Hello, and welcome to the CSAIL Alliance Podcast series. My name is Steve Lewis, I'm the Assistant Director for Global Strategic Alliances at MIT'S Computer Science and Artificial Intelligence Lab, better known as CSAIL.

In this podcast series, I will interview principal researchers at CSAIL to discover what they're working on and how it will impact society.

Today in our podcast I'll be speaking to Professor Yael Kalai. Yael is a cryptographer and theoretical computer scientist who works as a Senior Principal Researcher at Microsoft Research New England and as an Adjunct Professor at MIT in the Computer Science and Artificial Intelligence Lab.

Yael graduated from Hebrew University of Jerusalem and worked with Adi Shamir at the Weizmann Institute of Science, earning a master's degree in 2001. She completed her PhD at MIT in 2006. Yael did postdoctoral studies at Microsoft Research in the Weizmann Institute before becoming a faculty member at the Georgia Institute of Technology. She took a permanent position at Microsoft Research in 2008.

Professor Kalai is known for co-inventing ring signatures, which have become a key component of numerous systems such as Cryptonote and Monero cryptocurrency. Her master's thesis introducing ring signatures won an outstanding master's thesis award. Her MIT PhD dissertation was awarded the George M. Sprowls Award for Outstanding PhD Thesis in Computer Science.

Yael, thank you for joining us today.

Thank you for having me.

Can you explain to our listeners the focus of your research and some of your aspirations?

Sure. So my research is mainly focused on the field of cryptography. And it's very foundational and theoretical in nature, though I'm really interested in real-world applications. So people-- I think often, when you say the word cryptography, people think of just how do you send a message securely so people can't see what's written there.

But actually, the field of cryptography has advanced immensely. And it's way beyond-- what we're thinking about is way beyond the kind of securing communication. And it's more now in the realm of securing computation.

And our world is shifting, like the way where we store our data often is no longer in our own private machine, we store it in some cloud servers. Often we don't even do our own computation, this is done elsewhere in various cloud services.

And this raises a lot of new questions and concerns, both that of privacy, which is, of course, the bread and butter of cryptography, but also authenticity. How do we know that what the cloud servers is telling us they're doing? How do we know that they're actually doing what they're supposed to? And how do we know that when we send computation somewhere that the actual computation is that we request that it's being done? And so on.

And even though this question on its own does not seem related to cryptography, because it's a question of just integrity, whereas privacy here turns out that this is very much a cryptographic question. Or we solve it. We know how to solve it using cryptographic tools, I should say. So this is kind of the type of research that I'm doing.

In terms of aspiration, I think the aspiration for myself, and actually for my field at large, is to try to address these new challenges as they arise. And every day we have new challenges. For example, COVID-19 brought a bunch of challenges with it, much of which are cryptographic, which I've been involved in an effort here at MIT, which is led by Professor Ron Rivest and Danny Weitzner. And so challenges come to us every day, and my aspiration is that we'll be able to address it.

And as I said my focus is theoretical, but I my goal or my aspiration is to address these questions by modeling them truth to reality. You know you want the modeling to be correct. You want to ask the correct questions. And then my solution are typically kind of the most fundamental one and the hope is that people will make them more efficient and optimize, and so on and so forth.

I see. So getting to the question of how do we know, does that beg the need for these proofs, for example? So can you explain at a high level, the difference between interactive proofs versus classical proofs. And are proofs what is the answer to how we know?

Good, good, good. So proofs, yeah. So actually the concept of a proof really evolved in the last 30 years or so. You know those who are not in my field, typically when we think of proofs you think of just a written what you did in calculus in high school, where you just write this implies this implies, this is just a sheet of paper with kind of axioms and implications. And this is the way actually people thought about proofs for thousands of years.

But in crypto, we view proofs in a different way. And because this classic way of thinking of proofs

tend to be very rigid and not very powerful. So in cryptography, for example speak of zero-knowledge proof-- you know, so in the '80s Shafi Goldwasser MIT and Silvio Micali from MIT and Rackoff, they were interested in what's called zero-knowledge proofs.

Zero-knowledge, it's nothing to do with interaction or anything. You know at that point, we only had classical proofs. But what they wanted is they wanted proof to reveal no information. You prove something so I can prove to you-- So I can prove to you, let's say that these two pens are identical, or sorry, are different, but I don't want to tell you why they're different. From your eyes, they're identical. I see a difference. I want to convince you that they're different, but I don't want you to know where the difference lies.

And the truth is, we don't know how to do that using the classical proof. That's just, not only we don't know, it's impossible. Because any kind of proof that's written, you see some information that you can kind of carry that is a certificate showing that something is different between these two pens. So this is a test that's impossible using classical proof. And that's what led them to think about-- to define the notion of interactive proofs. Turns out that with interactive proofs, it is possible.

OK so the idea of-- so what is interactive proof? It's a process between you and me. So for example, going back to my two pens. I can, for example, I can tell you, look I know they're different, don't want to tell you why. OK I tell you my left is A, my right is B. Now I give you the two pens. You switch the order, don't switch the order, I'm blindfolded I can't see. Then you give me the two pens and I can tell you which is A and which is B. If we succeed in this experiment many times, you're convinced that there's a difference, but you have no idea what the difference is. Because you knew if you switched or not-- so by me telling you if you switched or not, you didn't gain any additional information.

So this is kind of-- so what they noticed is that the interactive proofs, when we interact with each other, yes so I give you the pens, you change, you don't change, you give me back the pen. It's some kind of interactive process. It allows for zero-knowledge. It allows to give zero-knowledge. But later, what was also noticed is that this is a much more powerful proof system, namely, that we can actually prove-- make proofs much more succinct, much shorter, much more easily verifiable.

So it gives you two things. It gives you-- hides zero-knowledge or it hides information, what cryptography is very good at doing. But the other thing it's doing, which is not often thought of as a cryptographic thing, is it allows for much more efficient verification. So if before, to argue, to give you a proof of something, you would need to read 100 pages. Now you need to read very, very little kind of communication, but you need to interact very, very efficiently. So it's been proven very useful, this

idea of interactive proofs.

And would you say this is useful for cryptocurrency, or IoT devices like smartwatches?

It's useful in many, many applications. Cryptocurrency is definitely one of them, as we talked about the world evolving and changing. This is again one place where cryptography kind of plays a significant role, not only in the beginning using like the basic hash function, signatures, which were kind of the original Bitcoin kind of ideas. But also now as you said, a lot of cryptocurrencies use these zero-knowledge proofs. And why they're so useful there is because they want to ensure that things are private.

So the problem with these cryptocurrencies is that they're public ledgers, and these public ledger stores-- all are transactions, and that's how it works. That's kind of what cryptocurrencies are, it's a public ledger that stores the transactions. The problem is it stores the transaction where you pay and who you pay to is suppose-- people really care about that part being secret. You don't want to post in a public ledger. But using cryptocurrencies, that's what's done. It's posted on a public ledger.

Now people can say, OK, it's not really public because you're not associated with your name, you're associated with some public key. So nobody knows that Steve is the holder of this public key. But these things, it's anonymized, but we all know that this anonymization-- said the word right? It's not really hides because you can trace information.

So for example, if I know, OK this public key bought I don't know, went to the drugstore next to your house and then this public key also went and paid for a soccer team which your son goes to, and I know-- eventually, I know it's you. And then I'll be able to trace all the other things you did. You know, you paid-- so and of course you can generate many public key so maybe to hide, but usually it's a very difficult thing. And often we can be anonymized.

So getting secrecy in cryptocurrencies is important. And there are several cryptocurrency companies, some of which started here at MIT and which actually use these zero-knowledge proofs. So the idea is, you put your things kind of, you use commitment so you hide, you kind of-- instead of putting in the clear your transaction, you hide it, and but you also add kind of a zero-knowledge proof. That this is a valid transaction and that things are good.

But I want to say another thing beyond hiding in zero-knowledge. Adding these succinct proofs, even if you don't care about privacy, helps a lot in efficiency.

So currently, the way, for example, Bitcoin works is if you want to pay me with your Bitcoin, you tell

me, OK I'm going to pay with this, you see this Bitcoin up there, it's mine. I'm going to give you a Bitcoin. It's with some number. This is my Bitcoin. Now how do I know that it's a valid Bitcoin? So I need to-- you tell me, oh I got it from Bob. OK so I go to Bob and I check did Bob give him, and was this not double spent, maybe it was double spent. So I need to check. Now where did Bob get it? How do I know that? So I need, Oh from Charlie. OK so I took all the way kind of to Genesis to make sure that it's valid. This does not scale.

And everybody needs to verify. Every time you kind of, people need to verify. So instead, if I just add a little certificate, like a succinct proof that this is a good proof, then that's it. So that's what's done.

I just want to say one more thing. Which is interesting, because you said, I started by saying classical proofs are too long. And the nice thing you can do an interactive proof. And now I'm saying this interactive proof is great, you can put it on the chain or the blockchain. But at this point, I think our listeners, and maybe you, Steve, as well should be like, wait, who are you interacting with? It's a chain, so you're putting proof, but you're talking about interactive proof. And this is just a chain. Where is the interaction going?

And the answer to that question is actually we make the interactive using cryptography, and this is where kind of the missing piece is. Using cryptography, we can make this interactive proofs non-interactive. We can go back to the original, kind of non-interactive proofs at a price. And the price is that we have our guarantee is that only a computationally-bounded cheating prover cannot cheat. So you can think of it, the only way you can cheat is by solving a really, really hard mathematical problem. So solving factoring, you can factor two really large numbers, which is believed to be really, really hard, or something like that.

So essentially at the end of the day, we have succinct, very short non-interactive proofs, but the guarantee is only against computationally-bounded cheating provers. Whereas traditional proofs, the guarantee is just a false proof doesn't exist. It's not about hard to find. Here, false proofs for false statements do exist, they're just impossible from our perspective. From real-world perspective, quote unquote "impossible to find," but they do exist.

And you believe that that has been the most innovative thing to happen with cryptography in the last decade?

So I definitely think this is one of the pillars. I'm super excited about this direction and it's wonderful to see how industry is excited as well. We've had conferences, workshops around this where we had

such a wide range of participants from academics, pure kind of basic researchers like myself, and bankers that want to use these ideas in their banking infrastructure. So this is definitely one of the pillars.

But I would not say the only one. For example, one of the huge successes of cryptography in the last decade, which probably started a decade ago, is the notion of fully homomorphic encryption. And what it does, it allows us to encrypt a message, but then allows us to do computation on this message underneath the encryption. So we can compute on it without actually knowing anything about the computation itself.

And this has been a very successful huge line of work, started actually with a student-- work of a student, Craig Gentry, who was then a student at Stanford. Now he's involved with cryptocurrencies. And so that's been a huge line of work. One of done--

A major participant in this kind of-- major players Professor Vinod Vaikuntanathan here from MIT, who's done a lot of work and has been pushing it. Also Shafi Goldwasser, Professor Shafi Goldwasser as well, from MIT. They've been pushing on it kind of forward quite a bit.

And it's very useful, for example, for health care where we really are concerned about our data, but we want to do some statistics and learn information without-- so you want to do things under the hood of an encryption scheme. So that's another one.

And maybe the new kid on the block now, there's a huge kind of excitement now in the crypto community is we have kind of a very really great breakthrough result in the problem of obfuscation. Which is, how do you take a program, and make it completely unintelligible so people can't reverse engineer or learn things, but can still run the program. So this is called program obfuscation. And this is again, I think, once we make it practical, it's going to be extremely useful. So far we didn't even have any-- like until very recently, we didn't have any even heuristics for doing that.

And the last five years, maybe more, eight years by now, maybe from 2014, so seven years-- has been kind of a huge boom in this field with a lot of really nice breakthrough results, and very interesting math kind of coming with it. So that's also very exciting. And there's a lot more, of course.

That there would be exciting. I think that the application would be for protecting intellectual property right for software programs so that--

Exactly.

Can you tell us what excites you most about your research in your field?

The problems of are very kind of nice and you know well-dressed and look very beautiful. But at the end of the day, what we're actually solving are fundamental, not questions. At the end of the day, you sit with a math problem trying to solve it. We reduce it, or we model it, it's a math problem that we, at the end of the try to solve.

I really enjoyed basic math and I love solving this problem, especially because there's a reason to solve them, they're not just for the sake of math. But I think one thing that really excites me is actually the people in my community. I think the community of theoretical cryptography is a great community. We have awesome people and we're very collaborative. We work a lot together and it's just really, really fun. We work together, we try to--

I think one thing that really excites me actually, is to work with the students and see the spark in their eyes when they have a result. And it's great, it's really fun. And I definitely think beyond just the beautiful math and the great applications, I would put a lot of the fun and credit to the folks that work with me, mainly my students.

And speaking of community, can you tell us a little bit about your work at Microsoft Research?

Sure, yeah. So yeah, so I'm a full-time employee Microsoft Research and I'm an adjunct professor here at MIT. And interestingly, my work there and here is really the same. At the end of the day, my work is to advance the state of the art and try to have an impact like all of us. We all do kind of the same work, just different angles to it. It's-- I find it really, really remarkable how this engagement between MIT and MSR Microsoft Research is going so well. I think it's really, really valued by both places, which is really great for me. It makes me very happy.

I think Microsoft Research values the fact that I'm an adjunct at MIT. It values the collaboration. It values our connection. And when I started, I was worried that Microsoft Research will not be happy that I'm spending a lot of time at MIT, or that MIT is taking me away from Microsoft, or anything like that. Or worried vice versa with a MIT that this-- but actually, I feel like it's just a win-win. I think both institutions view it that way.

I think Microsoft is very excited about the collaboration. They're happy to have their student, our students at MIT being exposed to Microsoft and learn about Microsoft. They're happy with me teaching at MIT. And it's just been really great.

Our students here at MIT come-- well, used to, a year ago-- to come down the street and visit us at

Microsoft all the time. Not only the students, also the faculty from MIT, and actually elsewhere also in the neighborhood and outside the neighborhood for long-term visitors.

So we've been having actually, with MIT in particular though, a very good collaboration, co-organized seminars and workshops, and crypto days, and so on. So I feel like this collaboration between Microsoft and MIT has been wonderful.

And as I said, in terms of my personal work there's not that much difference. I do the same thing beyond just specific things like hiring there, committees here, students here. But beyond kind of just the bureaucracy type work, my basic work, the work. I'm excited about most is done in both. It's kind of the same work, which is just advance the state of the art.

I see. Can you talk a little bit about what you've been doing with Ron Rivest and COVID-19, and I assume you have to do with contact tracing and--

Yeah, yeah, yeah. So this is-- exactly. So this is a big initiative, not just-- it's led by Danny and Ron Rivest, but it's actually a big initiative with many people involved. It's been a really interesting and great journey. So our work in particular-- the thing I was involved in, it's much bigger than that. But the thing I was involved in is indeed the digital contact tracing, and how-- what is the best algorithm to hide.

So what we want, we want to tell someone when they were close and when they were in contact with someone that was tested positive, but we don't want to reveal who that person is. And that doing it-- actually doing it both digitally, which is what we're trying to do, it's an automated contact tracing, and even manually, actually.

It's challenging thing, because I want to tell you, you were in contact. Now it'll be good for you to know, for example, a little bit when you were in contact, like yesterday. But if you just met one person yesterday, then you may actually know who you were in contact, you know who the positive person was. So it's a challenge to try to do it in a way that hides information.

And digitally becomes much, much harder, because for example, if you know exactly when you were in contact, then it's you may get a lot of information about the person if you know it's exactly a 2:00 PM yesterday when I was sitting in a cafe with my friend, you may know who the positive person was.

But even more than that, a much bigger problem, how do we know who you were in contact with? The reason the way we do it is we have your device, your cell phone in particular, send little, what we call

chirps around. And then when you tell us, Oh I was tested positive, we tell everyone, Oh listen these chirps that you heard-- that people hear chirps all the time, that's kind of our idea. These chirps that you heard is from someone who tested positive.

Now, the basic thing you want to ensure is that chirps that people hear are not traceable. They don't know it's the same person, if you walk around with the same chirp all the time. So maybe not saying Steve, Steve, Steve, but you know, whatever. Just a chirp with random numbers, but it's the same number all the time. Again I'll be able to trace you. Now I see the same person walking around, I see him going to MIT in the morning, I see them-- I'll know it's you.

It's this kind of anonymization, which does not work, because we can trace. So we don't want to have the same chirp that these chirps need to be kind of changing all the time. But how do you do it in a private manner and so on? There is some challenges to that.

And here in this project what was new to me, which is something I haven't dealt with before, I think mainly due to the theoretical nature of my work, is it's not just a technical problem. There is a psychological problem. So when you tell people you should use this, it needs to be simple enough for them to understand what they're using.

It needs to be, for example, are you allowed to use location data? What if we use it completely privately, like there's no way the cell phone will ever kind of reveal any location data. Is that OK? Turns out that Apple, Google, no way, we don't want to use it, because it's bad publicity. Even if you ensure to us that nothing is going to be leaked, we don't want to use it. We don't want to tell people that we're using their location data.

These kind of things are things that I'm not used to-- they are not constraints that I'm used to thinking about them. If it doesn't leak, it doesn't leak. But even that is a problem, because now you can say, OK it doesn't leak. You're using it in a way that doesn't leak. Maybe someone else will use it in a way that will leak in a different context.

What we're saving-- you know it's kind of a bigger problem than just doing the contact tracing itself. Like putting in perspective and how people think about it, and how Apple, Google think about the publicity that they get from it. It's like a lot of things came into play while working on these things, which are things that I'm not used to thinking about. It was really fun.

And also the collaboration between policy people, and technical people, and just health work people, contact tracers, and just a whole kind of new collaboration. Very diverse set of people was really

interesting.

Yeah, it seems like it's a hard social engineering problem to solve, right? Versus a technical or software engineering problem to solve.

Exactly. Exactly. Precisely.

Excellent. Well, Yael thank you very much for your time today. It was fascinating. It was great to talk to you. And we hope you continue your great research at Microsoft. And we'll see you around CSAIL when we get back on campus.

Yes. Thank you so much.

Have a great day.

Have a great day. Bye.

Bye.