relaying blocks. Explain what a relay node does.

[0:41:41.5] SB: Yeah. Relay nodes, like yes, exactly what you said, the purpose and life for a relay node is to relay these blocks very, very quickly to everyone else in the network. Now how these relay nodes achieve it is it depends on the individual relay network. What Falcon does in particular is – first of all, it doesn't validate every transaction in the block. It doesn't do a full block validation. It just partially validates the proof of work forwarding on the block.

Also, the second trick that Falcon uses is cut through routing, which is a very age-old trick in networking, which basically says instead of waiting for the full packet to arrive in the network, you send the bytes as you get them. Falcon does the exact same thing, but with Bitcoin blocks.

Instead of waiting for the full bloc to arrive, you just pass on the bytes as soon as you get them. What that does is each relay hop that you take, instead of each hop waiting for the full block to arrive and then forwarding it on, you actually just forward them on. Essentially, you paralyze the transmission across each hop, so each hop only adds a very minuscule amount of validation time, instead of having the store and forward approach.

[0:43:01.2] JM: How do you validate the proof of work if you don't have the full transmission?

[0:43:08.0] SB: Good question. The proof of work is that magic number that I said, is only found on the block header, which is the first 104 bytes. Yeah, which is the first 104 bytes at the block. The second I receive that first 104 bytes, I actually can tell was there a lot of work put into finding this block header? If that's true, then I forward on the block.

Could someone give me a bad block with a bad transaction in it afterwards? Sure, but it would cost them quite a bit to do it. They only get to broadcast one block's worth of transactions to the network. It's a very high cost for very little denial of service if they're trying to use it as a general service vector.

[0:43:59.8] JM: There have been relay networks before. I believe the Bitcoin fast relay network is one of them. Is your main improvement on that, the cut-through routing, or are there other improvements that you've made?

[0:44:13.3] SB: The cut-through routing is the main improvement. The Bitcoin relay networks successor Bitcoin fiber also uses cut-through routing. Cut-through routing is I think one of the Falcon's core contributions to the relay network space. Yes.

[0:44:26.3] JM: Interesting. Now, how many relay nodes are there on the Bitcoin network and who is – by the way, who is incentivized to make these relay nodes? Because it seems like a very charitable thing to spin up a relay node, because you could be using that node for hash power.

[0:44:46.0] SB: Yeah, exactly. You bring up a very good point. For Bitcoin, the current state is I have 10 relay nodes that we run, support knowledge of planning the Falcon network and then Bitcoin fiber is another relay network. Falcon has 10 nodes. I think the Bitcoin relay network has five or six, so you're looking at maybe 20 nodes in total, but the second part that you bring up is actually very interesting, which is why are we incentivized to do this.

Actually, this is a follow-on project to Falcon that hasn't been announced yet. We're actually building another relay network, which is more general purpose than Falcon. Falcon is built for Bitcoin, this relay network is going to be built for pretty much any consensus protocol that wants to use it.

The aim is to be trustless and it's also commercial enterprise. Charges a fee for using it and also it's trustless, which means that these cryptocurrencies can now rely on the relay network for performance. In Bitcoin after Falcon and fiber showed up, they can increase the block size quite drastically even, and they can process it much higher throughputs.

However, then it puts the relay networks in this awkward situation of if the relay network decides I want to sensor this transaction, or I want to sensor this block, they have a lot of power. This next project which we're calling a block trout is really a trustless version of Falcon, so that it's not put in the centralized situation that Falcon would be if people started designing protocols assuming that they have this relay network. That's really what the core insights are behind in this next project that we're doing.

[0:46:44.2] JM: Well, depending on what kind of commercialization you're going towards, you can sign me up for your ICO, or your presale. This attempted an enterprise, or I'm sorry – well yeah, an enterprise like a pay to play Falcon network that is trustless. Does this overlap with your research into consortium blockchains? Because I mean, you would need to have a customer to pay for that network. I can't imagine – I mean, Bitcoin or Ethereum if they wanted to use that, there would need to be some agreement within the network to use that thing. I'm trying to imagine, who would the customer be for this kind of pay to play Falcon network?

[0:47:26.2] SB: Yeah. We have a large set of incentives worked out, but it's not so much as like a pay to play thing. Blocks really does is it makes a decision on each network. It decides, "Do I serve this network or not?" That's really the only decision that the block trout network can make.

Now, if it makes a decision to serve a network, the high-level idea is for some number of transactions per second, which is going to be higher than what we are at today of course, it's going to be free as a trial run kind of thing. Then if you want to push it up into very, very high transaction throughputs, and we're targeting thousands of transactions, but again we need to demonstrate this on an actual network first, at least on our experiments to show that this is possible.

We're aiming for thousands of transactions. To get there, some of the transactions will have to subsidize the relay network. The idea is with so much capacity, you actually bring down the cost for everyone. We're really trying to aim for a world where the miners, the users and the network are all better off than they are today. That's really what we're aiming for. We're not trying to extort money out of permissionless blockchains or something.

We're really looking at it in that way and we're really looking to – I think the commercial enterprise, I think is a good idea just because it provides a level of support that individual is running relay networks can't do. You can start doing much cooler things on top of them. On top of an enterprise-backed relay network than you can with a grad student-backed relay network.

[0:49:19.1] JM: Okay. That makes sense. I know we're nearing the end of our time. One thing I found in my conversation with Joseph Bonneau was that when you ask somebody who is very measured and diplomatic and a very deep thinker in this space for some heretical thoughts, like what are the – I think I asked him some questions about things that he believes about cryptocurrencies that are heretical, or that most people – if you walked into a Bitcoin conference, what would most people disagree with him on?

He gave me some really interesting answers. I'd love to ask you essentially the same thing. Are there things in the cryptocurrency world where maybe you see them on Twitter all the time, or you see them in blog posts and they're just taken as these dogmas, where you're like, "No, that's just not the case." Things that you disagree with most people on that you can tell me about.

[0:50:13.6] SB: I think a lot of the disagreements that I would have actually are put in the decentralization blog post and paper, which is that these systems are really the version one. I think there's a lot of dogma in some communities around the way we have things now are sort of perfect. We don't want to touch or change anything. I think that's probably the thing I disagree with the most and I think this is really the version one of these protocols.

You're going to start seeing blockchains that look very different. In 10 years, we still be – would the blockchain design look very different than what Bitcoin and Ethereum look like today? Yes, I think that's true. I think embracing that change is something that some communities don't do.

Yeah, I'm not sure if I have anything that's particularly dogmatic or controversial, like outside of that though. It's more like, I try to keep an open mind about these things, because I've seen surprising things happen with business models or ideas that I'm like, there's no possible way this can work and then it turned into something big.

When Bitcoin first came out, I wasn't really quite sure what to think of it. I obviously wasn't in grad school and knee deep in understanding this stuff well. This seems like it would be just like a passing fad. Now five years later or so, I'm like in the cryptocurrency space and knee-deep, super excited about it. I think I've been wrong enough times to not hold too many dogmatic beliefs.

[0:52:01.9] JM: Yeah. I mean, when I was – some very early shows I did in this space, I interviewed some Bitcoin people who were very sophisticated and then I talk to some – casually on the side as well. I was convinced by what they said that Ethereum was never going to work. That this was just a disaster and it was going to end in tears. That's obviously not been the case thus far. I mean, I guess you got to take everything with a grain of salt and I don't know. It's a hard area to assess.

[0:52:34.5] SB: Yeah. I do think this area has a lot of promise though, and I can see these problems in how we do things today that this area has potential to address. That front I remain very optimistic about this area.

[0:52:49.9] JM: Well, Soumya thanks for coming on the show. It's been great. We talked about a lot of different stuff and I'm sure having you on in the future would be great. I think this is a going to be a popular episode. Maybe as your research develops around consortium blockchains, we can do another show on that.

[0:53:03.8] SB: Yeah, of course. Yeah, I'd be happy to. It was great being on the show. Yeah, I didn't notice the time flying.

[0:53:09.7] JM: Okay. Well, that's great to hear.

[0:53:11.0] SB: Yeah, it was great.

[0:53:11.6] JM: Great to hear.

[END OF INTERVIEW]

[0:53:15.5] JM: We are running an experiment to find out if Software Engineering Daily listeners are above average engineers. At triplebyte.com/sedaily you can take a quiz to help us gather data. I took the quiz and it covered a wide range of topics, general programming ability, a little security, a little system design. It was a nice short test to measure how my practical engineering skills have changed since I started this podcast.

I will admit, although I've gotten better at talking about software engineering, I have definitely gotten worse at actually writing code and doing software engineering myself. If you want to check out that quiz yourself and help us gather data, you can take that quiz at triplybyte.com/sedaily. In a few weeks we're going to take a look at the results and we're going to find out if SE Daily listeners are above average.

If you're looking for a job, Triplebyte is a great place to start your search, fast-tracking you at hundreds of top tech companies. Triplebyte takes engineers seriously and does not waste their time. I recommend checking it out at triplebyte.com/sedaily. That's tripleB-Y-T-E.com/sedaily.

Thank you Triplebyte for being a sponsor.

[END]