EPISODE 350

[INTRODUCTION]

[0:00:01.1] JM: Social networks like Facebook and Twitter facilitate interactions between individuals. Every message I send to you on Facebook goes through Facebook's servers before reaching you. This is known as the client server model. Since the early days of the internet, engineers have always envisioned a peer-to-peer model where I could communicate to you directly without a company or some other kind of middleman brokering that relationship.

André Staltz works on Scuttlebutt, a peer-to-peer system for social graphs, identity, and messaging. Scuttlebutt is used by this group of open source hackers many of whom live off-grid and they don't have constant access to the Internet. In this episode we discuss why someone would want a peer-to-peer off-the grid social network, we discuss how to build one, and the progress that's been made on Scuttlebutt by some of the most talented open-source engineers in the world.

This is a great episode, I enjoyed it a lot and I hope to do more shows about these off-grid group of hackers who I had not heard about before I did this episode.

[SPONSOR MESSAGE]

[0:01:23.4] JM: Do you want the flexibility of a non-relational, key-value store, together with the query capabilities of SQL? Take a look at c-treeACE by FairCom. C-treeACE is a non-relational key-value store that offers ACID transactions complemented by a full SQL engine. C-treeACE offers simultaneous access to the data through non-relational and relational APIs.

Company’s use c-treeACE to process ACID transactions through non-relational APIs for extreme performance while using the SQL APIs to connect third part apps or query the data for reports or business intelligence. C-treeACE is platform and hardware-agnostic and it’s capable of being embedded, deployed on premises, or in the cloud.

FairCom has been around for decades powering applications that most people use daily. Whether you are listening to NRP, shipping a package through UPS, paying for gas at the pump, or swiping your VISA card in Europe, FairCom is powering through your day. Software Engineering Daily listeners can download an evaluation version of c-treeACE for free by going to softwareengineeringdaily.com/faircom.

Thanks to FairCom c-treeACE for being a new sponsor of Software Engineering Daily, and you can go to softwareengineeringdaily.com/faircom to check it out and support the show.

[INTERVIEW]

[0:03:02.4] JM: André Staltz an open source hacker. André welcome to Software Engineering Daily.

[0:03:07.1] AS: Thank you for inviting me, Jeff. Very nice to be here.

[0:03:09.7] JM: Yeah. You came to my attention when I read your blog post which is called *An Off-The-Grid Social Network.* It's about this peer-to-peer system for social graphs and identity and messaging. It's called Scuttlebutt. Explain what Scuttlebutt is.

[0:03:26.6] AS: Scuttlebutt is basically just a peer-to-peer database but the purpose is to implement the social networks. I'm talking really something that really looks like Facebook. Now, it's sort of like entirely peer-to-peer which means that you can use it even in the same LAN. It's not like Bitcoin where you need to be connected to the internet. You can still be like be – It has a sort of mesh effect. There are some types of P2P which are sort of mesh. That's basically in a nutshell. Of course, there's a lot to it.

[0:04:02.5] JM: Yeah, we'll get into that. There have been other attempts at peer-to-peer social networks prior to Scuttlebutt. What had been the shortcomings of those other social networks?

[0:04:16.9] AS: The one that comes to mind right now is Diaspora which got the attention some years ago. If I remember correctly, even Mark Zuckerberg also donated to that project. Diaspora is quite a different idea because it's what we call federated social network. It's like email that you

need to have all these different service that will give out accounts to you. You still needed to register to it with an email and a password.

That means that your account was on one of these Diaspora servers. That's a totally different approach to what Scuttlebutt has because in Scuttlebutt you can't even have an account and a password somewhere else because all of the data is on your computer. One of the nice things even for on boarding as a user experience is that once you open the Scuttlebutt like clients, there's actually many clients, that's it, you're in, because the only thing that you need is a public and private key, so crypto, and that will identify you. That's sort of like your identity.

I also remember a bit from Diaspora, I wasn't one of the early adopters or anything, I just watched it from the sides. I remember they had all sorts of limitation problems and security problems. Scuttlebutt was built from the ground up with a lot of crypto in mind. You could consider it closer to Bitcoin in the sense of how serious it takes crypto.

For instance, your newsfeed is not just like some data in a database, but your newsfeed is like – Your messages are what we call sig-chained, signature chained. That means that each message will sort of sign the contents of the previous message. Everything is chained, and that is sort of to prevent tampering so other people can't pretend that you said something or skipped out some parts of your newsfeed.

[0:06:24.8] JM: Yeah, we'll get into the elements of the log structure that is – It's fundamentally a log structure. It's a pend only log structure. Let's start with the network aspect in the higher-level aspects, and we'll get into the technical stuff. You write in your post that "with social networks like Facebook or Twitter, the network connections centralize with their servers" and this is what you're talking about here. In Scuttlebutt, the network architecture looks like the social architecture. Why is this important

[0:06:59.6] AS: This is important because people don't tend to understand how network connections are. I actually ask some friends what do they think – What happens when they use Facebook. They just actually thought that it goes through like radio waves just like phone calls and SMS happens. They thought that you're using Facebook client that would directly connect

to your friends, Facebook client through radio waves. That's people's intuition if they don't know so much about technology.

That means that once that intuition is broken, they feel very surprised or they feel this is wrong. I have some sort of experience with that because in Brazil, where I come from, people use a lot all these services and sometimes the government decides to shut down one of the service. They shut down YouTube for a while for two days or something like that they shutdown also WhatApp in the same fashion. Just some judge decides that if WhatsApp company and Facebook company don't comply, we're going to just shut down this thing in the whole place.

This is so fundamental because like a lot of businesses in Brazil are run on WhatsApp, not to mention personal communications and that kind of stuff. People get surprised, like, "Yo, what is this? I thought that I could just communicate directly with my friends and people that I need to do business with," but actually there was some kind of thing in the middle. I think that that surprise is bad because most people don't understand how technology works and our natural assumption is that that the network connections match the sort of interaction with people.

One of the ways that Scuttlebutt does that is you can set up a server, but that server typically represents a community. You can get an invite for that server. Let's say we have, let's say, a football club or a baseball club and you want to invite everybody related to that club, so you set up a server and you send out invitations. That will really match the community and the people related. The idea of Scuttlebutt is that you would have many of these for each community or circle friends or something like that and then you're able to connect to everyone that you need.

[0:09:27.0] JM: Yeah. Scuttlebutt was originally made by Dominic Tarr, and he's a developer who lives on a sailboat. This guy is really interesting. Tell us more about Dominic Tarr.

[0:09:39.7] AS: I wish I had met him in the person. I met some other New Zealand people. He's from New Zealand, and Dominic created a ton of NPM modules and he started experimenting with some ideas with peer-to-peer. Originally, he built Scuttlebutt. Just the name Scuttlebutt, that was some kind of database that reflects what we have today. Then he noticed some security problems and he made secure Scuttlebutt.

What we're talking about right now is actually secure Scuttlebutt and not just Scuttlebutt, but we just shorten it and we called it Scuttlebutt because actually the original Scuttlebutt is just sort of legacy now. I don't really know the motivations that Dominic has because I haven't met him, but I know that he and others in New Zealand, it's not just him, are very sort of oriented on avoiding capitalism and they are looking into cooperatives and new models of economy and society and the kind of stuff. They're really into these kind of things. I would assume that that motivated some of their work here with secure Scuttlebutt.

[0:10:48.1] JM: Yeah, after Trump won the election, I was looking into like where the places where you can go if some sort of apocalypse breaks out, and I think New Zealand is number one. New Zealand is the number one place for preppers and people who just want to live, because the impression I get looking at Dominic is like this guy is – He's really independent. He wants to do self-reliant. Self-reliant is possible. He wants to be off the grid. It looks like he's just a full-stack engineer. When I say full-stack, I mean he can build his own self-navigating sailboat, which is just incredible.

[0:11:24.3] AS: That's just ridiculous, yeah.

[0:11:27.6] JM: Yeah. Do you know anything more about him? How does he make money? How does he – Does he just have a nice –

[0:11:32.2] AS: He does consulting as far I understood. Very similar profile of person is also substack, or James Halliday who made Browserify and 100 other NPM modules. Also, James lives on Hawaii and he tries to also build everything from scratch, even his own house basically.

Yeah, these people have a different philosophy. You mention about apocalypse, and I think that's a very interesting thing because when you think about how was the internet built, the question that motivated the internet, basically in DARPA, was what happens if we get a nuclear bomb on this kind of data center, on the East Coast, or on the West Coast? Then they thought we need to sort of decentralize these kind of stuff, right?

I think we're starting to wake up to that scenario again, not like an actual apocalypse but in the sense that the trend on the Internet has been centralized stuff; Google, Facebook ,Twitter and

whatnot, and now were thinking like, “Okay. These things can be vulnerable to abuse of power. How do we decentralize again? We’re seeing the second trend. I would say that the internet – Sometimes on the Scuttlebutt network we talk about the old internet, which is basically the web dominated by I can and Google and all these kind of companies That's not the actual ideological internet that we had from 20 years or 10 years ago or something like that.

[0:13:10.2] JM: Yeah, this is the same conversation I had with – Who is it? Controversial guy. A Moldbug. The guy who works on Urbit. He’s basically same thing. Lamenting the fact that we have this big dream of going to a peer-to-peer architecture for the internet and we've ended up with client-server centralization not even from the standpoint of like, “Oh! These nasty corporations and their centralization,” but just like is less fault-tolerant.

[0:13:37.4] AS: Yeah, it’s less fault-tolerant, and we can see that pretty clearly, not so much or yet in America, but like in Brazil. People can shut down WhatsApp and that's really a bummer. It really affects people's lives in a very strong way. Yeah, I would say so.

[0:13:57.4] JM: We’ll get back to Scuttlebutt. These people, like substack, are they driven by this wanting to be prepared for some kind of societal collapse or is it more like just raw self-reliance? Is it just some combination of those things?

[0:14:14.0] AS: I won't speak on behalf of their motivations, but I as far as I’ve understood what is evoking from them is basically a realization that some structures we have can't be trusted. Also, a realization that, “Hey, I don't actually need them. I can actually live on my own.” That's totally possible. That type of lifestyle where like, “Sure, I could live under the system, but I don't need you. Thank you very much. I'll do my own stuff.” That type of behavior.

[SPONSOR MESSAGE]

[0:14:57.4] JM: Dice.com will help you accelerate your tech career. Whether you’re actively looking for a job or need insights to grow in your current role, Dice has the resources that you need. Dice’s mobile app is the fastest and easiest way to get ahead. Search thousands of jobs from top companies. Discover your market value based on your unique skill set. Uncover new

opportunities with Dice's new career-pathing tool, which can give you insights about the best types of roles to transition to.

Dice will even suggest the new skills that you'll need to make the move. Manage your tech career and download the Dice Careers App on Android or iOS today. To check out the Dice website and support Software Engineering Daily, go to dice.com/sedaily. You can find information about the Dice Careers App on dice.com/sedaily and you'll support Software Engineering Daily.

Thanks to Dice for being a loyal sponsor of Software Engineering Daily. If you want to find out more about Dice Careers, go to dice.com/sedaily.

[INTERVIEW CONTINUED]

[0:16:17.9] JM: It's appealing. It's appealing even for me, somebody who is very reliant on the system. There is this occasional sense of anxiety that if this system blows up then I don't know what I'm going to do. You need to figure things out from scratch.

Anyway, let's get back to Scuttlebutt. Let's take a top-down approach. Can you describe the UX for the typical user who is getting started with Scuttlebutt for the first time.

[0:16:46.2] AS: The typical approaches is you go to our website, which is by the way centralized, but you need to start somewhere. That's Scuttlebutt NZ, and there we have a ton of instructions and then you're basically going to install a desktop app. We have many, but the most common one is called patchwork. That's a client for Scuttlebutt. Then you download that and you're basically on the network. Now, that means that you're connected to no one in the beginning. When you open the desktop application and you haven't done anything else than you're going to see emptiness.

Now, there's a couple of ways of getting data and that means that one of the ways is that you connect to some pub, that's what we call the server. You need to get an invite from someone. Also, on Scuttlebutt NZ, we have some links there to some pubs that you can sort of like join with a free for everyone invite. The other ways basically being on the same LAN, local area

network as someone else, and then you're basically already seen each other through that client app and you can just discuss.

Sometimes, let's say, if someone is on Scuttlebutt already or let's say that they have a lot of data. There's no sort of like global idea of what is Scuttlebutt. Just like there is no global idea of – There is no global registry of all is the one-to- one physical conversations that are happening right, because that's the idea of a mesh.

Let's say that some person has that some person has a lot of data on theirs Scuttlebutt client, you can just connect to the same Wi-Fi as them and then your client is going to start downloading almost everything that that person has and they are basically on that network because then there's going to be some metadata on who else is on that network and other kinds of pubs. You start discovering that kind of thing. You'd basically don't need to make an account because once you open the client app it will generate the public and private key, and that's your identity. Your public-key, you pass that as if it'd be your username.

Now, there can't be global usernames because someone in China could select the username John and someone in America could select same username John, and because it's a mesh, they can't really just say that this – Just like you can't have unique names for people on the planet. Same problem.

[0:19:17.4] JM: As you mentioned, Scuttlebutt uses this protocol secure Scuttlebutt, SSB, and this can spread information even in the absence of typical internet service providers. Give more description for how SSB works.

[0:19:32.8] AS: Secure Scuttlebutt is the database. Now, everything is based on the database. There is a protocol for network called secret handshake under secure Scuttlebutt which basically does key exchange. There's a white paper written by Dominic explaining how secret handshake works. That's just a way of two entities that have their public keys. They're just exchanging information as far as I understand.

The database only has newsfeed. It's a pretty simple system because it's kind of like Twitter. You have accounts, and the accounts can be bots, the accounts can be real people. Basically,

the only thing you have there is just one newsfeed and so there's not any kind of other place where data is stored because every sort of metadata can be also a message. Such as I am following John Smith. That will be a message on my newsfeed. Also, I liked what John Smith just said. The like is also a message on my newsfeed. Everything is recorded on that as if it would be my diary, sort of everything that I do.

Even things like my description of who I am. That's just a message on my newsfeed. There are different ways on how you can share that diary or log with other computers. One of them is LAN. When you're in the same local area network, it's going to do multicast in this. That means that you're basically broadcasting your newsfeed to whoever is on that network. That means that other peers are going see that and sink their knowledge of what is your newsfeed.

Your newsfeed is also immutable, so you can't actually delete any message, and that's how also things are sig-chained, so chained with signatures. It may sound like a downside that you can't delete anything, but that's actually pretty good because you can't actually delete things in real life. Let's say, things that you. If you say something on a stage to a lot of people, you can't just tell them like, “Hey, forget that I said this. Permanently and reliably forget that I said this.”

People have their own memories and just can't do that. It's the same case with Scuttlebutt that you can tell people like, “Hey, try to delete what I just said, but you can't guarantee that people will delete that.”

I think that's quite an honest approach, because on the internet, you can actually delete content. There's still, let's say, the internet archive and you can take screenshots of whatever people say. It's not really reliable to just like try to delete data. I think Scuttlebutt doesn't have like a downside. I think is just being honest about the nature of data.

[0:22:36.3] JM: The append only ideas are quite popular in software architecture today. I think this is just one side effect of storage getting so cheap, is people – They’re saying, “Okay, what can we do with infinite storage?” Well, we can shift our entire architecture to having this single event store where you have an append only log of everything that has happened, every change to the state of our application, whether you’re talking about Netflix or Facebook or Twitter. You can put everything chronologically such that if an error occurs or if an outage occurs and you

lose the database, as long as you have the event store, you can replay all the events in the event store in order to restore everything. That's sort of what Scuttlebutt is doing.

I think this architecture has been studied a lot by blockchain people, because the blockchain people are trying to create an immutable append only transaction log for financing and it's – Well, sorry. Not all blockchains are for finance, but Bitcoin obviously. I know that blockchains often use a data structure called a Merkle tree. I don't remember exactly how that works. Is there a similar approach in Scuttlebutt?

[0:23:55.8] AS: That's a good question. There might be. Just explaining a little bit what al Merkle tree is. I'm not an expert on these things, but as far as I understand, it's a hash of hashes. You can have a lot of data and then you just hash parts, and that's I think how BitTorrent and torrent in general that you can create a quite small hash out of like a lot of data, that type of stuff.

[0:24:24.2] JM: It's a compression tool as well.

[0:24:26.8] AS: I wouldn't comment on that. I think, in general, hash is just like – Yeah, it's not compressed because can't like get that data back just from the hash. As far as I understand, I dig into Scuttlebutt quite a lot and I didn't see something similar to a Merkle tree.

You mentioned events and the whole history of events, and I think that's, in general, an approach called event sourcing. We use that almost every day with git. git is also append only kind of. One of the interesting things is that people already build a git layer on top of Scuttlebutt because they are similar type of databases, so they just sort of wrote that adapter.

[0:25:17.1] JM: Right. I think that's profound. Even GitHub is essentially a social networking and identity is core to what GitHub is. It's interesting that at the core of Scuttlebutt is this idea that you have an identity which is manifested in your crypto keys and then your information, which is just in this law, the append only log, and you can build essentially any popular internet application off of those features alone.

[0:25:48.7] AS: Yeah. There like there are ton of ideas — Such as the SoundCloud as well. They already built also a SoundCloud clone or inspired service on top of Scuttlebutt, it's called Ferment. For the data itself, let's say the songs or the music that's stored I think in the BitTorrent of something like that. SoundCloud is also a social network because can follow people and that kind of stuff. I think there's ton of other stuff, like Instagram. It's basically just a Facebook without text. It's just images. I don't know. There's ton of stuff. Maybe not something like Snapchat, because it basically deletes data constantly. I think that's okay because it's an honest approach. You can definitely just record your mobile screen and then you won't — You will have that data forever if you want something on Snapchat. There is definitely a ton of things that could be built, and the git layer right now works really well and it's actually comparable to GitHub because, let's say, when you push a repo to the git-ssb layer — SSB stands for secure-Scuttlebutt.

Then you're basically pushing a bunch of messages to your personal log, so your newsfeed log, and those messages are just git commits, like the content is that. One nice thing is that the push is immediate because it's just getting from your [REPL] and inputting that to your local Scuttlebutt database, so you don't need to wait for a considerable amount of time.

Also, if you're in the same network as let's say a bunch of coworkers, they will be able to get that with very high-speed because they're on the same network, and then you can also run git-ssb web, and that runs a local server, like some HTTP, HTML server, and then you can open that on your browser and you see something that looks like GitHub. You have there the readme. You have issues. You have pool requests. When you open an issue that will create a message on your Scuttlebutt log, then — It's just works. It's very interesting approach.

[0:28:09.7] JM: If I get on the same Wi-Fi somebody as somebody else with Scuttlebutt, then Scuttlebutt can just spread the information that I have seen with the information that the other person has seen and resolve the diff. Can you describe what happens in a sharing event?

[0:28:34.5] AS: A sharing event. What do you mean with sharing event?

[0:28:36.0] JM: By sharing event, I mean — I shouldn't have used that terminology. Let's say I've gone off and I've Scuttlebutted with a couple other people and then my friend has gone off and Scuttlebutted with another couple of people and then me and my friend connect to the

same Wi-Fi and our logs are going to merge so that the information spread sort of like the gossip protocol sort of.

Can you describe what happens there? Does it just get appended on? What's the merger —

[0:29:08.7] AS: Yeah, it's appended because you only see, "What am I missing?" And then you add that. The diff is always pluses. You don't have minuses because we don't have delete. Now, you may ask, "Do I want data from my friend even like his friends and their friends?"

One of the important things is that there's always choice, because everything is in the client. There's no server which decides stuff. Everything is decided on your client, so it's potentially configurable that you can say, "I will only want to get feeds from this friend, or friends of that specific friend, or friends of all of my friends, or two hops after that." You can configure all these things.

Of course, you don't need to necessarily get all of the data that he has. Now, of course we have default there. I think it's two hops of friends of friends and you end up getting a lot of data sometimes from people that you don't immediately recognize. As I said, everything is on the client so you can also say, "
I don't want data from these type of people," so you can just block them. Yeah, you can configure all these things.

[0:30:22.7] JM: It sounds like you don't have the same problem that you get with financial blockchain, because with the financial blockchain, you need everybody to have a copy of the entire transaction log unless you want to build some sort of centralization on top of it which a lot of people are doing, and that's totally fine. It's just that if that were the case, then you would have needed to gather everybody's information and that would quickly not scale, because if you want to keep everything client-side, you don't want to have this giant record that includes the entire social network.

[0:30:58.0] AS: Finance is fundamentally very different because you can transact money with anyone. Of course, there needs to be this sort of a global knowledge of what is money. In the

Scuttlebutt docs, we called this global singleton, just like a single, let's say, shared module that everyone references.

Now, in Scuttlebutt, things are just mesh, so you don't need to have the information about everything, because it's really sort of based on relationships. It's much simpler to understand than all kinds of other things because the data is your newsfeeds and your friends' newsfeeds, and that's pretty much it. Of course, you can make more friends in that type of stuff.

[SPONSOR MESSAGE]

[0:31:51.4] JM: For more than 30 years, DNS has been one of the fundamental protocols of the internet. Yet, despite its accepted importance, it has never quite gotten the due that it deserves. Today's dynamic applications, hybrid clouds and volatile internet, demand that you rethink the strategic value and importance of your DNS choices.

Oracle Dyn provides DNS that is as dynamic and intelligent as your applications. Dyn DNS gets your users to the right cloud service, the right CDN, or the right datacenter using intelligent response to steer traffic based on business policies as well as real time internet conditions, like the security and the performance of the network path.

Dyn maps all internet pathways every 24 seconds via more than 500 million traceroutes. This is the equivalent of seven light years of distance, or 1.7 billion times around the circumference of the earth. With over 10 years of experience supporting the likes of Netflix, Twitter, Zappos, Etsy, and Salesforce, Dyn can scale to meet the demand of the largest web applications.

Get started with a free 30-day trial for your application by going to dyn.com/sedaily. After the free trial, Dyn's developer plans start at just $7 a month for world-class DNS. Rethink DNS, go to dyn.com/sedaily to learn more and get your free trial of Dyn DNS.

[INTERVIEW]

[0:33:50.5] JM: With blockchain, Bitcoin, for example. The way that Bitcoin is moving is, assuming Bitcoin works out, and it looks like it's going to work out at least from my point of view.

[0:34:02.9] AS: It's been working.

[0:34:03.0] JM: It's been working. People like Blockstream who want to build these sidechains and lightning networks on top of it and what these things allow is just faster transactions because you get these trust networks, so you don't have to necessarily have the entire block chain be copied to everybody's node. You get these centralized agencies of trust and brokerage.

You could imagine people building nodes of centralization on Scuttlebutt. For example, if I want to have an in-browser version of Scuttlebutt and I just want to hit some server, if I do want to have client-server. That's the thing — That's what's cool about Scuttlebutt is you could imagine client-server level Facebook on top of it and it will be totally fine. There's nothing wrong with that and it wouldn't be mutually exclusive with the peer-to-peer beauty of Scuttlebutt.

[0:35:00.1] AS: Yeah, and actually just to mention quickly. There are some clients. I think one of them is called patchfoo that do exactly that. It's running on the server and it's a wave of people to get it running in the browser instead of downloading a separate client so they could also use it on mobile or other devices. That's totally fine. It's something you can do, as you said.

[0:35:21.3] JM: Because it would be cool — I think Scuttlebutt is a great idea. I think it's really cool and it would also be cool if — The disadvantages — Under the current situation, my mom is never going to use Scuttlebutt because she would have to download it and set up her —

[0:35:35.7] AS: I wouldn't say never. I would say in the current situation, because it can reach a point which is quite user-friendly especially if we think of mobile. If we put Scuttlebutt on mobile, then you just need to download an app and that's it. Usually, you need to download an app and make a user account, which is also the annoying part. People don't like making new accounts and setting up passwords and confirming in their email. It's always a hassle. With Scuttlebutt, it would literally be just download, open, and that's it.

[0:36:09.2] JM: That's true, and it's compelling that's we are moving — If you look at Snapchat, it's a leading indicator. People are getting more interested in privacy technology, and I mean

you're seeing journalists — Every journalist is adopting Signal or telegram or something encrypted that it allows them to collect crypted information from sources. There is a public movement away from centralization. Who knows how far it will go, but it's there at least.

[0:36:37.7] AS: Yeah. I think people are starting to wake up about the problems that centralization can create.

[0:36:44.9] JM: Can you talk more about the open-source community of Scuttlebutt? How do people work on different stuff and what are the things that — What are tasks and the features that people are working on right now?

[0:36:57.9] AS: The development style is like — How to say it? It's like sort of group-driven. It's not so much like a benevolent dictator for light-driven. Even if Dominic has been the original creator, we don't have this approach where Dominic dictates everything, that type of stuff. We actually meet every Wednesday. I think we even have like a meeting this week where we meet over voice chat and we go through like what's the latest news, what is each one working on, and it's totally volunteer based. Of course, Dominic normally works on the actual protocol and the reliability of these kind of stuff. Then there's people working on the client apps, like patchwork and etc. It's just volunteer-driven.

The things that we're having right now are sort of like Domenic is making the network more reliable in the sort of network transfers faster. One of the things he's working on is called epidemic broadcast trees. Don't ask me what that means. It's pretty deep. He found some stuff from some papers that helps.

Then the patchwork development pause for a while because the main maintainer is just traveling for two weeks or something like, but there's plenty of PRs coming for that because we got a lot of people into the network and they just wanted to improve stuff. I think two days ago we had like an update to the patchwork client that made things look nicer.

I'm working on getting Scuttlebutt working on mobile and that's quite a challenge because I just basically getting the Node.js app and trying to put it with React Native. Those two things were

not meant to be, and also React Native, to some extent, is not meant to be. It's kind of a hack. So it's a hack on top of a hack.

[0:38:55.2] JM: That's right.

[0:38:57.0] AS: It's sort of like slow working. It's incredibly painful to get it working, but I think it's probably less efforts than rebuilding all of the protocol for iOS and for android. I think things are possible we just need sort of like UDP and TCP and LevelDB is the main sort of database part where all the newsfeeds are, and some crypto stuff.

All of these things exist already for native, like for android and iOS. We just need to find how to get all these things talking with each other in the proper way.

[0:39:37.3] JM: We just had a show about React Native and the Expo Team. Can you tell me more about what are the challenges that you're encountering trying to — I guess you answered my next question, which is going to be; what language is this written in. I guess it's all in JavaScript, right?

[0:39:55.2] AS: Yeah, it's all JavaScript. Of course, we use a LevelDB and Sodium. LevelDB is for the database and Sodium is the crypto, and those two are like C-based, so they're native. They're the native modules in Node. Node sometimes have these native stuff.

[0:40:15.9] JM: As you're migrating this to React Native, describe how that looks or what are the challenges that you're encountering.

[0:40:25.7] AS: There is a thread where — Thread and in Scuttlebutt network which is also accessible through HTML. There's a way where I describe all of my sort of hurdles with that. Then, basically, I'm trying to first — I was trying to get the TCP layer working, so there's basically a Polyfill for React Native. I think it's called React Native TCP, where it basically just imitates the TCP sockets that Node has. That mostly works. I noticed that there was a problem with it, that when you create the sort of constructor — You create the TCP — The constructor. It didn't support giving opts and things like that where not all of the parts are polished for all of these Polyfills and you'd find bugs here, bugs there.

One of the biggest challenges that React Native doesn't support synchronous APIs, so if you do something file system read directory synchronously, they just can't support that in React Native, I think it's because of how the threats work that you just can't expect things to be there immediately, so there's some waiting involved always.

It would be good if everything would be like asynchronous callback base so that we can do everything. I need to map out all those parts that use synchronous APIs and try to re-architect things to use asynchronous. I raised this issue also with sort of the protocol people, like Dominic, and they acknowledge that, "Yeah, we should try to find a way to only use a-sync." There's a ton of stuff.

I got to some point where you opened the sort of very raw app and it generates your private and public key, that's good. Stores them, that's good also. It's a milestone. Then I got some sort of network connections working with my computer. So the computer would send some message to my phone, through TCP, on LAN, and vice versa. I'm trying to get the newsfeeds to sync. That's my next milestone, to actually get newsfeed from A to B and backwards.

[0:42:43.9] JM: I always admire people who work on projects like this because this is not the kind of thing where you can just search stack overflow for —

[0:42:52.2] AS: No way.

[0:42:55.6] JM: How to do this. What do people talk about on Scuttlebutt? Because I imagine, today, it's just like a bunch of developers, basically open to attackers and maybe some privacy advocates who are maybe not hackers as much but are just interested in this technology. Social networks always — The early days are so fascinating. I was an early user or a Quora, and Quora was so interesting in the early days because the network was seeded with just venture capitalists and engineers and entrepreneurs. It was a very fascinating web of information.

Back in the day, I was a poker player, and I was on this poker foreman. The social networks really — The meme's and the utilities that develop within a social network are completely reflective of the people in that social network. How does that manifest on Scuttlebutt?

[0:43:50.3] AS: You are correct to imagine that most people there are developers. There are some also non-developers, friends of friends, and that the connections. It's interesting how the blog post that I wrote evoked this sort of lifestyle of off-grid, and a lot of people interested in this topic of living off-grid started joining and there was a lot discussions around that.

Typically, we had before that discussions on open-source, on decentralization in general, living in a decentralized fashion with off-grid and that type of stuff. Nowadays, we get a lot of questions as well how the thing works. I think it's a really good thing that we target developers first, because the platform needs to improve in many areas before we can start putting all kinds of people. I think it's good that we attract developers in the beginning. For instance, I myself was attracted to this network and I'm helping with the mobile part and people who find other types of lacking things will also probably feel like helping in those areas that lack.

[0:45:06.2] JM: What about the conversations that actually take place or people tweeting about what they had for breakfast or what their self-driving sailboat is doing today. What exactly are they?

[0:45:16.3] AS: Sometimes those two things that you just mention, yes, Dominic has written about how his sailboat is going to face some kind of cyclone. He also said, "Today, I fish the squid and I ate it for breakfast," and he had a picture of the squid there. It's like white squid and he said it tasted like just sponge.

Also, substack, or James Halliday, is often posting about how he's building his own home with his partner and it's pretty interesting. If you were interested in what I wrote in the blog post, that's just like the tip of the iceberg. They have so much stuff there going on.

I think, in general, the discussions are like mostly development, also philosophy and all kinds of economy and things like that. I haven't had the need to block anyone yet, even though I'm getting data from all kinds of people who have no idea who they are. I haven't had the need to block anyone. There're pretty good vibes there going on.

[0:46:26.9] JM: If you reach a certain scale, there is going to be the moderation issue. I wonder how you could solve moderation.

[0:46:34.3] JM: Oh, that's an interesting question. I don't think we have any — We don't have an equivalent notion of moderation as you have in, let's say, Facebook or Twitter. Of course, you can block people. If you see someone who you just don't want to read the things they say, you can just block them.

Now, what cannot be done is that two people in Japan are talking nasty stuff and you want to forbid them to talk those stuff. You just can't do that. Just like you can forbid people from saying nasty stuff with two walkie-talkies. Even that the creator of walkie-talkie has no power in preventing people from using it however they want. It's the same idea here. We really cannot control what people do with it. It's like technically impossible.

[0:47:24.6] JM: How much of tech needs — This idea is basically like you're building the core of a protocol that you can build other applications atop off that's peer-to-peer. Now, you're building the client for iPhone. Are you starting to wonder how much other decentralized technology — Do we need some decentralized hardware? Do we need open-source cellphones to truly get to the dreams of pear-to-pear that people want? It seems like we're just moving towards a world where some people are working peer-to-peer stuff at a pace that is as cutting-edge as the people working on centralized stuff.

[0:48:11.7] AS: One of the things that peer-to-peer network like Scuttlebutt introduces is the need for storage, because all of that data is with you all the time. Typically, you don't have that with Facebook or something else, You download that always.

The amount of data can get quite a lot. Currently, my client stores, I think, around 2 GB. Most of it is just images. The actual logs, like the newsfeeds, they are around a hundred or something megabytes. I mean that's okay. That's manageable, because you can just delete pictures. Those are sort of like not essential for the, let's say, signature chaining part, and then that's like mostly the data. 100 megabytes, or 200, or 300 is just not a problem even for mobile phone's.

At some point, let's say, you're using it for 10 years. Maybe it starts becoming a problem. I think sometimes demand drives supply just like we have for KTVs nowadays. We didn't have those 20 years ago, so people demanding better image just drove the supply for better TVs. I think, maybe in 20 years from now, if peer-to-peer social networks are thing and they demand lot of storage, then your phones are just going to have 3 terabytes of storage or something like that.

Does it touch the question that you were —

[0:49:39.4] JM: Yeah, yeah. Absolutely. Are there scenarios in Scuttlebutt that are similar to the idea of where you — In Bitcoin, you can lose your wallet, the credentials to your wallet, and then you just lose your ability to access your account. I presume the same problem exists in Scuttlebutt.

[0:49:57.6] AS: Yeah, the same problem exists and there's nothing that we can do about it other than you being very smart about keeping your data. There are some approaches to this in Bitcoin that work quite well, let's say the recovery words. You can have 24 words that will generate your key. You could just use that and store it like as a paper somewhere and then you'll be fine.

The other approach is that you sort of leave the key management to someone else. You just store it in some kind of bank and you login as you do today. You can do that with Bitcoin. Why couldn't you do that with Scuttlebutt?

I think what's important is that people have the choice. I can store my key in some kind of bank that handles it for me or I can store myself and all kinds of different ways, but we just essentially cannot. It's impossible. No one has the authority over all of the council. We cannot just recover things for people if they lose it.

[0:50:56.1] JM: How is the growth trajectory looking like for Scuttlebutt?

[0:50:59.1] AS: It's a bit hard to estimate what's going to happen, but I myself, I'm investing a lot of time into this and I think now that the activity gone up, Dominic and others are also very excited about putting effort into this. We'll see. We'll see how people react to it. I have a lot of

plans on how to make it mainstream. That blog I released was just like one piece of the whole plan, and I definitely plan to put a lot of the different content pointing to Scuttlebutt. I really want to help this go mainstream probably not to the extent that it replaces Facebook, because that's like a humongous challenge. Just having enough people and enough resources so that people know that there's an alternative, right?

“If I want to leave Facebook, where do I go?” Right now, most people would say, “I have no idea.” Maybe in the future people would have some idea of what to do if they want to leave Facebook.

[0:52:03.5] JM: It seems like the way we’re going in we’ll have as many social networks as we do television channels, because it’s becoming so easy to build a social network and we have so much unexplored territory in terms of social networks. There’s a bunch of these things; musically, Instagram Quora, GitHub, and it’s just going to keep growing. There’s going to be more and more of these things.

When you say you want to invest a lot and you want to make it into mainstream, are you potentially thinking about starting a business around Scuttlebutt eventually?

[0:52:38.0] AS: Not unnecessarily, but I don't exclude from that possibility either. I would actually disagree with what you just said because — Respectfully disagree. You know what I mean, because we used to have a lot of social networks when Facebook wasn't the largest thing ever. We had, let's say, MySpace, we had [inaudible 0:52:57.2], we had all kinds of stuff. I think the tendency nowadays is that Facebook, because it's the biggest, people have the assumption that everyone is on Facebook. It becomes something huge just makes it become even bigger.

Now, with all of the Snapchat features being replicated in Facebook, I would say probably Snapchat won't have a bright future.

[0:53:23.9] JM: My belief around Facebook, and this is what Peter Teal, the first investor in Facebook. He said he thought was that the core value of Facebook is that it enforced the real identity. It’s like it makes you put your real identity. You can only have one account. It’s your only

thing. On Twitter, you can have multiple different accounts. That can be replicated, right? Google has it — Google on Facebook, and Google developed independently, but you had your real identity on Google, sort of. I guess I have multiple Google accounts, one is for work, one is for non-work. Just the notion that you put some sort of stake in the ground that this is your real identity and you perhaps provide some sort of verification around that identity. I think that's really the novel approach that Facebook took, and obviously that compounds interest because you as you're building up more identity features on Facebook it becomes more like your true identity, but that's not to say that you can't build a separate identity platform.

[0:54:25.9] AS: Yeah. I don't know what to comment.

[0:54:27.9] JM: Okay, fair enough. I don't mean to argue with you. I'm just saying I think Facebook is quite dominant right now, but I think it could be rivaled in the future. I certainly want way more social networks, and I can think of plenty of social network ideas where I would not want Facebook integrated with it.

[0:54:47.5] AS: Right now we have that effect for IM chats. Some people are on Signal. Some people are on WhatsApp, and Facebook Messenger and all kinds of different ways. It can be a pro and can be also a con. A lot of people feel annoyed that there's different ways of contacting people. I don't know. I wouldn't say that what Facebook is doing right now is good because then we're basically giving them the monopoly over communications.

[0:55:20.0] JM: I do feel like there's a growing allergy towards Facebook. How do you feel about Facebook these days?

[0:55:26.2] AS: I feel very negative about it because it's just such a monopoly — It's basically like digitalyzing your relationship. We're just digitalizing all kinds of stuff nowadays, and Facebook happens to be digitalyzing your relationships and just making profit out of it. I think the way that it does that is by presenting itself as a necessary evil, because even publishers are not happy about Facebook, but they just have to be there because everybody is there, and the same effect for people who are using it as users. They just have to be there because people are there.

Of course, Facebook tries to create an excellent image of themselves and how they want to connect the world and etc., and how they want to keep everybody comfortable, all the users and that kind of stuff. Essentially, we ended up with no choice. People are just kind of on Facebook because it's the big necessary thing.

The other day, I wanted to go to this bar in Stockholm with friends and I wanted to use the Wi-Fi but you needed to connect to Facebook and like their page. It's like there was no choice. Once you connected to their Wi-Fi they would open their Facebook page and you have to like that otherwise you would just not get access to their Wi-Fi. Guess what? My computer blocks every link from Facebook. I just did that, and so I just didn't have internet.

We are getting to that stage where Facebook will just be absolutely necessary and people are going to have the assumption in their brains that it's just necessary, and then we may have no choice if we don't build an alternative. That's why, really, I want to do something about Scuttlebutt so I have that choice.

[0:57:21.8] JM: Yeah, it's a noble pursuit and I hope Dominic or substack ends up hearing this episode. I'd love to have them on the show and talk more about this growing — Not just the growing allergy to Facebook, but just the peer-to-peer efforts. The peer-to-peer crowd and this off the grid people are so interesting to me because these are people who are — They're so technically able and yet they choose to opt out of the —

[0:57:52.7] AS: Technical world, basically.

[0:57:53.6] JM: Technical world. They choose to opt out of the world that could give them massive, massive amounts of money and power.

[0:58:00.1] AS: Actually, a good terminology that is going around in Scuttlebutt to describe these people are solar punks. They're people who are kind of related to cyberpunks that they have lowlife, high-tech, but they have a very optimistic view of the world and how we can use solar power to just be independent and just get our resources from the ultimate source of resource, which is the sun, that kind of stuff. You should deftly invite them because I don't know

if I represented them correctly here. Of course, I don't know them in person, so maybe they have different views of the world.

[0:58:37.2] JM: That's beautiful. I would love to have some of the people on.

Okay. André, thanks for coming on and talking about Scuttlebutt. I really like your article. It was really great overview. Really concise yet very well developed. I'll put it in the show notes. I think the listeners will like it too. Thanks again, André.

[0:58:54.7] AS: Thank you. Nice talking to you.

[END OF INTERVIEW]

[0:58:59.8] JM: Thanks to Symphono for sponsoring Software Engineering Daily. Symphono is a custom engineering shop where senior engineers tackle big tech challenges while learning from each other. Check it out at symphono.com/sedaily. That's symphono.com/sedaily.

Thanks again to Symphono for being a sponsor of Software Engineering Daily for almost a year now. Your continued support allows us to deliver this content to the listeners on a regular basis.

[END]