# AllERCDevs (EU/US time friendly session)

**Start time:** Jun 01,2023 10:59:41 PM  
**Duration:** 53mins 48s  
**Airgram link:** <https://app.airgram.io/7065496599555735553/my-meetings/b8f31df0008c11ee91af0a2cbd030b4d>

# Notepad

# Transcript

Victor Zhou (@xinbenlv) (01:27)  
That's way. Another minute. And then we can start. Alrighty. Welcome to the second or ERCF. Thanks for joining. And this is Victor. We have not started recording yet, but just to want to say thank you for speakers to come, and also for everyone who's interested to come.

Victor Zhou (@xinbenlv) (02:44)  
This meeting is going to be recorded. And it's going to be uploaded to the ethereum cad header thanks to ethereum cadders organization for helping. And if you don't mind, I can. Start recording. This meeting is being recorded.

Victor Zhou (@xinbenlv) (03:19)  
Hi, Everyone. My name is Victor. Welcome. To all erc ref. It's a regular meeting, hopefully to bring erc authors, the App developers and everyone who's interested in building this ecosystem together. To.

Victor Zhou (@xinbenlv) (03:39)  
Increase collaboration. To help move the smart contract forward. And to advance open, standardization. This is the second one. And we're trying to do this in the rotating approach. Which in every two weeks we will have a meeting.

Victor Zhou (@xinbenlv) (03:58)  
And which being one of. It. To be Asian US Time friendly and the other one. To be. European US Time friendly and we are very happy to have a very rich agenda today. By the way. This event. This regular meeting is highly inspired by.

Victor Zhou (@xinbenlv) (04:24)  
All core dev, which is I believe happening right now. And also the old wallet dev, which. Is initiated and founded by Sam, who is also with us today. And the first part of it of today's agenda would be a spotlight.

Victor Zhou (@xinbenlv) (04:41)  
Speech which Sam is going to give and I invited him to talk about. Wallet test framework spells as wtf. I think it's interesting name. I also think it's a very interesting approach where there's so many different wallets and so many different.

Victor Zhou (@xinbenlv) (04:59)  
Implementations and smart contract depending on wallets. So having an effort coming in for wallet, both the small controversy and non smart contraversion is interesting. So with that, I want to hand over to Sam. And thank you, Sam, for coming to give us this talk.

Sam (05:21)  
Hey and you can.

Victor Zhou (@xinbenlv) (05:23)  
Shoe free to turn on or just keep your camera shot. It's okay. We're very casual. Perfect.

Sam (05:31)  
I cannot share my screen. Unfortunately.

Victor Zhou (@xinbenlv) (05:37)  
Let me make sure you can share the screen. Can you try now.

Sam (05:43)  
There we go. Perfect. Thank you. All right. So here to talk about Wallet Test framework. It's a little thing that we've been building with some support from the EF. It's a collection of tests for ethereum wallets and the tools to support them.

Sam (06:03)  
So this will probably look familiar to any of the old school web developers here if there are any. If not the acid. Three tests were a collection of tests. That were put together back in the day to ensure cross browser compatibility.

Sam (06:22)  
And the btf is kind of our version of that for wallets. We want to build a tool and collection of tests. That ensure that wallets all work the same way for their users, and that developers can be reasonably confident that.

Sam (06:40)  
All of your daps will work with every wallet. So these are the people behind it. There's me, there's my Twitter, there's my mastodon. And nikki is also working on this with me. They couldn't make it today.

Sam (06:54)  
But they're the one working on most of the tests. So what are the goals. We just got this out of the way. Here. There we go. Okay. So I went over this a little bit, but. The main goal is to provide a common set of tests that's available to all wallets to test their basic features. So this includes things like the eth underscore rpc methods.

Sam (07:21)  
Sending transactions. Receiving funds. Subscribing to logs all of the things that you can normally do with the ethereum RPC. That's what we're going to be testing first. we also want to make these tests available to anybody with a wallet to try

Sam (07:40)  
So. There's two ways to run the tests. There's an automated way in a manual way and the manual way. You just go to a website. And the website tells you what to click on in your wallet. And then it contests your wallet without any support. So this works for any wallet out of the box. Well, it will work for anyone out of the box. Let me not get ahead of myself. There.

Sam (08:04)  
Yeah. We want to improve the user experience of wallets by making them more consistent. So. This is kind of the motivation behind it. Some wallets, for example, used to send a transaction when you called Sign transaction, and there's a lot of little inconsistencies that are out there.

Sam (08:23)  
While Test Framework is designed to help reduce those inconsistencies and make everything easier for wallets and their users, we also want to improve the standardization process. So I come from the eip editing world, where a lot of people write a lot of standards. And one of the biggest problems with writing standards is testing to see if an implementation is compliant.

Sam (08:46)  
So if we have a standard test framework, it makes it much easier to write standards for wallets to implement because you can also test that the implementations are compliant. So those are kind of the big goals that we're trying to accomplish.

Sam (08:57)  
We have about a year to do it. We have the basic test framework written, and I'll be able to give you a demo shortly about that. But first, I want to talk a little bit about the technologies behind it. So the whole thing is written. Typescript.

Sam (09:12)  
I'm not a typescript developer, but this is what a lot of wallets are written in. So we wanted to make it as friendly for wallets to contribute to. We use ganache the fake blockchain so that runs in your browser.

Sam (09:24)  
This doesn't use mainnet everything's isolated. And it lets us do really interesting things. Like controlling Forks and lock timestamps and all the fun stuff you need to test wallets. Obviously, we use solidity, and we're the process of migrating to the vm library for handling, interacting with the wallet itself.

Sam (09:45)  
Yeah. So pretty much time for a demo. Any questions before I get into that.

Victor Zhou (@xinbenlv) (09:58)  
That's awesome. Looking forward to the demo and anyone. This is tended to be a synchronized. Comm. So if you have comments, questions, feel free to just. Say it. Yeah. Otherwise you can go forward to a Sam. Yeah. Please interrupt me.

Sam (10:16)  
All right. So. It's not exactly public yet because it's not perfect, but we do have a public facing heroku app that you can try this out on. It's a perfectly normal DAP in quotation marks. Where you connect like a normal wallet and it adds a test chain to your wallet.

Sam (10:40)  
Every time your wallet asks you something. This is the manual part. By the way, this is automatable. We just haven't written that part yet, but as long as we're on the manual stuff, every time your wallet pops up, you click which event the wallet is generating. So in this case it's an add Ethereum chain. You fill out the little form here.

Sam (11:04)  
And the rpc. Url. And then if you see over here, the dap tells you what to click inside of your wallet. So in this case, it's saying, approve the request. So we're going to approve that request. We're going to hit complete. And now we're actually running the tests.

Sam (11:26)  
So this goes through. It's using Mocha in the back end to generate these little nice reports and. In this case returns the mind block. It sends a transaction through. The blockchain. Mine's a block and then checks if the wallet returns the same block when you look it up by hash.

Sam (11:50)  
And there are going to be tests that require more interaction. We're just focusing on the non interactive ones for now because they're a little bit easier to get started with. But yeah. So every one of these goes through and uses the fake blockchain, asks the wallet for some information, and then checks it against what the fake blockchain reports. And that's pretty much the whole structure of it.

Victor Zhou (@xinbenlv) (12:09)  
What's the chin ID. What's the chin ID that you use for this.

Sam (12:19)  
Generated.

Victor Zhou (@xinbenlv) (12:21)  
Oh, it's rendering generated. Are you starting to use random generation now? Okay.

Sam (12:26)  
Got it? Yeah. So some wallets like metamask. Will cache the latest block and the nonce of accounts. Per chain. So you have to go in and hit a developer option to clear that. So instead of dealing with it, we just generate a random chain ID in the upper half.

Sam (12:46)  
Of the 32 bit integer and then. Just use somewhere in there.

Victor Zhou (@xinbenlv) (12:56)  
That's a good use case, because. I happen to also be on the author list of the neip which is trying to spell out. The explicit boundary of chain ID. There was the discussion whether Chennai should be deemed 256 bits.

Victor Zhou (@xinbenlv) (13:14)  
Or just 32 bits or 64 bits so that it can fit in the regular computer. I would assume if you're using random here. In this case it would be 60 bits in the boundary.

Sam (13:28)  
So for this one, I think I literally.

Victor Zhou (@xinbenlv) (13:33)  
Fake number, fake mat. Yeah. So that's deemed to be less than I think. 64. Pips, I guess. Yeah.

Sam (13:44)  
I'm not particularly trying to be cryptographically secure about it. If it fails, you can just try again. You end up. Getting. A decently good reports with back traces. That's pretty much the whole demo.

Sam (14:05)  
Yeah. So let me just flip back over here. I have a couple of links. If anybody's interested in following along, we have our blog at wtf Allwallet dev. We have our source code on github Wallet Test Framework framework and if you want to join our discord or follow along with the development there it's all available on All Wallet dev and.

Sam (14:28)  
That's pretty much all I wanted to show you today. So thanks for coming out, and thanks for listening.

Victor Zhou (@xinbenlv) (14:33)  
Thank you, Sam. This is awesome. Can you share the slide link so that if you're going to check. They could.

Sam (14:43)  
I think you're muted. Yes. Okay. Yeah. I look forward to it.

Victor Zhou (@xinbenlv) (14:49)  
All right. Thanks, sam's. I couldn't be more excited about this. I'm a big fan of open standardization, and then having a way and especially a scalable way. To test and check whether a certain implementation.

Victor Zhou (@xinbenlv) (15:04)  
Matches with complying with certain standard is a big advancement and we see that play out in css in JavaScript. Basically. In a lot of open web scenario. And then we're just taking that to the new height. Sam is taking that.

Victor Zhou (@xinbenlv) (15:21)  
And a wtf effort is just taking that to a new height. And I can't wait to see those reports and listing. All the wallet and give them a little bit nudge and incentive to be more compliant. I am super excited about this. Thanks for sharing, Sam, and.

Victor Zhou (@xinbenlv) (15:40)  
Go this effort, My love is. Okay. And also we write about time for the first feature talk. And then this is the Agenda today. We have to sign up for demo and then we can go for that first and then we also do the erc peer review request, which we have two of them today as well.

Victor Zhou (@xinbenlv) (16:06)  
And then we can always have more demos. Request. After. This already schedule the agenda. So let me check if Peter is here today. For the Hexlink demo. It's like peter's not I am going to Ping him and in the Meanwhile I think the good news is cayenne's one of my favorite project as well.

Victor Zhou (@xinbenlv) (16:34)  
Kims is here. And I do believe that cain's Project turbo ethan, among others, would be a big. Push for. All the. App. And so I'm personally using that kudos to him and. I also briefly encountered him in East Denver.

Victor Zhou (@xinbenlv) (16:54)  
So very happy to see you. Cam. Cams, go ahead and share the. Turbo eth and anything you want and if you can keep it under five minutes would be great.

Kames Geraghty (17:09)  
Sure. Yeah. Absolutely. So first off, thanks for the invite. I really appreciate it. It's always nice to share the tech with people who care and are interested. So. Yeah, I appreciate that. Cool. Let me open up the screen share. I got a lot of Windows open. Cool. Can you guys see the Turboeath website? Is that showing up? It is cool. Perfect.

Kames Geraghty (17:31)  
So I guess I'll provide a little context around. Why I built this. What's kind of the focus? So I've been in this space at almost six, seven years now. I worked everywhere from consensus to pool together and did identity and all this stuff.

Kames Geraghty (17:45)  
And so for me as kind of an app builder and someone who's been tinkering for a long time with a lot of the different protocols and what's out there. I noticed there was something missing in terms of how do you bring together.

Kames Geraghty (17:57)  
Integrations for these different protocols, whether it's identity or authentication or threshold cryptography. And so I started Turboea to kind of scratch that itch. So for those who are familiar with, like, say, scaffold Eggs or some of these other templates where it's really about kind of like building from scratch.

Kames Geraghty (18:17)  
Like scaffold, eth really focuses on building a smart contract where turbo E takes a different kind of approach. So I'm going to go ahead and just open up this integrations menu to show off kind of what are the integrations we have right now? And then I'll dive into how some of that works. But as of today, Turboeath is really about helping developers who want to basically experiment and tinker with a lot of the different tech that's available within Web Three. So we have everything from disco Identity. I have lit protoc.

Kames Geraghty (18:46)  
Protocol enabled we have pulled together. A dfi sign in with ethereum erc 20. So it's a mixture of these tools. So you can come to Turboeaf Turboeaf xyz. You can test some of these out. So I'll do a quick demonstration.

Kames Geraghty (19:03)  
I'm already logged in, so we don't have to log in. But like, for example, this disco integration, what this does is you only have to pop in an api key, and then what it's going to do? It's going to go fetch your decentralized identity. Here's my did document. And here are some of my credentials that are related.

Kames Geraghty (19:23)  
To my account have a lot of GMs I think over 50 GMs. So that's been cool. Another interesting example. Is lit protocol. So what we can do is there's a pretty basic integration that shows you how to have access controls and conditions.

Kames Geraghty (19:43)  
So you can set this up. Let me see if I can. I'll just copy my account real quick. So I can put in an address. I can save that. And then I can see the access controls that are going to be built. I can say, Hi encrypted.

Kames Geraghty (20:00)  
Message. And then encrypt right. And I'm now encrypting that for myself. And I can sign that. And that's controlled by Lip protocol. I'll skip that for now. Another one, I think, is pulled together. We have integrations.

Kames Geraghty (20:17)  
And it allows you to deposit withdrawal. But I want to talk a little bit about our approach. Right. So as developers, we like to have things modular, they're composed, can easily be shared. So a little bit of what we're doing within this repo.

Kames Geraghty (20:31)  
Obviously the repo on github turboeath template web three app. But if I go into this integrations folder and I go then into or the branch and then this folder. What we've attempted to do is colocate a lot of these integrations. So instead of an application.

Kames Geraghty (20:48)  
Having intermixing kind of all these protocols, developers can come here and pick and choose kind of what they want to bring into their app. So for example, the erc 20 integration has all the components abis. It even has a custom wagme config. So if you run the wagme cli, you can auto generate your react hooks.

Kames Geraghty (21:08)  
So really what this repo tends to do is simplify if developers want to share code share snippets whether it's smart contracts or identity or kind of. Anything in between. And the goal, I think is really just to help devs kind of start these new projects right now, over 26 forks, 157 stars, and just continuing to grow.

Kames Geraghty (21:33)  
So yeah, that's the gist. I mean, obviously, love, if you guys are building demo apps and you want something that I think is a little more. Flushed out in terms of it's. Got the latest wag me or we're going to update to V One. I think this weekend Rainbow Kit, a few nice ui components. Hopefully it helps you kind of build out your prototypes, but yeah, check it out turboeaf and if you have any questions, definitely feel free to Ping me, reach out Twitter or discord or wherever it is anyways.

Kames Geraghty (22:01)  
That's the presentation. Thanks for giving me a chance to show you guys. I appreciate it, Victor, for the invite and happy to answer any questions as well.

Victor Zhou (@xinbenlv) (22:12)  
Yeah, I'm Super excited about this as well. I think there's so much lower barrier for building at App. I started using it and it works. Wonderful, great. And then do you want to tell people if they want to contribute or if they want to reach out and say, hey, we have this awesome platform. We are the next lit protocol, whatever, how to contact you, how to convince the community to adopt them and put them into.

Victor Zhou (@xinbenlv) (22:37)  
Yeah.

Kames Geraghty (22:40)  
I think the best way is to join the discord. Right. A couple of things like one feel free to just open up an issue on this template. We use the issues for very actively curating bounties and integrations. I think we have over.

Kames Geraghty (22:55)  
Ten integrations bounties so far so if you do want to see your project integrated, make a ticket. We'll put some op tokens behind it and we'll get you integrated. But if you want to kind of join the community and a lot of these tickets, I have a link.

Kames Geraghty (23:08)  
To the Turboeaf discord so you can just hop in here. We have almost 80 people now. People are helping each other. We get feedback. so if you want to join the community that's kind of the best way

Victor Zhou (@xinbenlv) (23:21)  
Yeah, this is a super valuable community. A lot of people are helping out there. So it's a good place to learn as well. I really love this games. Every once in a while look forward to see new integrations or new demos on this. I can't wait to see it and then also maybe one time you can come over and have a longer time to share with us.

Victor Zhou (@xinbenlv) (23:42)  
Why you even started this? What's your long term vision in terms of where it goes in the future.

Kames Geraghty (23:49)  
Yeah. I'd be happy to right. Just to touch on that briefly. The long term vision is we need more devs in this space. And I think that just makes it lower barriers. Right. And I think for me, the front end is a great way to do that. And it's all about that. But I'd be happy to hop on again and kind of share that vision at some point.

Victor Zhou (@xinbenlv) (24:05)  
Awesome. Thanks, Kim. This is what we really want to see. And then hopefully that can advance the app we're building really open web and then things like that, including. What came shares, Turboeath, and then also the wallet.

Victor Zhou (@xinbenlv) (24:22)  
Test framework just make it so much easier to comply to interoperate. And this is why we want to have this effort.

Kames Geraghty (24:30)  
Yeah, totally. And again. Thank you for the invite. I appreciate it. Thank you, Kim. Yeah. Thank you for your time. Cool.

Victor Zhou (@xinbenlv) (24:39)  
And with that. We're going to the next. And then I think we will have two ERC requests for your comments. I think the first one was sign up by francesco suno. Francisco, are you there. Okay. I think at least George jown team is here so we can go with George and Davis, who are here to share with us the social recovery interface. So this is, by the way.

Victor Zhou (@xinbenlv) (25:09)  
An erc draft. So I think the intention is to share with people what you think and how you draft it and then maybe introduce the key design decisions that you made. And the community, the group of people who have different background but mostly surrounding building the app or providing infrastructure for the app or providing infrastructure for smart contracts will be able to give you their feedback and we intend to make it very peer review so every feedback is welcomed and also the authors are the driver.

Victor Zhou (@xinbenlv) (25:42)  
So they will make. Their own decision on whether to take that or not. But we want to.

George Zhang (25:49)  
Trigger and invoke more conversation. So with that.

Victor Zhou (@xinbenlv) (25:54)  
We'll be welcoming. I think the team for erc, assuming going to be seven, nine, three. And then if you could keep it under five minutes for the presentation.

George Zhang (26:10)  
Sure, Yeah.

Victor Zhou (@xinbenlv) (26:14)  
Good to see you. Hey, Victor, good to see you again.

George Zhang (26:18)  
So here I'm actually going to give a brief overview of the ercs. I guess some of my teammates might be able to develop it deeper into some of the specific technical decisions when it comes to designs.

George Zhang (26:32)  
So for this vip, we are actually just trying to abstract away some of the works we're kind of doing for ourselves. So like when building a smart contract wallets. And specifically, one of the main reasons why people want to be a smart contract wallet is because.

George Zhang (26:47)  
Different kind of recovery mechanisms to make it actually easier for people to kind of own and control their wallets. And so. In this proposal, we are trying to standardize. And abstract way of pretty general interface for social recoveries running out smart contract wallets.

George Zhang (27:09)  
And we just kind of defining a set of interface together with kind of how to use them. And what we think is a pretty general social recovery process. And for Smart contracts. And. Specifically. In this proposal. So we're just kind of discussing.

George Zhang (27:29)  
The specific interface we are kind of proposing. Okay. So am I supposed to. Let's see. It's okay. Yeah.

Victor Zhou (@xinbenlv) (27:41)  
So I think it would be easier for people to look at a screen and try to understand what you say. But I could also pull up my presentation.

George Zhang (27:48)  
I can present it if you want me to. It's up to you. Okay? Yeah. Let me share my screen. Okay. Yeah. Actually, I think it might be a better idea. Yeah. Let me share. Sounds awesome.

Victor Zhou (@xinbenlv) (28:00)  
Okay, give me 1 second.

George Zhang (28:04)  
Okay. Would people be able to see this.

Victor Zhou (@xinbenlv) (28:09)  
Not yet.

George Zhang (28:12)  
Let me stop sharing from my side and then okay, yeah, I think. I can see yours. It's working, right? Yes, it's working. Okay. Sweet. Actually I think I'm going to share the link. Essentially, this is the erc proposals.

George Zhang (28:32)  
We're kind of just abstracting away some of the key components. For what we think for the social recovery process, identities. Guardian Permission verifiers recovery Policy verifiers recovery accounts. So the identity here essentially is just kind of anything that would be able.

George Zhang (28:53)  
To. Sign and prove authorized. With their signatures. So we can just design it to be pretty general, to be able to accommodate wider range of potential identities. And we also define the Guardian permission verifiers, essentially to be an interface for part of the logics that essentially defines what exactly.

George Zhang (29:20)  
Is the process for us to verify the permission from the Guardians. And for the recovery policies. It's mostly. Trying to set things like. Time blocks Different weights. From different you are trying to assign to.

George Zhang (29:39)  
Different Guardians. It's kind of pretty common ways of verifying policy, verifying policies for different Guardians when doing social recovery, for example, like you could have multiple Guardians. And you can kind of assign different way students and when some of the ways to achieve a center threshold your country recovery account.

George Zhang (30:00)  
The recovery account is just kind of interface. We are kind of assigning for the smart contract while the account itself. So it's kind of interface for the entire account. It's going to going to include some of the logics there.

George Zhang (30:12)  
Below is just kind of more specific kind of interfaces we're kind of defining for the specific four components we talked about. In the proposal. We also kind of have a specific workflow. Work through.

George Zhang (30:29)  
Example we implemented. We're kind of trying to demonstrate. With the specific interface and how each part of the interface are used. I think to understand for the purpose of understanding. This diagram probably kind of helps.

George Zhang (30:45)  
To just kind of give an idea about how each piece is kind of used. Of course, here you can just kind of update Guardians on the top when you're trying to add different Guardians. With different covered policies to your accounts for the specific recovery process itself. Like when you're starting a recoveries.

George Zhang (31:09)  
So. You're kind of doing essentially going through three steps first, you're kind of just trying to verify that. You are actually having the permissions from all the guardians you set up in your recovery policies and.

George Zhang (31:24)  
Trying to generate the message hash. Similar to competitive message hash and. You're sequentially verifying the signatures from the each Guardians to show that you actually have the permissions and then you are going to verify that.

George Zhang (31:40)  
Actually based on this specific policy recovery policies, you set up. So. This is going to include things like, for example, different weight, different threshold for different guardians, and things like Time Lock features things like time block.

George Zhang (31:55)  
And so the idea here is if according to the policy, the specific set of permissions you submitted actually meets policies requirements to actually successfully recover. On the accounts. And then like.

George Zhang (32:10)  
Therefore is kind of handling the recovery. For example, the relayers will be able to then finally execute the recovery to replace. The owner of the account to your new designate account. So we are just kind of trying to abstract away some of them pretty common pieces saying social recovery, trying to make it pretty generous so that wallets can just kind of try and desire to fit the interface and the interface will be able to accommodate different kind of recovery.

George Zhang (32:39)  
Mechanisms and Guardians they want to set up and people will be able to customize it to their need. Without changing the Smart Wallet contract itself.

Victor Zhou (@xinbenlv) (32:53)  
That's cool. And then you're applying for erc, seven, one, two. Signs. Type data signature assigning right.

George Zhang (33:06)  
Yes. Actually. We actually. Have examples about how the signatures are going to look like.

Victor Zhou (@xinbenlv) (33:16)  
I see. That's interesting. Cool. Yeah. Any comments? Any feedback from the community, from the audience? I think. This is a very important user experience gap from the ideal future versus today's state of Ethereum and state of kupdel.

Victor Zhou (@xinbenlv) (33:39)  
So I applaud for George and the team's effort for this. Any feedback for them.

George Zhang (33:49)  
This is kind of our first. Erp proposal. So everything pretty new. Probably made a whole bunch of mistakes along the way. So any kind of feedback to how to improve the proposal. And. Make it feed communities, demand and need much appreciated. Yeah. One thing I can come up with is that have you looked into the erc six.

Victor Zhou (@xinbenlv) (34:14)  
Which is basically an extending version of erc 1271.

George Zhang (34:22)  
And then that allow it allows you to assign a token ID or.

Victor Zhou (@xinbenlv) (34:27)  
It bytes. 32. For signing. If you use 1271, you understand this. Like you you're asking a contract whether a signature applied is authenticated towards that contract. And basically erc. Six is adding a token ID. So you can ask the contract with the token ID and ask if this token ID.

Victor Zhou (@xinbenlv) (34:50)  
Of that contract. Signify verifies that authenticated with your signature so the use case would be hey if you want to have social recovery with people who hold certain nfts. Then they can say, Hey. I am going to authenticate myself with the nft holding under this token ID.

Victor Zhou (@xinbenlv) (35:12)  
And so Yeah. That can be helped. For example, if you like a board of company to Social Recovery Company's Treasury account. Then it can be tied to the nft holding of a company of the board seats, things like that. So that's one thing you can look into. And then by all means, it's your erc. So you make the technical decisions. Everyone else is just giving you the opinion and suggestions.

Victor Zhou (@xinbenlv) (35:38)  
And then your goal is possible to adopt yours and use yours. But making technical decision is all by all in your hand.

George Zhang (35:52)  
What's the exact numbers for the.

Victor Zhou (@xinbenlv) (35:59)  
Correctly. It's six.

George Zhang (36:04)  
That sounds perfect. Yeah. I think that's going to be like, a great choice to use when the Guardian type is actually. Kind of aft. I think it's definitely going to be a more suitable format. In that specific Guardian type.

George Zhang (36:20)  
When trying to verify the permission thanks a lot for suggestions.

Victor Zhou (@xinbenlv) (36:25)  
Yeah, Cool, thanks for presentation. And then you can always find George. And. The authors of this. Erc. On the eip. On the draft and so feel free to leave your comments there as well. This is awesome.

Victor Zhou (@xinbenlv) (36:44)  
And yeah. Kim's, thank you. Thanks for being here and thanks for sharing. And with that. We can move forward to Peter, who is here to share with us the hex link. I think Peter is up. So Peter. If you could keep it under five minutes would be great.

Peter Chen (37:04)  
Hi, yeah, Hi, everyone. Happy to be here to share with you our new Has link demo. I can share my screen for a little bit. Let me bring up the window first. Yeah. So basically. I can give a little introduction about what hexlink is. So basically the demo is demonstrate how a new user can utilizing.

Peter Chen (37:36)  
Can just receive cryptos or NFDS directly with your. Setup to get address first so the demo. can demonstrate how You can just receive a crypto first instead of you have to get a crypto address first to receive tokens.

Peter Chen (37:55)  
So for here I can input. My. Crypto email. And as you can see, I don't need to log in or anything. To just show my assets on crypto. And if I want to send a crypto to somebody. To send to my friends I don't need to.

Peter Chen (38:17)  
Know their crypto address. I can just simply type in their email address. For example, right here. I can just put in my. Another email. And I can just input the amount.

Victor Zhou (@xinbenlv) (38:38)  
What's ge, by the way, what does it represent? Getty oh, geez, represent the. Gordies.

Peter Chen (38:48)  
Yeah currently. We are still on Gorley but. We're going to move to the Minnesota next week. And here what happened is I will receive. A. One otp coding in my email, which I can input it here. And I click verify what will happen in the background is the bundler. Oh, by the way.

Peter Chen (39:20)  
I can show you guys the address that sent this transaction. Is a Account Production compatible Smart Contract account and this transaction is compiled by the bundler right now. As you can see it worked.

Peter Chen (39:38)  
And if I can show you guys. The email. On the other hand, Peter. The receiver email they should get. A as you can see, they will receive an email saying they just receive some gth. Receive some tokens, and if you send other erc 20 tokens, it will show this guy just received some erc 20 tokens.

Peter Chen (40:04)  
And if we go back here. And we type in the address. We just. And as you can see. There are some tokens just deposited. Into this account. So basically what happened in the background is. The background, the technical part.

Peter Chen (40:29)  
Is. Powered by eip 4972. Basically, we can drive an onchain Smart Contract account address directly from a user's email address. So. That enables that the user does not need to require to have. An onshing address first, which means.

Peter Chen (40:51)  
Their email can be calculated into. An unchained Smart Contract account address that can be used to receive. Cryptos. And yeah, thank you. Here's all for the Hexag demo.

Victor Zhou (@xinbenlv) (41:07)  
Thank you. This is awesome. I'm really a big fan of this approach. I think it just reduces the gap for entry barrier. And then for anyone who is drafting erc, for anyone who's doing the App and Smart contracts, I think it would be good to ask ourselves, how can we make it one step easier for the use.

Victor Zhou (@xinbenlv) (41:26)  
User and then taking into account that they might be. Holding email address where they may be holding Smart Contract wallets where they might be. Holding hexlink identity. Any comments and questions for Peter before we roll to the next agenda.

Victor Zhou (@xinbenlv) (41:54)  
No. Okay. Thanks, Peter. I think this is great. Thanks for coming. Thank you.

Peter Chen (42:02)  
Yeah.

Victor Zhou (@xinbenlv) (42:04)  
I think. We're done with all already. Existing agenda. There is an agenda. A hidden agenda that was not on the issue. Frame? Are you still there. Is a ens main developer. To the ens ecosystem. I'm very happy that we run into each other on East Ever. I've got a lot of question answered.

Victor Zhou (@xinbenlv) (42:33)  
He's great. So Frame, I think mentioned that he has interest in sharing something new. Idea. About I think subdomains right. These infrastructure supporting sites only so. Frame this your time. Take it over here if you can keep it under five minutes and then.

Prem (42:53)  
We can ask the comments and feedback as well. Yeah, look forward to it. Okay. Just to be completely transparent. I wasn't fully prepared, but at the same time, I am working hard on a project and I do plan at some point.

Prem (43:08)  
To author and eip. So this is just really a great opportunity for me. To just be here. Thank you. Victor. For telling me about it. Inviting me. I'm not exactly sure what I would share. Exactly. Maybe I could just talk briefly about our project.

Prem (43:28)  
I'm a community member of ens Dow, and that means I'm not on the enslab. Team. So I come at it from the open source perspective. And I got involved over a year ago. In the protocol. By reviewing the namewrapper Smart Contract, which is the new feature set that's been rolled out for ens.

Prem (43:52)  
Protocol and I was able to contribute some features. And find a few bugs along the way. And one of the features that I wanted to get into the name wrapper protocol was the idea of unruggable rentals, so rentals of subnames becomes a problem.

Prem (44:11)  
If the owner of the parent name. So when we're talking about subnames, we're talking about. Let's say something like Web eth. So if you want to have your name web eth as a subname of web. That can actually now be what's called emancipated, meaning given to an additional subname owner outright so that the parent name can't take it back. And that's one of the new features of the namewrapper Smart Contract that ens protocol has rolled out the idea of emancipating these subnames that creates a huge amount of potential for communities to build.

Prem (44:47)  
Independent naming under their name so Imagine bankless. there's also cbd id which is slightly different because it's a di but it's being used as an identity layer as a wallet naming for coinbase but under the eth which is what i'm particularly focused on you can actually build these subcommunities under these names and people can own them as outright nfts that they own

Prem (45:11)  
but there's a problem is if you allow the subname To be owned outright indefinitely, forever. Then you have a degrading of the namespace. So the namespace people will get the names they'll forget about the names they'll pile up in an old wallet. They'll lose the keys, the wallet. All of a sudden you lose the name to the namespace.

Prem (45:32)  
That's the reason why ens decided at some point to switch from. A perpetual ownership of eth Names to rental. It's $5 a year for most things. So in order to create this feature, we need to find a way.

Prem (45:49)  
For basically anybody to rent these subnames be the renters of subnames. Without controlling the rental price. See if they control the rental price, then all of a sudden. The subnames become ruggable.

Prem (46:05)  
Meaning that anybody can take back the name by arbitrarily rising the rental price. So if you raise the rental price to thousands of dollars, then all of a sudden you basically get the name back as apparently.

Prem (46:17)  
So what ended up happening is we created unrugable names. And that's what we're building now. Unrugable names. And that extends the capability of the namewapper Smart contract with what we call the subname wrapper. Subname wrapper creates.

Prem (46:31)  
A connection between. The renewal price. And the owner of the subname that is independent of the parent name, and so. This concept started to build into what I'm calling Unruggable protocol. An unduggable protocol.

Prem (46:49)  
Is basically kind of like an oracle service, but it features much more the idea of. The data that we need to depend on and who maintains that data. So in this case it's the $5 renewal price, which is our first Unruggable protocol. We're calling books. The first book we're creating is called.

Prem (47:10)  
The $5 price, the $5 price is tied to the ens. Dow's $5 renewal price. So it's a little bit complicated. And again, I'll have to be much more prepared next time. But that's a little bit of the overview of kind of where we're going with the unregular Protocol, which is basically kind of like an oracle service.

Prem (47:30)  
That needs to solve this using this use case which is unruggable rentals. To build this protocol.

Victor Zhou (@xinbenlv) (47:41)  
That's interesting. I thought it was. Like. It's an interesting design to have subdomains as also requiring renting. And I didn't anticipate there is this side effect that people can hike the price so that they could.

Victor Zhou (@xinbenlv) (47:59)  
Take away the subdomains by just hiking into intimidity.

Prem (48:04)  
Right. Yeah. That's interesting. I recently.

Victor Zhou (@xinbenlv) (48:09)  
Came to know that. There was. This tax approach where. They were tax by you assigning your own price. Which you are happy to give up this property. And then. There's a community based tax rate so you will be charged.

Victor Zhou (@xinbenlv) (48:32)  
In recurring tax given your self assessment. Tax Self assessed property price.

Prem (48:40)  
Have you looked at. From Vitalic's ideas that he posted on Twitter. Harbinger Tax, Yeah.

Victor Zhou (@xinbenlv) (48:46)  
Covenant tax? Yeah. That just recently come. And I think Vitalik talk about that like a few years ago. Isn't one of the interesting approach. To me. I'm exercising this idea into putting into context but realizing that setting the tax rate.

Victor Zhou (@xinbenlv) (49:05)  
Is a design choice or a problem.

Prem (49:08)  
Or how does the community reach a tax rate that they agree to is an interesting problem. I don't know what's your thought on that. My thought is that using our protocol extension. So the protocol extension is called subname wrapper. And using the protocol extension, anybody can build their own renewal controllers. These are the things that control the renewal and.

Prem (49:28)  
So you can build a Harper Tax model. You can build. A fixed eth price model. If you want it to be a little crazy and you could say you could just point one eth. And if eth goes to $100,000, then the price of the domain. But it's all on chain. Then no oracle Surface so you can kind of build whatever models you want as these. It's called Renewal controllers.

Prem (49:51)  
And the feature of the subname wrapper is that it wraps. The renewal controller with the name. And it prevents the owner of the name from accessing that or changing that renewal policy. So I'm hoping that it creates a whole ecosystem of experimentation around the idea of subname renewals.

Prem (50:11)  
Protecting the namespace which again if you have bank, if you think of it as bankless east, their namespace is pretty important. They may help to have that community for the next 10, 20, 30 years. And so these names that are under bankless studies become very important to have a strategy for protecting the namespace.

Victor Zhou (@xinbenlv) (50:32)  
That's true. Yeah. I agree with that. It's very important to find a strategy for to make sure that at least the subdomains are unraggable as well. Yeah. So this is exciting. This is opened up a lot of new opportunity. And when you get a chance, I look forward to see the erc, drugs.

Victor Zhou (@xinbenlv) (50:58)  
And then also the demo. If you want to and then people who are on this call feel free to comments, have questions, or if you like to collaborate. If you think what you do can depend on can use what premise is building. Feel free to reach out.

Victor Zhou (@xinbenlv) (51:19)  
Yeah. Any other questions? Comments.

Prem (51:27)  
For frame that's the link to our current demo.

Victor Zhou (@xinbenlv) (51:31)  
Oh, Wow. Find you the perfect name. Okay getting some feedback. That's a good approach.

Prem (51:49)  
It seems like you're using turbo east. We're using turbo East.

Victor Zhou (@xinbenlv) (51:55)  
Yeah. It just makes it so much easier. Like every hackathon actually should promote triple ethan, and I also hope to see. Competitors with turbo East. You just keep making building the app so much easier.

Victor Zhou (@xinbenlv) (52:10)  
Cool. Look forward to it. I see this is an early demo. Look forward to. See the next one on you already. All right. So I think we have all our agenda done today. So I generally like to think of it as a way to kind of put people together in a room.

Victor Zhou (@xinbenlv) (52:38)  
We can have social chat or if you have any before we have social chat. Do you have any technical. Peer required things that. You want to share with the community and then. This is recorded by the way, and then we'll be shared on YouTube. And then once we turn it off, we can just grab a cup of coffee and then just briefly chat whatever else you are doing without on the record.

Victor Zhou (@xinbenlv) (53:02)  
So yeah anything you want to share. No. Okay. All right, I'm going to turn off the recording. And so we turn it into the social chats of it. The recording has stopped. Now kick out a nose taker as well.

Victor Zhou (@xinbenlv) (53:26)  
now