Bruke Kifle: This is ACM ByteCast. A podcast series from the Association for Computing Machinery, the world's largest education and scientific computing society. We talk to researchers, practitioners and innovators who are at the intersection of computing research and practice. They share their experiences, the lessons they've learned and their own visions for the future of computing. I'm your host, Bruke Kifle.

In an era marked by ubiquitous computing and continuously evolving threats to data security, the importance of cryptography in our digital lives cannot be overstated. It serves as the bedrock of our technological systems, preserving data privacy and ensuring the secure transmission of information. From encrypting and decrypting messages, to verifying transactions and authenticating identities, cryptography encompasses a wide array of applications crucial to our digital age. And as technology advances, new challenges and open problems continue to emerge, requiring innovative solutions. Today, we are joined by Dr. Yael Kalai, whose research has not only advanced the theoretical foundations of cryptography, but has also influenced practical applications with many implications. Dr. Yael Kalai is a senior principal researcher at Microsoft Research and an adjunct professor at MIT. Her main research interests are cryptography, the theory of computation and security and privacy.

She's especially known for her work in verifiable delegation of computation, where she has developed succinct proofs that certify the correctness of any computation. In addition to making breakthroughs in the mathematical foundations of cryptography, her proofs have been useful in areas such as blockchain and cryptocurrency. Dr. Yael earned her Bachelor's of Science in Mathematics from the Hebrew University of Jerusalem and MS in Computer Science and Applied Mathematics from the Weizmann Institute of Science and a PhD in computer science from MIT. And is the recipient of numerous awards, including the 2022 ACM Prize in Computing. Dr. Yael Kalai, welcome to ByteCast.

Dr. Yael Kalai: Thank you. Nice to be here.

Bruke Kifle: You have such a remarkable background. You've studied under some of the most renowned cryptographers and you yourself have made major contributions to the field. But what I find interesting is, most people have a pretty unique journey or story that has led them to where they are today. So, can you highlight some of the key inflection points within your personal and professional career that have ultimately led you to where you are today into the field of computing and cryptography as your field of research?

Dr. Yael Kalai: Yeah, sure. So growing up and as a young adult, I really loved mathematics. Actually, I have to admit I was not good at anything else. So I felt like that's the only path that makes sense for me. And I studied... As you mentioned, my undergrad was in mathematics. I loved it deeply. I really fell in love with the subject. I actually started doing my PhD, like my master's in mathematics, also in

the Hebrew University. I left after one semester. But I think the reason I moved to computer science was, it's interesting. I still feel like I'm a mathematician in disguise. What really interests me is the beauty of mathematics. However, when you go to just pure math, it's very hard to find questions that are... I felt like, that are not incremental. That I felt are groundbreaking. There are, of course, groundbreaking questions, but they seem almost impossible to solve.

So when I just stepped forward trying to do research in mathematics, it felt like things are either close to impossible or very incremental. And I couldn't find a problem or a direction that excited me. So studying the field was exciting for me. But doing research was, I struggled. To find something that excited me. And that's when I took a step back and started thinking about maybe I can move a little bit to the left or a little bit to the right and still do mathematics, but in a way that's more not so incremental and something that may have some influence and maybe more than one person will read my paper, a paper that I write or something that will have more significance. So I really wanted to have real world impact and do base, fundamental mathematics at the same time. And these two seemed a bit contradictory, and I looked around to see whether there's a field, a subfield of mathematics that I can converge to.

And that's when I found theoretical computer science. So computer science is a much younger field, of course, and there were still a lot of open... Seemed fundamental open problems that did not seem impossible to solve. It was young enough that new interesting problems came about often. And underneath the actual problems that we're thinking of, is really problems in discrete math. It's really mathematical problems. So that was the journey that took me to the Weizmann Institute. In particular I went there because, actually I don't even... Okay. To be frank, I don't actually know really how to program. I'm not interested in programming. I have to say, when I got the ACM prize in computing, I told my kids, all three of my kids, even my young girl knows how to program. And my older two kids are really amazing programmers. And their reaction when I told them, it was like, "Really? Is this a joke? Do they know that you know nothing about computing?"

So, really my interest is in basic math. But theoretical computer science and cryptography in particular, have a lot of that to offer. And I think the reason I went to cryptography within theoretical computer science is many factors. I'm not actually a 100% sure. Definitely one main reason is, I took a class by... You mentioned my renowned mentors. I took a class by Adi Shamir in Weizmann Institute, who was a Turing Award winner and I was just blown away. I was blown away by the subject, I was blown away by him. I remember my eyes lit every time I walked into his class. And so, I don't know how much of it was based on just his personality and how much he captivated me and how much was the subject at hand. My research is not solely in cryptography. So I do also research in other areas of theoretical computer science.

So, I feel like I'm a bit malleable within theoretical computer science. But one thing I really love about cryptography is the questions that we ask are so fundamental and so philosophically interesting. What does it mean to know something? What does it mean not to give information? How do we define that something gives no information? It's very philosophical terms that we need to state mathematically and rigorously. And I find that really interesting to... And I think that's partly why our... So we're actually dealing with things that we want to use and apply. And we want to give proofs, and we want these not to give information. We want to be able to verify correctness of computations. What does that mean?

So I really enjoy the fact that we're dealing with real world problems that are just interesting problems. I can explain them to my mother. I would say my father, but my father happened to be a scientist, so that's not meaningful. And so, I feel like I'm working on problems that are interesting and I can explain them to the public, of course, how we solve them. I can't explain...

Bruke Kifle: Of course.

Dr. Yael Kalai: But at least the problem itself has general interest in my opinion.

Bruke Kifle: It seems like you have a strong, just excitement and passion for open questions and solving problems. And I love how it seems like every scientist always starts with this love or passion for math. And then coming across an instructor or a mentor or professor that really helps you further solidify that interest. So I found it funny when you said you felt like you were a computer scientist in disguise, because I remember taking a lot of math courses in undergrad and feeling like a math student in disguise. So I think it goes both ways. So all that has ultimately led you to... And you said amongst many other research areas, cryptography is one of the segments within the field of computing that really struck your interest. But I want to start off high level because, not a lot of people might actually know what cryptography really is. It's a fundamental pillar of modern security. But like you said, the technical aspects can be very complex. So, could you provide a high level explanation of what cryptography is and why it's important in our digital age?

Dr. Yael Kalai: Yes, definitely. So most people think of cryptography or what cryptography used to be is a way of securing communication. So in other words, if I want to send a message to someone, I want to make sure, A, nobody else can read this message. That when I send this message over some network, an adversary cannot see what I'm sending. So I want to be able to communicate privately. Another thing I want is to be able to make sure that the message I sent was indeed received without being changed. So I want to make sure that we have some form of, what we call authenticity, that the message that was received, maybe it was dropped, that may be didn't... In adverse, we can just perhaps drop a package, a packet, but I want to make sure that if it's altered, then the receiver will be able to know that.

So if something happened to it, it'll say, "Oh, wait. Something's wrong. That's not Yael's message." So what I want to ensure is, A, nothing about the message is leaked. And B, if it's tampered with, the receiver will know that it was tampered with. And so that's what mostly what cryptography was like for many, many, many years. Today, as you're saying with the way the digital age is changing, cryptography is much broader than that. And a lot of things that we deal with today, actually have to do with securing computation and not only communication. So what do I mean by securing computation? Today, a lot of large scale computation is happening.

For example, our digital medical data is sitting somewhere. A lot of our private data is sitting in various places, various servers. And we want to make... Okay. So we want to... Of course, that we want to make sure it's stored correctly. But moreover, we want to be able to do some computation on this data. For example, maybe we store our data somewhere in some hospital, but we want to allow a researcher to run some computation on this data. How do we run a computation on an encrypted data? So we don't want to give the researcher all the private data. We want to respect the privacy of the patients, yet we want to allow the researcher to do these computations and the encrypted data. So how do we deal with a lot of data that's private, yet we want to get some utility from this data.

Another thing that's happening is that, because there's large scale data. For example, think of, you mentioned blockchains. Today we have public ledgers used by Bitcoin, for example, or many other cryptocurrency companies, they have huge amount of data sitting on public ledgers. And for example, in the case of cryptocurrency, to verify a transaction is a huge computational burden. One needs to make sure that this coin was given to the owner and was not double spend. And whoever gave him the coin got this coin from someone else, who did not double spend. And that person got the coin from someone else who did not double spend. It's a huge computational burden. How do we know that things were done correctly? How do we know that it's indeed a valid coin? So ideally what we want is some proof. Some proof that says, "This computation was correct." So here I'm saying privacy is one aspect.

Another aspect is verification. How do integrity of the computation, someone is doing this computation. Someone told me in the blockchain, "Yes, this is a valid computation." How do I know that that's the case? So sometimes you manage to incentivize, you do some game theory, use game theory to incentivize the users to be honest. But sometimes you want to just have a little proof that tells you, "Oh, yeah. This computation is correct." Now here it seems to be unrelated to cryptography. Because usually when you think about cryptography, we think secrets.

Bruke Kifle: Exactly.

Dr. Yael Kalai: I'm saying no secrets. We just want a little proof. You think, "Oh, proof. That's math. What's cryptography about it?" But it turns out that we want of course, a proof that's very succinct, like a little certificate that certifies the correctness. It turns out that in order to get these succinct certificates, we must rely on cryptographic assumptions. So we must rely on some hardness like that. It's very hard to factor very large numbers. Or these hardness assumption that we use every day in cryptography. And we need to rely on these assumptions to generate these succinct proofs that certify correctness of computation. So just going back to your question, my answer is, what is cryptography about? It used to be only about securing communication, both secrecy and integrity. But today it's much more about securing computation, both secrecy and integrity.

Bruke Kifle: You just described that so perfectly, and I think I just could not imagine a better way to capture this idea of securing communication and the advancements that we've seen in the field to now moving towards securing computation. So I think what you described in the end is, this idea of the verifiable computing work. The delegation of computation, which as I understand is the breakthroughs that you've pioneered in this space have been a big factor for your 2022 ACM prize in computing. So how is this approach different from traditional approaches? So traditional approaches, as I understand, all those computations would have to be done and the system is inefficient. Is there some other traditional approach to computation?

Dr. Yael Kalai: Yeah. So here's the problem we want to solve. Someone that we may not trust, did some computation. It ran some program for a very long time. And it got an output. Now we want a little proof. A little. It's short. I need it to be short because I need to verify it efficiently. I want a short proof that indeed this is the outcome of running the program. Now the thing is, most... I know most, but many natural programs, there does not exist a succinct proof. A short proof. So for most or many natural functions or programs, how do I prove to you that I did something that something is... This is the outcome. Let me give you an example. For example, this is a bit mathematical maybe, but take a chessboard. Let's say you have a chessboard, you have some pieces on the board.

Now I want to prove to you that the black player has a winning strategy. Then I want to prove to you, no matter what the white player does, the black player has a move such that no matter what white player does, at the end he will win. How do I do that? I really don't know. How do I give you a short proof? The only proof I can think of is, I'm going to tell you, "Well, first the black player will make this move. Then for each and every possible move of the white, this is what the black player will do. Then for each and every possible move of the white player..." It's a huge proof. It's like an exponentially sized proof. So that's an example of a computation, I don't have a succinct proof for. And many natural computations, I don't have a succinct proof for.

So what we do is we rely on cryptography to do it. And there's a price. The prices, we actually don't offer the guarantee which mathematical proofs give

you. Which is, "Oh, there is no fake proof." Either the statement is true, in which case you can prove it or it's false, in which case you simply cannot prove it. Our guarantee is weaker. We have a computational guarantee. We say, "If it's true, you can prove. If it's not true, it's hard to prove." It's not impossible. Maybe someone can prove it. But if someone can prove a false claim, then he can break a very hard cryptographic assumption, such as he can factor very large number. And that we believe you cannot do. Because if you could do, then probably the economy would've already crashed. Because all our transaction depend on the hardness of this problem. So our proofs are not really proofs. They're what we call computational proofs. It's not like it's impossible to fake a certificate of correctness. It's just very, very, very hard. And we believe nobody can do it today.

Bruke Kifle: I see. I think we'll circle back on this later, but I would love how some of these underlying assumptions break down in the context of some of the advancements with quantum computing. But, I would love to table that discussion for the end because...

Dr. Yael Kalai: Happy to talk about that. It's very interesting.

Bruke Kifle: Certainly. I do want to deep dive a bit more on this. So you described this verifiable computing essentially as a way of ensuring the correctness of those computations performed by a server or some blockchain nodes. So as I understand it, one of the primary objectives here is efficiency. You want to ensure that the verification process doesn't introduce too much overhead or become computationally impractical.

Dr. Yael Kalai: Exactly.

Bruke Kifle: You do this by minimizing the resources of verifying the proofs. You reduce the proof size, you reduce the computational complexity. On the other hand, there's this concern of privacy or security. So I'm assuming that in cases where you want to verify these proofs, there's some data or some sensitive information about the computation that has to be disclosed during the verification process. And correct me if that assumption is wrong.

Dr. Yael Kalai: Yeah, actually that is incorrect. Let me explain. So it's very interesting. There's beautiful, beautiful works and achieving, getting zero knowledge proofs, actually proof that reveal no information. It's interesting because circling back to your first question, which is, how I got to cryptography. I mentioned Adi Shamir. But another person that I definitely should mention is my most amazing PhD advisor, Shafi Goldwasser. And the reason I'm mentioning her is not because probably she's the person that... Because of her I stayed in the field, but also she invented together with a Silvio Micali and Charlie Rakoff, the notion of zero knowledge proofs. And what they showed is, they can convert. Or what was shown actually after that, is that one can convert any proof into zero knowledge proof. And so, one can convert our little succinct proofs into zero

knowledge ones. So one can take, this is kind of technology that we know from the '80s, really. One can take any proof. Long, short, and you can convert it into a zero knowledge one with actually placing pretty minimal overhead on top.

Bruke Kifle: And zero knowledge refers to...

Dr. Yael Kalai: Good zero knowledge means, that proof reveals no information. Beyond the fact that the statement is true.

Bruke Kifle: Oh, wow.

Dr. Yael Kalai: It's really, no information is revealed. It's pretty amazing. I remember when I studied this notion, I practically couldn't fall asleep at night because I felt it's like magic. How can you convince someone that something is true while revealing literally no information beyond the validity of the statement? And you're saying, "Wait. But a proof is information. What do you mean no information? It's the proof. You can read and verify."

Bruke Kifle: Yeah.

Dr. Yael Kalai: Okay. So it's something to think about. It's actually... I am hoping maybe I'll get some of my listeners to join us in cryptography.

Bruke Kifle: So essentially, you can achieve this verification process without... While preserving privacy, without... Like you said, without revealing any information about the computation or about the input data to actually enable the verification.

Dr. Yael Kalai: Exactly.

Bruke Kifle: Wow. That's beautiful.

Dr. Yael Kalai: That's amazing. Yeah, I agree.

Bruke Kifle: So if I understand correctly, a big part of your work or this line of work is developing the methods for producing those succinct proofs that certify the correctness of the computation. So then you can offload those computations without compromising security or privacy. How do you come up with these kinds of ideas? In what context? Is it in lab meetings? Is it on walks? Sometimes I'm so fascinated by the type of innovative solutions that researchers come up with. So I'm curious, how do you come up with this idea?

Dr. Yael Kalai: Yeah. So there's two types of ideas. There are ideas of which problem you want to study. And then there's ideas of how you come up with a solution. The reason I'm partitioning the two is because, ideas for which problems to study, they're usually much more high level. You don't necessarily need to be in a research mindset. You walk around, you can read the paper, and up comes an interesting

question. Just by listening to what's going on in the world. I don't know. Now there's these large language models. Oh, guess what? It brings a lot of interesting questions. The cryptographers. So just living and looking around what problems the world throws at you. The world throws problems at us all the time, because changing like blockchains. Wow. Throw with this technology, bought with it a lot of challenges, and that's fantastic for us because, I felt like the difference between theoretical computer science and mathematics, that we get new challenges all the time, and these are really, really interesting challenges because it's kind of how our world is shaped.

So that's about which problems to solve. How to solve the problem, wow, that is really... Usually A, we really need to dive, at least that's the way I work. I think different people work a little differently. But for me, I usually dive, I don't know, a 100 feet deep and I get so obsessed. I think I'm probably... I remember problems that I worked on that I literally abused my students, emailing them every hour. They abused me back. So it's okay. But constantly, "Here's an email. Oh, wait. This doesn't work. Oh, it doesn't work." This intense thinking and then waking up in the middle of the night, because I think I have an idea. And then I get up and then, "Oh, it doesn't work." And then I go back to sleep. And then after two hours I wake up again because actually I think it should work.

And this kind of... Usually the solutions for me come when I'm really obsessed. I breathe the problem, I sleep the problem. I am very, very much obsessed about the problem, in a way that I feel like sometimes, "Oh my God. Either I need to solve it or someone else needs to solve it, because I need to get out of this misery. I need to live again. I need to breathe a little bit and not be..." I feel like my head goes, so... I'm so intense in thoughts, but that's for me. I know that different people think differently. Also, I really enjoy collaborating. And so when I get intense about, like now, I'm working very hard on a problem. And again, it's like we meet and then I go home and two hours later, "Oh, can you hop on Zoom again?" It's constantly... It seemed like my student who's working with me is constantly on the project as well.

We're both obsessed. It's like an obsession. But an obsession that I really enjoy. It's really, really fun. But I think, again, different people work differently, and I think it depends also on the type of problem you're working on. So I work really usually on problems that are quite technical. The solutions are usually quite complex. There are other problems that other people work on that are just a moment of brilliance. It's simple in hindsight. So there's many different ways to do research, even within theoretical computer science. Of course, outside of theory, it's very different because a lot of it just requires actually work. You need to sit and do. You need to, to write the program. A lot of it is, that you start the day, you finish the day and you see progress. In theory, often you start the day, you end the day, you just feel like you made negative progress.

Bruke Kifle: No progress.

Dr. Yael Kalai: Or you feel like negative because the ideas you thought should work, you realize they don't. Which actually is progress, but it doesn't feel that way when you... Yeah. So that's usually my style. But again, I want to just make sure technology, they're a lot of different styles.

Bruke Kifle: Certainly.

Dr. Yael Kalai: Different people work differently.

Bruke Kifle: So certainly obsession is, I would say probably a key theme if you're obsessed and love and are up at night thinking about it. But to your point, I have met folks like you said, who maybe it's a spur of the moment type of thing where you're walking your dog or you're taking a shower and the solution comes to your mind.

Dr. Yael Kalai: Exactly.

Bruke Kifle: But in areas where maybe you're working on very deeply technical or mathematical pursuits, the approach to problem solving can be different.

Dr. Yael Kalai: Yeah. I would say I have some of those as well, but it's a small piece. So sometimes I'll be in the shower, I'm like, "Oh, I think I can solve this piece." There's some idea, but usually it's one idea out of many. And then the questions, can you piece everything together? Does it... It's usually... Yeah.

Bruke Kifle: So you touched on one piece, which was actually the next topic that I wanted to discuss, which is, AI and large scale language models. So some of the stuff that you discussed with this verifiable, distributed or delegation of computing, at least I think has some practical applications to some of the exciting progress that we're seeing with LLMs.

Dr. Yael Kalai: Definitely.

Bruke Kifle: I'm curious more generally, how do you see this line of work or broadly your research aligning with some of the current trends with generative AI?

Dr. Yael Kalai: Yeah, so this is something I'm thinking about now, both with my students, with Shafi, who was my PhD advisor, with a bunch of people. I think probably many people in my community are thinking about this, and there's many questions. So, okay. Of course there's these large language models and we have no idea what they're doing. And we want to make sure they're doing, the answers they're giving us, we want to be able to trust them. Want to be able to trust that they're giving us the answers that they should. The thing is, it's not even clear what this means. And it's not clear what this means in many, many, many levels. So level one, you may not trust the company that generated these LLMs. I know, OpenAI. Maybe you don't trust OpenAI. Okay. In that case, you tell OpenAI, "Oh, in that case, you can use my technology."

You tell OpenAI, "Oh, you gave this model." Now every time, I don't trust the model. But every time the model spit something, just add a little certificate that... Oh, sorry. Let me start again. I ran ahead of myself. Let's say you don't trust OpenAI. OpenAI gives you a very large LLM. You're saying, "What is this thing? How do I know that it's good?" Well, in that case, OpenAI can tell you, "Look. What did I use? I used some neural net." You see, it's very small. The neural net is very small. I'm going to prove to you that this huge LLM is the output of applying this neural net to this huge amount of data. Okay. Let's say you can use my technology to add a little proof that says that this LLM is indeed the result of running the neural net on this data.

Okay. Question. How do you know the data is good? Where does this data come from? How do you know that it's valid data? That's a question. How do you prove that the data is good? What does it mean for data to be good? Or how do we prove that some data was sampled from the correct distribution? Even simpler. Let's say you want to sample a bunch of random bits. Okay. I sample. How do I prove to you that it's random? What does it even mean after you sampled is 010001 random? Is 00000 random? Is one more random than the other? I don't know. Each one of them have probability one over two to the length. So they're all equally likely. Why is one random? Defining what it means even to be sampled from the right distribution is a really interesting question that we're actually currently thinking about.

But even suppose you trust the data, you trust OpenAI, we solved all these problems using our technology, let's say. We're still have a problem. Because, OpenAI told you, you trust them. I ran this little neural net on all the data from the internet. I got some LLM. Yeah, we believe you. Still, how do we know that LLM is doing what it's supposed to do? Nobody understands what this thing generated. So now what does it mean? What does it mean to trust this thing? Do we trust it? It depends. What does it mean to trust? When we started the podcast, I told you that cryptography, one thing I like about it is that, it deals with philosophical questions and it puts it on mathematical grounds. And this is one example. I want to say, we all know... Just speaking in English, we all are concerned because we don't trust the LLMs. How do we know that they won't convince us, we'll ask them a question and they'll give us the wrong answer in order to be for the sake of maliciousness.

Because they want this to take over the world or whatever. How do we know? So we want to be able to trust. What does it mean to trust? When do you trust? If he gives you the right answer, the right answer is not well defined. You ask him, "Should I... What's the best thing I should do to prevent global warming?" Is the right. Do we know what's right? So what is it... On questions where the answer is clear and you can check, fact check, Okay. He can give you the check, he can certify, he can tell you, "Okay. This I got from Wikipedia." You see? Okay, fine, that we can fact check. But there are some questions or answers that he'll give us that we can't fact check. So what does it mean that he did it correctly? How is correctly defined in this case? How do we define trust? So there's a lot of

super, super interesting questions that we're now dealing with, and I think it's a very, very exciting era for cryptography.

Bruke Kifle: Yeah, these are very large, bold. I think you described them as philosophical questions. And I guess at this point, it's hard for me to even imagine how you can ground these in a mathematical or scientific foundation, but I am curious what lines of research or work emerge as a result of some of these open questions?

Dr. Yael Kalai: Yeah, we should talk in a year from now.

Bruke Kifle: ACM ByteCast is available on Apple Podcasts, Google Podcasts, Podbean, Spotify, Stitcher and TuneIn. If you're enjoying this episode, please subscribe and leave us a review on your favorite platform.

Awesome. So one other innovation that you have been attributed for co-inventing is ring signatures. Right?

Dr. Yael Kalai: Yeah.

Bruke Kifle: So, can you provide an overview of what ring signatures are and what exactly was their motivation when you pursued or developed?

Dr. Yael Kalai: Yeah. So actually this idea when I was in Weizmann, under the guidance of Adi Shamir, I co-invented with Adi Shamir and with Ron Rivest, who's also a Turing Award winner, and also one of my mentors throughout the years. So the goal there was the following. So as I said, a lot of cryptography is about authentication. We want to make sure... So for example, when I send you a message, I'm going to digitally sign it, so that now you can verify that it's me who sent the message. And the question we were asking is the following. What if I want to be able to sign a message, but I want to keep myself private. So it seems like, "What do you mean sign? Keep myself private." So I want to say, "Oh, it's me, but I want to keep me private." What does it mean? So what I meant... What we want to say, for example, we actually called the paper, How to leak a secret?

The idea we had in mind is, let's say I want to tell my professor in class that some of the students cheated in the test, and therefore it's not fair. Now I don't want to be a tattletale and tell, "No, it's me." And now everybody maybe will leak to someone that I was the one who tattletaled. So what I'm going to do? I'm going to write him, the professor, a message and tell him exactly why I believe that the cheat happened, so he'll be convinced that there was a cheat and I'm going to sign it on behalf... Okay, that's not the best example, but they did. I will sign it on behalf of someone in the class. So I'm going to say, "Someone in the class sent this message." And this is what it says.

So in other words, think about people in the NSA want to leak a secret. They don't want people to know it's them. So they say, "Okay. This is what I learned while I was in the NSA." I'm signing it by someone from the NSA, but nobody knows who was from the NSA signed it. So we were inspired, one the reason we were inspired, because there is such a notion of group signature where you can sign, but that's on behalf of an entire group. It's like a group that you got together and you decided that you are in a group, you agreed together and kind of a key and a secret key for you guys, and you think of yourself as an entity. So now you're one group and you're one entity, and you can sign on behalf of the group. But in ring signatures, I can just sign on behalf, you or me.

Bruke Kifle: Of an individual.

Dr. Yael Kalai: Of an individual. I don't need his consent. Actually, when we did it, one of the things I remembered is, I was a bit scared by it. Because I was thinking, "Oh, that's a little scary that we have these signature schemes and these signature schemes allow us." They give us the technology. That's what we did in the ring signatures to allow me to send a message on behalf of someone of me or someone else, so I can send a message, "Me or you murdered someone." You wouldn't want a message like that being sent, because nobody knows if it's me or you. So I can't really frame someone, but I can say, I can put them in a group that one person in the group is framed. So we put this... This paper was more of like, is it a good thing? Is it a bad thing? Was it interesting?

Bruke Kifle: So I am curious, now that you've raised that, beyond the use case for, as you described it, tattletales or more nicely for whistleblowers, are there practical applications or are there practical scenarios, real world scenarios where you think that this might be useful or relevant?

Dr. Yael Kalai: Okay. Let me say, I know that there were blockchain companies that used these ring signatures. And I'm pretty sure it wasn't for whistle blowing. I think Monero used it, and maybe there were more. I'm now blinking and I don't really know exactly their use case for it. How, for what sake did they use it? My guess is that probably they used it to get some privacy on the chain. So currently when you do... Well, there are various, now of course currencies. But for example, in Bitcoin, there's no privacy. So every transaction, you have some public key and your transaction, you say, "Okay. I'm..." Your name is not given, but your public key is given. Your public key is associated with your name. So you say, "Okay. I, public key, this gave money to..." Whatever. Whole foods, to whatever, whoever takes the... And it's written there on the chain.

So that's a bit worrisome that all your private transactions are written in publicly on a public ledger. Now, it's not that easy to see. Because as I said, it's all public keys. It's not written what the public key corresponds to. But it's very easy to de-anonymize this. So if someone really wants to de-anonymize and understand what your transactions are, they can. Unless you work really, really, really hard to keep anonymous by each time using a different name, a different public key.

It's actually quite difficult to do correctly. But it gives this... So I think they used it to enhance privacy. They didn't get privacy but to enhance. So you're not really telling, "Am I giving it..." I'm giving it to one of them, or there's some... I think it was used to increase privacy, but I'm not a 100% sure.

Bruke Kifle: I see, I see. So I think even before we started the call, you mentioned something earlier, which is, you develop this technology and once it's out there, there are many practical applications, some of which you may not even know of or be aware of. And I think this underlies a lot of foundational research where some of the work that you put out into the world can be, whether it be research papers, whether it be technologies, can be adopted and used in many settings oftentimes that you may not be fully aware of. How does that make you feel as a researcher? Does it make you excited that some of your work is out there and individuals are finding, or organizations are finding practical applications? Does it cause some concern for you? What are your thoughts as you see the adoption, the public adoption of some of your work as a researcher?

Dr. Yael Kalai: Yeah. So, let me say first that it's interesting. My research, as you said, is very fundamental. When I got the ACM prize and they told me for my work that has had so many applications, it takes a village to do this application. It's not really... By the time it's actually applied, it's quite far removed for my work. They use my basic ideas, but there's so much more ideas and making things more efficient. There's a lot more that go in until things are actually adopted. Tons of work, more work. So it really takes a village to go from fundamental ideas all the way to deployment. How does it make me feel? I'm very, very excited. I think if I needed to say, how do I value my research, I think I would say I value it if it has an impact. Now, it may not have an impact today. It may take time. But if at the end of the day it's in some drawer, nobody uses it, what's the point?

So I would love... I love it when I see it used. And for example, with this verification delegation, now we have a big community around it. We actually have this effort called ZK proof. ZK stands for zero knowledge. Because as I said, at the end we always put zero knowledge on top, but these succinct proofs. And this effort is led by fundamental researchers all the way to people. It's really a collaborative effort. And once a year we have workshops and sometimes it's so diverse. We have bankers coming to this workshop asking this question, and it's really fun to see, because they want to use it in their banks and how do they use it. And so I really, really love it. Though I have to say, it's funny.

I used to joke when I was younger that people ask me, "So you sit all day, but what do you do? Nobody will ever use this stuff anyway because you're a mathematician. So why are you doing it?" And I remember I always joked, said, "Well, at least I'm not doing any harm. There's so many people who are doing harm. I'm at least sitting quietly and doing my research." So going back to your question, sometimes I have this feeling, worry, "Oh, I hope my work is not used."

Bruke Kifle: [inaudible 00:45:27].

Dr. Yael Kalai: Or even, I don't know. They think of there were physicists sitting and doing physics and now there's atomic bombs. I don't know how they feel about that. Or nuclear weapon, thanks to physics research. So sometimes you do fundamental research and you're like, "I hope it will not make things worse." I don't know. I think today on the LLM, there's a lot of confusion. Is this good thing? Is it a bad thing? So I have nothing to do with LLMs. My research is not related to that. But I can imagine that if I contributed to that space, I would be today... I don't know. I don't know if I'd be very happy or concerned or probably both.

Bruke Kifle: I don't think that's too much of a concern. Maybe with the exception of ring signatures, where we'll have more tattletales in the world. But you touched on this idea of your working with different stakeholders and collaborations with fundamental research, but also practical applications. So you have this very unique role where you have a post at MSR where you're a researcher, but you also have an appointment as adjunct faculty at MIT, right?

Dr. Yael Kalai: Yes.

Bruke Kifle: And I'm sure that comes with engagements at CSAIL, at the Computer Science and AI Laboratory. So how do you balance your role, this interesting role, this dual role between academia and industry, at an organization specifically like Microsoft? And I'm specifically curious, what are some of the benefits and challenges of working in both settings?

Dr. Yael Kalai: Yeah. So let me start by saying, I love working in both settings. So I've been at Microsoft for 15 years. Microsoft has been an amazing place for me to work. Throughout this time, I felt a lot of support for basic research, in particular my research. I don't know to talk about Microsoft research as a whole, because I'm not that familiar, but at least in our lab, I can say, in our lab, there was always from day one, a huge support for basic research. And I feel like I could not have done better work anywhere else. So that's fantastic for me. It's a great place to do... It's been very good for me.

I do love working with students. Students is what gets me excited to see the spark in their eyes, is really fun for me. And they're so vibrant and tons of energy and I love it. So that's something that I enjoy a lot. I do have in Microsoft, interns. And I've worked with amazing interns and I'm really proud of... I look at them, all my interns throughout these last 15 years. And I look at where they are in top academic institutions and I'm very proud. But I also really enjoy having a longer... Intern is a three months engagement. And PhD students, it's a real strong relationship for five years. And that's something that I really, really enjoy as well. So I love being at MIT. The working with the students is just so much fun. They're so vibrant and energetic and it's brilliant. And I also really love the group, the theory group at MIT, just on a personal level.

So that is great for me. And in terms of the relationship between them, I think I actually get a lot of being in both places. Because MIT is large enough that the theory group is an entire floor in the building. So I come to MIT, I go to the sixth floor, I go up in the elevator, I go to the sixth floor, and I'm just surrounded by theory people. So we can all play in our little sandbox together with our little toys, but we're all just theorists. And Microsoft, on the other hand, it's extremely diverse lab. In my floor, right next to me I have Mary Gray who is an ethnographer. I have Henry Cohen who's a mathematician. I have economists, I have social scientists. I have everything like a game theorist from all kinds of different disciplines. And now I need to explain to them actually what I work on.

We have conversations, we talk. And that's when I... It keeps me true to myself. "Am I really working on problems that I'm interested in?" Because it's very easy to convince a fellow theorist, "Oh, this is super interesting. It was an open problem in the previous paper." We all buy the same... It's very easy to sell to us, why our problems are interesting. Much harder to sell to an ethnographer. So, it really keeps you on your toes. And I think that has been really great for me. Having both. Having to interact very broadly with people in computer science and outside of computer science. Economics, social science. And at the same time, I have my little group of friends, my playmates at MIT. So, I think having both was really fantastic for me.

Bruke Kifle: I'm curious, beyond the dialogue and conversation with some of your colleagues who are in different areas of research, you described earlier how cryptography or theory of computation has a lot of philosophical questions or a lot of game theory. Do you actually find yourself collaborating on research initiatives to address some of these complex problems? Or is it primarily open discussions, coffee chats that help you frame or rethink some of the way you approach your work?

Dr. Yael Kalai: You're talking Microsoft right now. Right?

Bruke Kifle: That's correct.

Dr. Yael Kalai: Actually, so usually it's the latter. I talk to people in these kind of conversations, sometimes just water cooler conversations, but sometimes it's actually, we give talks to one another. It's more formal than that. And that's usually where I gain a lot of insight. Even though our lab is very interdisciplinary and people collaborate across various disciplines together, I am less of an interdisciplinary person, just because I tend to dive so deep into thought that it's hard for me to get up... And when you collaborate with someone outside of the field, you need to stay in the surface a little bit because they don't know. So my research is typically not... Is less collaborative in that with people outside my field. Of course, all my papers except for one during my PhD is in collaboration with other people in my area. But not so much. Actually, I have one paper with my husband and machine learning and cryptography with Shafi as well, but most of my papers are collaboration only within theory.

Bruke Kifle: I see.

Dr. Yael Kalai: Yeah. Actually the machine learning paper is also in theory, but I guess my husband, he used to be a theorist, but now he's more applied. He's my only example of someone that... But it was also happened during quarantine where collaborating with anyone else was a bit difficult.

Bruke Kifle: Harder. So it was out of necessity.

Dr. Yael Kalai: Yeah. I was like, "Okay. I guess it's you."

Bruke Kifle: I see. And then, so being at a company like Microsoft, and organizationally, I know MSR has a different charter and a different organizational structure compared to the consumer or product arm of the company. But do you find opportunities for engagements with the product side of the company? Do you find product use cases driving some of your research directions?

Dr. Yael Kalai: So I do have some engagement occasionally. But typically the way they end is by me pointing them to someone more applied. They'll ask me a question and I'll hear what they have to say. And I'm like, okay. So I think the expert you really want is so-and-so. Because again, my work is so, I'm not an expert in the application regime. But I feel like often when my involvement is more as a middle name, they hear my name, but I actually just point them to the correct person they want within research.

Bruke Kifle: I see. But at the end of the day, the fundamentals are core to the application, right? So...

Dr. Yael Kalai: Yes. The fundamentals are core. You're right.

Bruke Kifle: I see.

Dr. Yael Kalai: You're right.

Bruke Kifle: That's awesome. So I want to wrap with a couple of questions around future directions. I know I tabled this question earlier, and it was because it was probably one of the more interesting questions that I wanted to get your thoughts on. Earlier you talked about with the verifiable delegation of computation, the idea that you do not offer a guarantee. Rather this whole thing is based on the assumption that the problem is hard or it's computationally hard. However, we're seeing a lot of over the years, over the past decade, and I'm sure in the coming decade, a lot of rapid advancements in quantum computing. So with some of these advancements, there's growing concern about how it'll impact traditional cryptographic systems. But even in cases like the one that you described with the verifiable delegation of computation, these attacks on encryption methods that would normally take years because they're so hard, could now theoretically be done in days with

quantum computers. So how do you envision the future of cryptography in the face of quantum computing?

Dr. Yael Kalai: Yes. Okay. That's a very, very good question. We're now really working hard actually on upgrading cryptography to be what we call post quantum secure. So traditionally the assumptions, the hardness assumptions that we used are all known to be broken using quantum computers. So if we will have large scale quantum computers, these will be able to break our cryptographic assumptions. So that's a huge concern. Actually, there's a big initiative by NIST, which is the standardized, National Institute of Standardization, and they have a huge call on trying to upgrade all the standards to be post quantum secure. So that's something we're working on a lot, is getting everything to be secure under assumptions that we believe quantum computers cannot break. But I want to say that with that, I think quantum computers, if they will exist indeed, I mean large scale quantum computers, then it brings with it a lot of promise.

So another type of research that's happening a lot now in cryptography is, what could we do? How can we use it? Not just... So your question was, the attacker is more powerful. That attacker can use a quantum computer. Oh, we better watch out. Yes. So indeed that's a concern and we were working hard to address it. However, we the honest people, namely the people who generate all this cryptography, can also use now quantum computers. And that's a big hammer. So what can we do with it that will make our life easier? So that's also something that a lot of people are thinking about, including myself. But I want to mention one last thing about, again, the attacker having quantum power. It's much harder than actually some of our schemes, let's say, are secure. We prove our secure against assumptions that we do not know how to break. Let's say by quantum computer. The way we prove security, as soon as the adversary is classical. And it's not... In many cases, it's not clear that the proof goes through that we can upgrade the proof.

So when I say proof, let's say I have some scheme. Some signature scheme, and I prove to you it's secure. Namely, if the adversary cannot break the assumption, then he cannot fake a signature. The way I prove this to you, my proof strike. The way I argue security, so far assume that the adversary is classical. If the adversary is quantum, some of our proof techniques fail. So for example, sometimes the way we argue say, "Well, if there's an adversary that succeeds in generating signatures, I will run him again and again and again and again to generate many signatures, and then I will use that to break the assumption. But a quantum adversary, you can't run more than once. He measures a state, a state collapses. These weird quantum phenomena happen in the quantum world that do not happen in the classical world. So the situation is much more difficult than just upgrading the assumption to be post quantum secure.

We need to upgrade our proof, our guarantees, like how we prove security to be post quantum secure. It's quite the challenge. And we, myself and the many,

many others are working really hard to get there. But there's a lot of progress in this space right now in cryptography, and we're making fast progress.

Bruke Kifle: Well, yeah. As quantum computing continues to rapidly advance, like you said, it seems like there will be a core requirement for some of the underlying assumptions, proofs and cryptographic systems to also equally make the same progress or else we might be in trouble.

Dr. Yael Kalai: Exactly.

Bruke Kifle: So just as a closing question, I'm curious, what are some emerging or exciting areas that you believe will shape the future, both in cryptography, in theory, but also the field of computing more broadly? What keeping you up at night? What's exciting you these days?

Dr. Yael Kalai: Yeah, so look, what's really exciting me as well as I think exciting the entire world is with these LLMs are just unbelievable. And we need to... This raises so many challenges, as I mentioned before. And I think, thinking forward, we need to think of verification in the setting of LLMs, a large language models. I think that's the new kid in the block that's creating a lot of noise, and it's really, it's a revolution really. Sometimes I feel like I can't believe I'm alive to see this. It's super exciting and I think there's a lot of challenges that we as a community need to solve. Of course, I have a lot of problems that I'm interested in, that I was obsessed with and still obsessed with before the LLM came along. So I am not dropping them, but this is kind of... If you ask me to step back from my own obsession and look at the world and see where it's going, that's definitely I think where we need to focus our attention on trying to ensure security of these large language model.

How do we ensure that they're doing what they're supposed to? How do we generate... How do we get some verification from them? How do we instill some trust in these systems?

Bruke Kifle: Certainly, I think it's imperative. It's a groundbreaking technology, and it's changing the way everything is done in this world. So researchers, organizations are pivoting their strategy in many ways to be AI or LLM focused in many ways. And I think it's the right thing to do because the technology's here and it's here to stay.

Dr. Yael Kalai: Exactly.

Bruke Kifle: So just to wrap up, we have a lot of listeners who may be early in career, who may be students or who may be computing professionals who are looking to explore a new area of computing. So generally, what nugget or bite or advice would you give to these folks interested in pursuing a career in computing and research more broadly?

Dr. Yael Kalai: Yeah. So the truth is, my advice is find something that keeps you up at night. I think I've worked with so many students over the years, and I see the difference between those who are successful, very successful, and those who are less. And the difference people think, "Oh, it's because they're so much smarter." Actually, I'm not sure that's the case. I think the successful people are those who found their passion, who found something they really want to solve. They're so excited by it. And what I would encourage anyone, my kids, students, anyone, any human being is really find something, find your passion inside your work. First, it's fun. It'll make your life so colorful and fun and engaging. And second, I believe that that's the way to success, to do something that you really, really interested in. And yes, sometimes it does require a little bit of pivoting.

Look, I was interested in fundamental mathematics and theoretical math, and I pivoted a little bit. I'm now in computer science, so I'm not saying just no matter what, you want something and you're going to just something, but don't lose your passion. Find something that you enjoy, that you're passionate about, that you wake up in the morning excited to do. That would be my... And I think if you find that, you're golden. From here, success is just, it will follow.

Bruke Kifle: I think that's a very great piece of advice and certainly a great principle to live by. Find what you love, do what you love and you'll never have to work a day in your life.

Dr. Yael Kalai: Exactly.

Bruke Kifle: Well, Dr. Yael Kalai, thank you so, so much for joining us on ByteCast and looking forward to some of the amazing work you will continue to do.

Dr. Yael Kalai: Thank you so much. Thank you for having me.

Bruke Kifle: ACM ByteCast is a production of the Association for Computing Machineries Practitioner Board. To learn more about ACM and its activities, visit acm.org. For more information about this and other episodes, please visit our website at learning.acm.org/bytecast. That's learning.acm.org/bytecast.