Eric: So really excited to bring you the episode with Benedikt Bünz. He is the chief scientist over at Espresso Systems. Now, last November I had on Mike Mosier. He is the former FinCEN Director and he's with Espresso Systems as well on the team to talk about FATF recent virtual asset provider released.

That was last November. That seems 10 years ago, right? Particularly after the events of this past week with Celsius and also three, three Hours Capital bringing down the market effectively and of course contributed to by the federal reserve. So when I found out that Benedict was on the team, I was super excited.

I said, wow, this is serious. So I'm really excited to bring this podcast because it allows me to do a deep dive on zero knowledge proofs, which is something that after all these episodes, I really haven't done the deep dive. And then we can also get into some really cool things that Espresso is doing with scalability and configurable privacy.

This is very exciting stuff. They're cutting edge. I'm a super big fan of the project. And again, I made a reference to the fact that this intro is being done on June the 16th. And so there's plenty of commentary out there about what's going on. Certainly encourage people to keep their head about 'em.

But in, and at a future episode, I would like to do a retrospective and tease out what it really means in the long term, not now because there's too much noise and there's plenty of places you can go for the real time stuff. This podcast is, has a different focus, but I will say this a lot of people in the space often comment that 95% of the projects today, the coins won't be here in five years or sooner.

People have different handicaps on that time. But as my client put to me no one is really quite prepared for how we get there, that process, and that process can be painful. This week can be part of that process. And I hate to say it, but there's probably more of that process to come, but that doesn't mean that everything's crumbling.

It just means that a lot of things are happening at once. And it's gonna take a while to sort itself out. But on this show, we don't really focus on events like this. We focus more on what's next and ways to think about the future, which I think is as important now as it is, as it's ever been. So if you look back at all the episodes that we've had on the encrypted economy, it's about building this encrypted economy and we get into substance.

We don't really get into hype. We don't get into token projects or like why something at a moon? We've never done that. In fact, I don't consider this a crypto podcast at all. I constantly say this. It is about encryption and how it, how there is an economy building around encryption. And what we discuss on this podcast is going to endure at the very least.

What you're gonna learn here is gonna better prepare you for what's next. So with that, I actually encourage you to share with others who you think will benefit. In fact, they, maybe they need it. Maybe they're too focused on the now and the current market movements. Maybe they need to take a step back and start to think about, Hey, what are the podcasts that are about building?

What podcasts are about the future, how we get there. Building the encrypted economy. So there's my pitch taking this episode. This is a great episode. Open your mind. Hope you learn a lot. And we got a bunch more common like this, building into the future. So really excited to bring you Benedict buns of espresso systems.

Enjoy welcome to the encrypted economy, a weekly podcast, featuring discussions, exploring the business laws, regulation, security, and technologies relating to digital assets and data. I Amer Hess founder of Heigel council. I have spent decades representing regulated exchanges, broker dealers, investment advisors, and all matter of FinTech companies.

For all things, touching electronic trading with a focus on new and developing technologies. So today on the encrypted economy, super excited to have Benedict buns on the podcast. He's the chief scientist over at espresso, which we'll be talking about. We had Mike Mosier on the podcast I guess a number of episodes ago.

And we were talking about a stealth project that he was involved in, and that stealth project was in fact espresso and I'm a fan of Benedict. When he when Michael told me that Benedict was associated with the project, I was like, no way. I'm like, this is great. And I got on the call and Benedict was on the call when he, and I was just so excited.

I was talking about bullet proofs and the stuff that he wrote, and I was going on and on about it. I'm I gotta admit I'm a bit of a fan. I'm a bit of a fan. He's Benedict's been involved in a lot of projects, a lot of research, and it's been exciting to see what he's been working on and how he's moved from project to project.

So definitely somebody to keep an eye on the space. And today we have him on the podcast. So Benedict welcome.

Benedikt: Welcome. It's quite the introduction. Thank you very much.

Eric: yeah, I know. I know. So let's start out a little bit with your background. And I know there's some recent news that you joined a 16 Z crypto research team, which is not in lieu is just another project that you're associated with.

You're also, we're talking finishing up his PhD. When he is, when he, outside of 24 hours a day, like where do you spend your time anyway, why don't you get a bit of background and we'll before we kick off into

Benedikt: it, right? Yeah. My name is Ben ins I'm for the last couple years, I've spent most of my time as or I'm a researcher, I'm an academic.

So I spend most of my time at Stanford finishing up my PhD in the applied cryptography group there. And I applied cryptography really. It's about the, I worked on the underlying techniques of blockchains, such as zero knowledge groups and some of the stuff that we might get into.

And one of the. We did about, now it's almost one and a half years ago together with Ben fish and others. And Charles lube, Ben fishes also, they also did their PhDs at Stanford. We found it ES expression systems which we'll talk about a little bit today. And there I'm the chief scientist.

So again, working on the hardcore research there, and I'm also now finishing up my PhD finally. It's been a great time, but I think it's time to move on and I'll start also working with this very exciting research lab that, and recent Horowitz or a 16 Z is building, and the goal of this research lab is really to be a like a high tech research lab has amazing academics for there, for the blockchain, for the blockchain crypto space.

The comparison that they always use, which I think is a very high bar, but maybe I hope that we can achieve it is there's this lab called deep mind, which is, does a lot of things for artificial intelligence and kind of the equivalent of that for crypto is what we wanna build there.

Eric: Excellent. So before we get into it, and you knew I was gonna ask you for this. Let's start off with the high level of the various zero knowledge proofs, I know you've done this before, but on this podcast you might have some people that have to get caught up start off with CK and, optimistic and then maybe hitting on snark, stark, Bulletproof, and last, but by far not the least plunk.

Benedikt: I think. Maybe it's even better to start off with what the heck is even a zero knowledge proof. I think, that's always a good point to, to start, even if you've heard it before. So zero knowledge proofs have two amazing properties. So at a very high level, they allow you to show that something is true without revealing why it's true.

So how does this help, or what does this mean? For example, I can have a transaction and encrypted transaction such that no one can see only the send and receive can see the contents of the transaction, but I can, then how do I verify that? For example, the sender actually had enough money to send this transaction.

This is where zero knowledge proofs come in, where I can give you a zero knowledge proof that the transaction is valid. And after reading this proof. You'll be convinced that the transaction is valid, but you have no idea why it is valid. So for example you have no idea how much money was being transferred.

So this is a great tool for privacy. The main thing is that these same tools, these zero knowledge proofs sometimes also refer to as EK Snarks. We can go into the classification in a second that they also have another property, which is that I can prove to you that some really complex transaction or some complex computation.

That takes an hour an hour to do say I can prove to you that this computation was done correctly. And then checking the proof is much, much faster than redoing the computation. So for example, this is being used in something called rollups something that we're also now working on at espresso system, where I can prove to you where I can take a bunch of transactions.

And instead of everybody in the blockchain system, having to verify all of these transactions, I give you one proof, one short proof. It's also really short that all of these transactions are valid and all you need to do is check the ability of the proof, which is much more efficient than checking all of the transactions.

So it's a tool for both privacy and for scalability.

Eric: Awesome. So then I guess we'll take our next baby step. Does the Z basically the different rollups before diving into the snor Starks bulletproofs et cetera?

Benedikt: So there's many different, there's many different proof systems.

So these are different, like just like how there's many different browsers or how there's many different, I don't know, right? One is the class a zero knowledge proof and one then there's like brand name almost in instantiations of it and they have different properties.

So for example, I've worked on bulletproofs, which is particularly efficient for these, for basically the, this privacy aspect of proving that a transaction is valid without revealing, the amounts. And that for example, is being used in Monero. For privacy. And then there's other proof systems like star wear have, Starks is a proof system developed by star wear which, it's better on some efficiency metrics.

So there's basically all of these efficiency metrics. Like how hard is it to create a proof? How hard is it to verify it? How big is the prove all of these dimensions? And that's why there's many different proof systems which are optimal for different applications. I think, honestly, I know people love to focus on the differences.

I think that's. They're not that important. I think it's much more important to understand how these proof, I say that while most of my work is on trying to develop new and better proof systems, right? Like for me, it's very important, but I think for the general users of blockchains and people trying to understand the space, it's more important to understand at a high level what the what these proof systems does rather than trying to understand the distinct differences and, there's different ones work basically better for different applications.

Right.

Eric: And, but so maybe instead of going into each and every one, maybe just do a little deeper dive on the specific tradeoffs across each, I think you referred to it, but

Benedikt: yeah. So the tradeoffs are, I guess there's, couple dimensions which are really important.

So one of them is the. It seems to be the case that for example, for scaling so taking a bunch of transactions and creating one proof out of, like that, that all of these transactions are valid. So for rollups, it seems like that maybe it's okay to have a slightly larger proof size if the prover is like if in turn, the prover is much faster.

There seems to be a tradeoff, basically a little bit faster prover. So being able to create a proof for more transactions or doing this faster, seems to lead like the proof systems that are better on that they seem to have better they seem to have slightly larger proof. So for example, star is an example of that.

The other one, the, another dimension is that we have very efficient and good proof systems, but some of them use something that is called a trusted setup. So some of these SNAs these pairing based SNAs they use a trusted setup. And what that means is that there's a party that has to create the proving key and the verification key.

So the prover has a proven key and the verify has a verification key. And if the party who creates that is somehow they use some secret, some secret randomness and afterwards they should discard that they should destroy, like ideally destroy the hard drive that they use, such that no one can recover this.

The problem is if they don't do that, or if they're malicious, then they can actually break the proof system. So they can prove to you that two plus two is equal to five or that a transaction that, Creates money out of thin air is valid. And we have, this is this is of course, something that ideally, especially in the blockchain setting, you shouldn't wanna have it's much better to not have such a trust, etcetera.

And that was one of the reasons why we developed bulletproofs because it was a very efficient proof system that does not rely on such a trusted setup. Versus for example, the proof system used in Zcash relies on such a trusted setup. And they actually had an issue where, they had to disclose a vulnerability such that, we don't think it was exploited, but there was a significant issue with this trusted setup.

And it just shows you the danger of these things. And, as Chris photographers, we work really hard on, on trying to minimize the risks there and, for example, develop proof systems without trusted

Eric: setup. And is this the trust is set up distinct from what goes into an optimistic rollup because there you're rolling up a number of different transactions at once.

Yeah. And so there's a delay

Benedikt: There's two kinds of, so this is now on the rollup side, so rollups are about combining transactions and trying to verify, all of these transactions at once. So

Eric: when you just to be, just for clarification, so that there, what you were talking about with the trust that set up really goes to the zero knowledge proof itself.

Benedikt: This is in the zero knowledge prove itself versus the rollup. Okay. Yeah. So in the rollup, there's two, basically right now we know two variants of rollups we know ZK roll up, which uses, one of these zero knowledge, proof systems that I've talked about like Starks or some variants of SN or plug, is a, is a.

Seems to be one that is really has a lot of good applications right now. So that is ZK roll up. So you prove that all these transactions are valid and optimistic, roll up works slightly different. It doesn't actually use zero knowledge proves, which has, and I'll talk about the traders in a second, but the idea of optimistic roll up is that I'm just gonna claim that all of these transactions are valid, and no one is going to verify.

But then afterwards it basically says from, instead of immediately, everybody verifies, it says that we can do fraud proofs so that, then someone checks the transactions and if they discover a mistake, if some, one of the transactions isn't valid, then basically there is a mechanism such that they can point out.

This is, this transaction was invalid and, there's some economic incentives and a system built in place such that, the person who claimed that a transaction was invalid when it was valid. When in fact was invalid is getting punished they have to pay some fine or they get slashed some amount.

So that is optimistic roll up. Optimistic roll up does not use zero knowledge proofs. And the benefit of that is that you, so the big benefit of optimistic roll up is that at least today it is much faster, and you can take more transactions and aggregate them at once. It really is quite efficient, and you can, do this with very complicated transactions.

And the downside, the big downside of optimistic roll up is that. When I see a transaction, in, in such a roll up, I'm not sure yet whether it's valid, I need to wait. It's like up to a week for to, because there's this challenge, period. Someone might challenge this block and say oh, here, there was a transaction that is invalid.

And that really creates complications. And like you have to wait basically a whole week in order to get your funds out of the rollup. And yeah, personally, I'm like, I think that optimistic rollups are, I think there's, it's great to have a diversity of ideas. We should have many as many ideas as possible, but I personally think that optimistic rollups are good right now, but I don't know.

I think ZK rollups are getting better and better. And the zero knowledge, proof systems are getting better. And I really think that this is where the future. It's heading, I've been, no, I,

Eric: I, I, it's actually interesting cuz I've heard, a lot of I've heard a lot about, the ZK roll up and the optimistic roll up and every time I hear it I think to myself how is optimistic roll up the future, but I'll ask you like, what does optimistic, going forward from here?

There's a lot of, I agree with you. You definitely hear a lot about the big improvements being made in in, in zero knowledge proofs. The question is what continues to make sense for optimistic. Rollups

Benedikt: Optimistic rollups make a lot of sense right now. I think that there's always you have to think about the now and today simply because they're the only system that, so with the ZK rollups we can right now roll up Simple transfers of value.

Are like this like decentralized exchanges, but basically simple I just send you some money, but we know that blockchains are, have much more complicated, smart contracts, for example. Which do much more complicated computations? And we don't yet have that. This is for example, something that, that at espresso systems we're working on, we don't yet have good rollups for arbitrary smart contracts.

That's just a ZK. Rollups sorry, we don't have ZK rollups for arbitrary, smart contracts, but we have optimistic rollups for arbitrary, smart contracts. So in the now day, if you wanna roll up smart, If you wanna roll up complicated transactions, it's not that, one is better than the other. It's that optimistic roll up works and Zika roll up does not yet work.

And I guess the big question is whether I think that if we had the amazing Zika roll up and everything was there, then we would be clear that we would take it. It's basically the question, I think, I guess if I was an optimistic roll up person, I would say it's like the fusion of the like fusion energy, which is always 30 years away.

No matter what you ask I, I personally obviously don't think so. I'm an, no pun intended, but I'm an optimist in that regard like that ZK rollups will get better and better. Certainly we need solutions. Now we and who knows, right? Like it could be that optimistic rollups are always, slightly more efficient.

They just have this large delay period. So yeah I don't know. There's it's really the space is so exciting because so much innovation is happening and it's really hard to see, where the space is gonna be in, in five years. But yeah, I'm a I'm a big optimist Onika rollups

Eric: right.

And so I've, you mentioned plunk and your confidence level in plunk, and I've also heard a number of other projects express a lot of confidence in, in, in plunk. So maybe you could, what makes it so exciting?

Benedikt: Yeah, I think that that seems to sit in, in, in a sweet spot of like very efficient verification, very small proofs, but like one of the things that so plunk is actually there's kind.

These proof systems are nowadays like set up out of multiple components. So there's basically, okay. There are basically two components. One of them is that you, I always talk, we do a proof system for some computation, right? So like normally a computation can be expressed in many different ways.

For example, it can be written in Python, it can be written in C plus it can be written in the assembly code that your computer actually interprets. And then there's different compilation steps. From, one of these higher level representations of a program of a computation to a lower level pro representation and kind of the same thing has started to exist for these proof systems and is really the main component of PLU is a slightly different kind of programming language for how I express computations.

So instead of, I dunno, C it uses like something, I don't know, C plus and it is a really, interesting and flexible like representation of how I how I represent computations uses something with like simple, the simplest way that, that these proof systems used to express computations is just as multiplication and addition.

So if I wanna do anything, I have to write it as a bunch of multiplication and additions. And we know actually that any, anything I wanna compute I can write as multiplication and additions because that's literally what your CPU does, but it's helpful to have more expressive or more intermediate gates.

For example, I might wanna do a hash functions are really important or other operations and or, and if gate or, there's different operations that I might wanna do. And plunk basically allows you to build these custom gates is what they're called very efficiently.

And really gives a lot. A lot more flexibility, which allows people to design, really elegant solutions and things around it. And it's really become almost like not just one proof system, but a family of proof systems that all use this. This kind of same higher level plunk language.

Eric: Excellent. So the encrypted economy covers a lot of the technology for homomorphic encryption secure multi-party computation even trusted execution environments, but basically, technologies that are, tend to be working off of very large data sets and large computations.

When you're talking about the multiplication, in addition, immediately snap into that's like a fully homomorphic concept versus like a partial homomorphic concept is like just pluses and minuses. So question what I find interesting is trying to understand where these two worlds actually.

Intersect and, because oftentimes you have conversations with people and they're very deep in the weeds on zero knowledge. And then you have conversations with people who are very involved in secure multiparty computation and homomorphic encryption, and zero knowledge doesn't really come up too much.

And yet they seem in many ways, they're both relating to encryption, very different types of objectives. But I guess the question is how do you see them intersecting? Or is that something not even think about?

Benedikt: No, of course. I think about that and, they're all I think they're actually all, like they all intersect in many different ways in both the tools that they use underneath the hood.

For example a lot of zero knowledge proves use Like they use homomorphic encryption to do the additions much more efficiently. So for example, in these zero knowledge proofs, oftentimes like this is of course simplifying, additions are very easy because we have good homomorphic, like additive homomorphic encryption, which allows you to do additions and subtraction very efficiently.

If I wanna add two vectors, that's very easily done what's a little bit harder and where we have to work a little bit harder are the multiplication. So usually for example, we measure the complexity of, the, these student knowledge proof systems in the number of multiplication, not in the number of additions.

And the reason is really because we have good. Additive homomorphic encryption additions, but we don't have as powerful multi homomorphic encryption or, and this is these are just like ways where really, when you go deep into Louise, we, the weeds, these things intersect, with multiparty computation, for example you can think of zero knowledge proofs as a very special kind.

There are also many intersections, but you can think of zero knowledge proofs as a very ki special kind of multiparty computation where the computation is basically that the prover, Proofs that some statement is true, and you can cast that as a multi-party computation.

And there's also some proof systems which are built based of multi-party computation, usually the advantages of those they're very large. So I don't think they're very good for the blockchain space, but the advantage of those is that there, one, one big I didn't one big dimension that I didn't mention dimension that I didn't mention in the proof systems is the security against quantum computers.

So whenever you talk to photographers, like at some point quantum computers come up because a lot of the cryptography that we've built over the last 30, 40 years is not secure against quantum computers, but now and we don't have quantum computers yet. But if they ever come online, they can factor integers, which breaks a lot of ours.

Computation and then that's a big danger. And so for example these zero knowledge proof systems that are built from MPC, multipart computation, they're actually, you can build proof systems that are secure against quantum computers, which is really exciting. They're just not as efficient yet as the ones that we use today.

Eric: And while we're on it and maybe for some of this is going down the rabbit hole, but I always challenge my listeners bit on these podcasts. So it's, they're used to it. I've had Brett Hemingway FA on a number of times and, we've talked, I don't know if you're familiar with him.

He's a university of Pennsylvania professor and we've talked just about the value of trusted execution environments and how it's almost like a dirty way or a cheap way of getting a lot of the benefits of homomorphic encryption without necessarily implementing a full, fully homomorphic, encrypted system where using the SGX from Intel.

Is this something that you deploy, like when you're implementing the systems that you, that leverage plunk is that the way, is that something that, that actually works or is, are there system or technical limitations against using those kind of environments?

Benedikt: Yeah, I think the biggest limitation, I think it, it really depends on the application, trusted execution is the.

The cryptographers cheat code, because it can do almost everything really fast, that cryptography can do, but it has one huge caveat, which is that it's like it's really differ, it's like almost impossible to get. Totally. So for example, with Intel SGX, which is one of these trusted executions, there is every year at the security conferences, there's some bug, that say oh, we've completely broken it.

And that's the danger, right? And I think it really depends on your application, whether it makes sense to use trusted execution, would I rely my blockchain on trusted execution where if any person can find any bug in, this and it's really it's a completely different attack model in that, What trusted execution says is that the attacker has physical access, right?

Like they can crack open their laptop or their com their server. And they can like, literally they can buy as many of them as they want to. And they can see the chips and can do, there's like laser attacks and electric, there's crazy stuff that people can do. And they say even still, they shouldn't be able to extract some key, there's some secret key embedded on this hardware.

And basically, what Intel is claiming with SDX is that even though someone has access, like physical access to it and can run programs on it and everything, they cannot extract this tree key. And that's, that's very different from me saying I cannot extract your key, where you sit behind a firewall on the other side of the world, I still, maybe through an internet connection, maybe a hacker has a chance, but it's much, much harder for an attacker, it's completely different game. And it's turned out like many times that it's really, there's, some, someone is able to extract such a key and then, into some update.

And then in the next version, it gets harder. So all this being said is I don't trust, trusted hardware, for example, for an application like a blockchain, because what this would mean is that if anybody at anywhere in the world can break any of these chips, the system is completely like is useless, is broken.

They can steal all the money. What I do think it is useful for and what it is being used for is for example, like what is. It's, if someone has a huge incentives to break these things, they can break them. But for example, if you want to store some sensitive information on your phone, like your wallet, like for wallet management, I think it's a really good application.

Like the problem with wallet management is that you want to run your wallet on your phone in some secure manner, but you also run a bunch of other apps on your phone and, you want to have an added layer of security, such that these other apps cannot somehow get your private, say you installed a malicious app.

And I think these kinds of applications locations and stuff related to that, also making it easier to manage keys, people are really bad at managing, like they forget their own passwords. Like managing passwords, managing keys really hard for users.

And I think that trusted hard work has a lot of roles. In, in, in kind of these applications where, it's also if one of them breaks then like one user loses their money, right? If you can break one trusted hardware there, then I can maybe steal the money of one user, which is bad as something that we don't want, but it doesn't mean that the entire Blockchain or the entire system goes down.

So I would trust it for an individual person's security where they can also opt in to use it or not use it, but I wouldn't trust it for the entire systems in entire systems security.

Eric: The wallet application is something where you've had the, what was it? Multi-party computation Alliance or the MPC and that the membership there has doubled or plus.

Yeah. And it's all been from, firms that are exploring the enterprise wallet management, like unbounded technologies. We had Nigel smart on the podcast while ago before he, I guess they were AC acquired by Coinbase . Yeah. Big deal there. So the wallet side of it has been explored in detail.

What do you think? I'm gonna press it a little bit, just speculate. What would you see as the next? Application the most logical application of intersecting, the, what I'll just call generally the blockchain space. And we'll just keep it very wide with these homomorphic encrypted or, the combination of homomorphic encryption with secure multiparty computation with other encryption primitives, what are your what would be your take?

Benedikt: Yeah, one I guess one application that is not that far yet from the wallet application, but like an extension of it is that it used to be that the idea of a wallet is you create your key. You create your key, you put it somewhere in, like you literally put it on of paper in like cold storage.

Can't be. And only when you wanna move the money, which happens like once a year or something, you get it out through some complex security measurement and something like that. This is true for a lot of applications, say if I have Bitcoin keys, but this isn't necessarily true anymore in a world of defi where people wanna do things with their crypto like they wanna, trade it. They wanna, yield farm, they wanna do different things of it. And for that, they need to access their private keys many times and maybe in, in different, complicated ways. And that actually makes. The MPC application more interesting because it's very different from, oh, I just, once in a blue moon, I need to reconstruct it, to send the money or I need to constantly have basically access to these private keys.

That's certainly much more of an E evolution on this technology, my friend, DMA coma and Kogan at Och they're I think they're working on something like that. I'm happy to connect you if you wanna have him on he's also from the same. He also did the PhD in the same group, this, a whole Arma of people.

And so the other application, that I think is more, more long term, but is really exciting is these very private, one, one thing that, that like very private smart contract where, for example, I wanna play, I don't know, like a fun one is we wanna play let me see how this, we wanna play, like we wanna compute something on the blockchain together.

But we don't even want the blockchain to see the smart contract. And so for example, 1, 1, 1 interesting one would be, say we run an auction on a blockchain between many different parties. And we want that, like in theory with MPC, what you can do is you run an auction and the winner knows say it's an I don't know some there's different kinds of auction, like an ascending auction or but yeah, there's different kinds of auctions, but say we run an auction and we could have an auction such that the winner knows how that they've won, obviously, and they know how much they pay.

But, and the seller knows how much they're selling it for, but or the auctioneer knows how much they, the winning bid was, but no one else learns any information. So for example, you don't know how much the other people bid. You don't know how much say it's a, there's second price auctions where, you, you bid something, but what you pay is actually the bit of the second highest bid.

So you wouldn't maybe even know how much the winner bid and all of these information, we can keep private and run this as an auction on a blockchain. And at the end of the day, we still, get to run this auction and have a winner. And I think, applications like that, that really involve different parties and, something that they wanna do together and something that they wanna keep private, that could be the next frontier.

Eric: Exciting. All right. So just about 40 minutes in I wanna talk about espresso now. I think everybody knew that I was gonna, dive into this, into that first part with you but let's start switching over to espresso cause that's an exciting pro project for sure. And so let's talk about the problems.

Now let's talk about what espresso is and what kind of problems it's trying to address.

Benedikt: It's so espresso is really, it's, I enjoy talking about the first part because it also is, highly related to what we do at espresso systems. So it's really, you. A single shot solution for both privacy and scalability and

Eric: that's and that's pun intended, right?

Yeah,

Benedikt: of course. But it's the and it heavily relies on, where world experts and these zero knowledge proofs and it really relies on these Z case arcs. We really, it really relies on this technology to build both, an, a novel privacy solution where we have I think, a different approach to privacy that maybe some other projects have had, and also a new approach to scalability through Ze, through ZK rollups, but not, traditionally ZK rollups have been seen as a layer, two solution.

So an improvement to Ethereum, but we. There's actually a lot of, there's some problems with that. And there's a, a lot of benefits that we can have by building a new solution from the ground up that, has native rollups integrated. And yeah, that, that has a lot of efficiency and scalability benefits.

Eric: Excellent. And then in talking about, what espresso is designed to do, we're talking about, from my understanding it's configurable privacy on the blockchain, and I think we talked early on about the potential use cases for that.

Obviously stablecoin, that's a big one particularly in with some of the most recent problems with stablecoin. Do you

Benedikt: wanna maybe yeah. Yeah. Yeah. So what I guess, like we should first expect play the listeners. What we mean by configurable privacy. Right now, if you look at the blockchain ecosystem there, there's kind of two extremes.

So if you look at Bitcoin and Ethereum, these systems actually have, even though sometimes they're touted for privacy, they actually have terrible privacy. If I know your Ethereum address, I can just literally go to the blockchain and see all of your transactions. And that is that is I can see exactly how much money basically you make, how much, like it's if your bank account was, if you tweeted every little transaction from your bank account, and then on the other hand, we have good, privacy solutions like Manero or Z cash, which make all of your transactions private.

And that is good for some use cases. But I think there's a lot of use cases that kind of sit in the middle. Where you don't want every single transaction to be fully public, but also maybe you don't need or want, or for regulatory reasons aren't allowed to have full privacy. So the one that you already mention are these stable so stable coins and I'm talking about this kind of algorithmic stables, and then there's like stables issued by a company, say circle or something like that.

Or might issue one or, I Don’t know one of these, like that, that these. Stable coins where, you know, the stable coin issuer. So they, the way that they work is you give them a dollar, you give circle a dollar and you get one UD DC, like a digital dollar back, and that then you can do everything with it, but there's a lot of regulation around these stable coins because it's basically digital money.

The us government, for example has a lot to say about that. And the regulation for example, says that the stablecoin issuer should be able to view and freeze assets. If there's, for example, a subpoena or if there's a court order there should be able to do that.

And what we enable with kind of the configurable privacy is we enable, that you can make these stables private to its the public, but then. There is, a viewing key or there can be a viewing key. It's optional, it's configurable and optional, a freezing key, which allows you to view and freeze assets, for example.

So you can, the issuer could, if they wanted to, they could view these assets, or we can even on top of that, you can distribute the key, the viewing key, if you say I don't really like it. If there's one issue that can see all the transactions, that's a reasonable concern, it's so configurable that we can even split this key amongst many different parties and say that, only if they all agree, if they all sign off, then they can view the transactions.

So it really is, I think it gives you it. What we're trying to build is a solution that, that opens up this entire middle space between full privacy. And full transparency, but give you the best of both worlds, which I think is, for example, for these stablecoin applications, we don't have any sort of private stablecoin yet.

So this could enable really the, our goal is to enable the first private stable con.

Eric: And so I, and I know this came, this popped up on your website, a couple of different places that you are actually working with circle. Yeah. Yeah.

Benedikt: We're collaborating with them.

We have a lot of talks with them. I think they're, really excited about this and, really excited to see where this goes.

Eric: And I know like tether I remember reading some surprise in the community as to tethers ability to actually freeze transactions a while ago.

And there was some controversy like, oh, how is it that tether can do this? Are they using a similar methodology?

Benedikt: So freezing is, yeah, freezing is very so it's just built into their smart, so they have a smart contract and it's built into their smart contract and they To defend tether here.

This isn't really TE's choice. I think this is there's essentially regulation that requires tether to do this. The, or it requires tether to have the ability the way that it works is so it's very easy, or it's somewhat easy to do this. If you have a non-private stablecoin like tether stablecoin because you can basically just program it into the smart contract where you have, like freezing is just a flag.

And then if it's frozen, then you can, move the money anymore. What makes it, like what is made this problem challenging for us, but what we've solved like this is actually now published and open sourced and is and is going to be launched soon on, first on a test net on the RI be test net.

So what makes this, so you can play around with it yourself, which is gonna be really exciting. So watch out for an announcement for that's coming soon. So yeah. What is what made it interesting for us is that we wanted to have, even tether shouldn't, without using their viewing key, or you can even have a different viewing and freezing key, the public shouldn't know, or they shouldn't be, a table somewhere that says, who has which asset and how much and what is frozen.

And the basically what we enable is the ability to freeze, an asset without even, knowing while still re retaining the privacy of all of the other users. So if you're not being frozen, then, you don't. It doesn't even leak how much money you have and whether you have any coins and so on and so forth.

So really the goal is to have privacy by default and security by default, and then only in, in very extreme circumstances. When you, for example, have to require with comp apply with law enforcement to. Still have a solution for that,

Eric: right? For anybody who's ever been through I'm an attorney.

So anybody who's ever been through like a cybersecurity hack of a client, you, you really appreciate the ability to freeze. All that stuff is, don't touch my coins is great, but then, you'll be surprised at how many people like turn around and say hold it.

That was then, yeah. But right now there's. You know exactly. So it, it does have, it's hard to deny it's beneficial. It's beneficial uses. And so it's interesting. So you've, so espresso has, there's a lot of different projects that espresso's undertaking. And I guess it's like this wide open canvas, obviously the privacy component of it and linking it into things like freezing also as part of that use cases.

So one of the big things that you've talked about is the scalability of espresso and how. Probably one of the bigger differentiators of espresso is this great scalability and I don't know, maybe you could talk a little bit about, how you've achieved that, and how you see yourself even maximizing on that differentiator.

Certainly stable coins are one way to hit that scalability side of it, but,

Benedikt: right. Yeah. I think one of the things that we noticed so re know really this privacy, this customizable asset privacy, we call that cap or Cape customizable asset privacy on Ethereum.

Because we've deployed this as an Ethereum smart contract, but what we noticed in in, in the process and, we were aware of that before, really this working on Cape really drove this home is that wall Ethereum is great. And we're huge Ethereum fans that, for applications like that, that for example, use a lot of cryptography.

Ethereum just isn't that scalable today. Like it just doesn't really work. Your transaction fees might be in the tens or sometimes even hundreds of dollars, which just for a normal user, like that's too much, like they wouldn't use it. What we thought is that we can build a separate system, which is still linked to Ethereum through a bridge.

So is, you can really see as still part of the Ethereum ecosystem, but which uses some of the same technology that we're experts in only these CK rollups or these zero knowledge proofs and these CK rollups in order to scale transactions and maybe even, scale arbitrary scale things like Cape, but even, then maybe even more generally scale arbitrary Ethereum transactions.

And that is what we are now heavily working on. We'll we're working on kind of a first test net for that. And our approach, the way that our approach differentiates is that coming back to these rollups these rollups have one, both Zika and optimistic rollups they have one key limitation still at this point, which is that you can do these rollups and, the checking, the proof is very simple, but it turns out that there's a data availability attack, which is that if the roll up server inserts, some random transactions then, and doesn't tell the world about it.

And it then creates a roll up proof. The roll up proofs ensures that all of the transactions are valid, but what it doesn't ensure is that you have afterwards, you basically need some data, some, Merkel tree path, but you need some data in order to create the next transaction in order to create, like in order to basically still be able to transfer value, you need some additional information and what a roll up server could do if they're malicious is that they don't release that data and they basically can hold your money hostage.

And the best solution or the solution that these roll up servers use today is that basically they need to publish some, Limited amount, some data per transaction still on the blockchain. So there's needs to be some additional data. That, that kind of a short sum, a summary of all of the transactions that needs to get published on the blockchain and essentially what this means.

Why is this a limitation? Essentially what this means is that the amount of scaling that you can get through a roller like that with so called on chain data availability is the term that people use is limited. It can give you a factor of 10 speed up, but it cannot go beyond that, right?

Like it's not an, even if you have many transactions, there's a limit to, how much speed up it can get you. And if we wanna get a hundred X or a thousand X speed up. We need a different solution. And what we found at espresso and what we are building is basically a solution that integrates this data availability.

So integrates a solution for keeping this data available so that everybody has access to it together with the roll up system. So build a highly integrated design there that, solves both problems at once and allows you to get to, speed and, really unlock the full potential of these.

Eric: And so for any project, any coin token that's trying to or even, I guess the, the main net however you wanna frame it they can actually. Use the Cape, your Cape to wrap all their tokens so they could basically have their whole ecosystem wrapped in your Cape.

exactly. Yeah. Yeah, no

Eric: So ecosystem, or really just it's really token by token,

Benedikt: token. So for Cape it's, the idea is so for, again like this the two components there's so for Cape, yeah. It allows you to wrap, tokens, which automatically gives them this one, you can wrap them in different way, it's customizable. So you can choose the color of your Cape basically. And basically decide what kind of privacy you want. For this token, do you want full privacy Z cash level privacy, we enable that. Do you want, like that there's a viewing key and that's, a, which brings privacy to, all of these different assets.

And then once, our layer, one chain is ready, which, will enable like it, it will be EVM compatible, so you can roll up and you can use it with any, just your normal EVM smart contract. Then, we add not just privacy, but also scalability to all of your smart contracts and assets.

So that's the two step plan. The privacy part is ready now. Cape is ready now. It's open source and come talk to us. If that is of interest in you, to you and the scalability part is what we're working on, on now. And. Will hopefully be ready soon and, I think which yeah I'm also very excited about.

Eric: And so how would you like the solutions that you're providing, would you view them as something akin to a, to there's more talk these days about modular modularity in these blockchain systems. Would you think of some of these whether the privacy or the scalability combined, would you think those as modular components that, Hey, you worry about X, and we'll worry about this, we'll handle a scalability and the privacy or is that maybe the wrong construct?

Benedikt: Yeah, no, certainly there's it depends what level of a, of abstraction you use. Of course there's modularity and, one of the great things about blockchains is that everybody can write a different smart contract and somehow they still interact with each other.

Even though, like they're written by different people on different sides of the world without knowing each other, like that is, really where a lot of the power of these blockchains come from. And we preserve that. I think then at the lower tech level, especially the scalability the privacy solution is very modular.

The scalability solution. We've actually found that by integrating some of these components. So not trying to do the roll up separately from the consensus system. So that's the consensus, the proof of state consensus and the roll up by end the data availability by basically having one solution for all of these three things together.

We're able to get a lot of benefits from that. So that's where we maybe broke the modularity. And design one unified system because it had such a benefit. I think the danger with these modular systems is that sometimes it's works great. And, it's a bunch of building blocks and you have a perfect Tetris and sometimes it becomes more of a Frankenstein, right?

Where, you attach different parts and, it works together, but at every intersection there's, and I'm not a, I'm not and not an extremist in either direction. I think just in different situations, different approaches have been better.

Eric: And so when you're with regards to the scalable solution and privacy, obviously one use case is a stable coin, what types of projects do you think, do you see, like leveraging this the most right out the gate? Or, learning the most,

Benedikt: forget the game. Yeah. I think honestly I think that the privacy part, I think like my personal take is that the current privacy of, say like Ethereum is untenable for many real world applications.

If I'm a business and I wanna use the blockchain for any sort of, real world applications, whether that is transferring stable coins or whether that is some sort of token, that I've issued or some sort of defi do I really want all of my competitors to see all of my transactions, right?

Is that really what I want? This is worse privacy. If you think about it, this is significantly worse privacy than we have in our traditional financial world. Where in the traditional financial world. Sure. My bank knows all of my transactions. But my competitors, don't my, ex-partner, doesn't right.

Like my, that like all of these that's, I think we should at least get to the same level of privacy that we have in the traditional financial world and Cape certainly enables that and much more. So I think that, almost any token, like for any application benefits from the privacy, for the scalability aspect, we've obviously, build it in such a way that, more, more I think the biggest benefit comes maybe for more complex smart contracts.

So for example, like Cape transfers, like our own product, Cape transfers is something that we of course are top of our mind. When we build these. These smart when we build our scalability solution, but at the end of the day, we want we think that sort of, this can really be a scalability solution for all of the things that, run now have high fees on Ethereum or are, hopefully I think the most exciting ones, applications are always the ones that aren't even possible yet because the transaction fees are just too high, right?

It's when email came around, it's not that, it like made all of the snail mail much faster. It obviously did that as well and made it cheaper, but it just enabled, applications, sending a thousand email today for 0 cents that weren't even possible before.

So I think, the most exciting applications are always the ones that. We haven't even seen yet before

Eric: I let you go. And I've been testing you on this jellyfish. Tell us a little bit about jellyfish.

Benedikt: Yeah. One of the things that we have at as REO system is in order to build this technology we've built, I think frankly, even if we completely take me out, I think we can say we're one of the strongest cryptography teams in the entire space.

I think we have,

Eric: and with you and universe, anyway, keep going.

Benedikt: Who knows? But the, I think we have I lost count like 5, 6, 5 cryptography, PhDs, an incredibly strong team. And one of the things that we've built is a cryptographic library, with a lot of these tools, for example, using one of.

Probably the best implementation of plunk that exists out there. Even implementing a lot of the kind of those like different variants of plan, like hyper plunk and turbo plunk and we implemented, yeah, I know these,

We need more names

Benedikt: So yeah, some of these improvements that we also came up with ourselves implemented in jellyfish in this library and, in rust, which is, seems to be the best language that is being used for these systems.

So it's both very safe, it's very fast. And it's open source and anyone can check it out and use it. And yeah, this kind of. Contribution, in this space, you always have to, you build your own project, but like part of what I think you should do and what really helps the space and what makes it so amazing is that you really give back to the space a little

Eric: bit as well.

So before you, before we break off anything that I haven't asked you about that maybe I should have that you want to expand on? No, I think we was,

Benedikt: We covered a lot of ground. I think that this is I beat you a lot to the readers to, for the listeners to digest. I think, if you can, if any of the listeners can think of exciting applications, my email is Benedict espressos.com.

it's very easy to find. If you have an exciting application, that you can think of for all these systems or if you have questions, I might be slow to answer, but please do feel free to email me. Yeah. I think it's an exciting space and thank you for, educating people and that, I think that's always extremely, I.

Eric: Thanks so much for coming on. I think this was an excellent episode. Best of luck with espresso, we'll keep following up with espresso because I expect great things out your group. It's very exciting. Cool. Thanks.