Professor Ross Anderson

Interviewed by

Elisabetta Mori

12th March 2024

By Zoom

Welcome to the Archives of Information Technology. It's 12th March 2024, I am Elisabetta Mori, an interviewer with the Archives of IT. Today I'll be talking to Professor Ross Anderson, we are on Zoom, I am in Livorno, Italy and Professor Anderson is in Cambridge, UK. Ross Anderson is Professor of Security Engineering at Cambridge and at Edinburgh Universities. A pioneer in his field, he has devoted his career to developing security engineering as a discipline, building systems to remain dependable in the face of malice, errors or mischance. He received important distinctions and awards, he is a Fellow of the Royal Society and the Royal Academy of Engineering, a Fellow of the Institute of Physics, as well as the recipient of the 2015 Lovelace Medal, awarded by the British Computer Society.

Welcome, Ross.

Thanks.

Thank you, thank you for being here today. So let's start with where, when were you born?

Well, I'm told I was born in 1956 in the maternity hospital in Birkenhead, although my memory doesn't stretch that far back. We spent the first five years of my life in Wallasey, near Liverpool, because my dad was working as a research director at a vaccines firm near Speke. Then when I was five we moved up to Scotland where my parents are from and where we lived thereafter. We lived until I was eleven near a mining village called Annathill, which has now been demolished. It was four miles from Coatbridge and three miles from Airdrie, and [00:01:50 sound drops out] initially in Coatbridge on the bus, and then from aged eight I went into the High School of Glasgow.

Can you describe your parents?

My dad was a research pharmacist, he started off working for a drug company and he was busy developing ulcer drugs and working on oligosaccharides. And my mum was also a chemist. While I was very young she worked as a locum in a hospital and then when we got a bit older she got herself a chemist's job, after we had moved to

Gourock in the west of Scotland when I was eleven. But while we were still near she was working in the local hospital.

Have you got any siblings?

I've got one brother who's four years younger than me and who ended up becoming a surgeon, and he lives in Bolton, so kind of halfway between where I am in Cambridge and where my parents were in Gourock.

So what were the important influences on you in your early life?

Well, I first got interested in electronics when I was eleven or twelve because we had moved from Lanarkshire to Gourock on the west coast because it was no longer possible to get into good schools in Lanarkshire, so it wouldn't have been possible for my brother, Ian, to follow me to the High School of Glasgow, and so my parents moved so that he could go to Greenock Academy when he was a little bit older. And from Gourock my dad commuted into work at Strathclyde University. By then he had got a job as an academic and he was still consulting for Glaxo, a drug company, so we used to meet Glaxo people and went to France occasionally on holiday to borrow one of their grand country houses, but from the age of eleven I was living in Gourock. Now, one of the first things I did when we moved there was to join the Boy Scouts, because the kids there who I befriended were the Scouts and the Scouts had an amateur radio club which was run on Friday evenings by a chap called Ian Simpson, who was an engineer at IBM. IBM had a facility in the Spango Valley which was just three miles away from Gourock over the hill, and so that's where I first figured out what transistors were and, you know, what shortwave listening is about, and you would go about building simple analogue circuits.

[00:04:32]

Did this have any relations with what you chose to study later on in your life?

Well, I was supposed to become a doctor, because many people in my family on both sides had been medical. You know, my uncle was a GP, and so on and so forth. But

when I was sixteen I was in the local library in Gourock and I came across a book by… about… from the turn of the last century, by Felix Klein, called *Elementary Mathematics from an Advanced Standpoint*. And this was a series of lectures that he'd given in Göttingen to maths graduates who were planning to become maths teachers, explaining how the kind of maths that they'd studied at university could be used to motivate and inspire schoolkids. And I was just hooked and I decided that I wasn't going to be a doctor after all, I was going to be a mathematician. And this really annoyed my dad because, you know, he would say, 'Mathematician, mathematician? Why don't you keep maths as a hobby like your grandfather Jack and be a doctor, that way you can be sure you'll earn a decent amount of money. How on earth will you feed your family as a mathematician?' And of course, this isn't the way that you persuade a sixteen year old of anything, and so when I was seventeen I went up to Glasgow University and I'd applied to study medicine and I'd got a place, and I'd also applied to study science as an insurance offer in case I screwed up my exams, and so when I arrived at Glasgow I started doing the science course and it took the medics a whole term to realise that I was doing science rather than medicine, because it was beyond their experience that someone who had admitted doing medicine would study another subject instead. But because the top four or five of us went off and did different interesting things, and then there was about twelve or fifteen who studied medicine, and then there was a couple of vets, and then there were twenty who studied law, more or less in strict rank order of how good their exam results had been. So of those of us who did particularly well in school there was another youngster who went out, who went to Glasgow University and then went on to Cambridge, and there was another who went to Oxford, there was one who went to London and studied Chinese, there was another who studied Russian and, you know, thereafter it was just the serried ranks of medics and lawyers. So that was my experience as a youngster. I mean, before I came to Felix Klein's book, I considered maths to be pretty boring, because I was just good at it, I'd get the new term's maths books and I'd check that I could understand it all and do the hard problems at the back and just toss it into my desk and forget about it, but you know, suddenly that got me going.

And so what led you to attend Trinity College at University of Cambridge?

Well, in Scotland you, if you're bright and you do well at your Highers you can leave school at sixteen and go to university and the degree courses there are four years long rather than three. If you wanted in those days to go to Cambridge, one of the paths to do that was to stay on at school and do an extra year, but another was to do a first year at a Scottish university and then swap. Now, when I got to Glasgow University I realised within a month or so that all the professors had been lecturers at Cambridge and all lecturers had been PhD students at Cambridge, so it became obvious to me that I'd kind of gone to the wrong place. So I filled in the application forms for Cambridge, I went down just before Christmas for an interview, I got an offer of a place at Trinity, and I swapped to Trinity in October 1974 when I had just turned eighteen.

So what are your memories of Cambridge?

Gosh. Well, as a youngster I was way out on the Asperger's spectrum, that phraseology wasn't known at the time, because British psychologists didn't discover the work of Herr Asperger until the 1980s, but I was a bit of a fish out of water at school, you know, I wasn't particularly sociable, I was good at school work, I wore glasses, so I got bullied, and it was a great relief to get to Trinity because among Trinity mathematicians I was in my element. You know, there was a whole bunch of people who thought and behaved and socialised just like me. So that was the liberation.

[00:09:24]

So you arrived there in October 1974, and then what happens?

Well, I did Part IB in my first year, which was perhaps a bit too ambitious. Cambridge sometimes encourages bright kids to skip a year and get straight into the meat of it, but as a result I had a hell of a hard work in my first year and at the end of my second year when I'd finished Part II, that is the, you know, the full undergraduate degree, I was feeling a bit burned out. And so I didn't feel like doing Part III, which is like a maths MPhil, because I didn't reckon I would be able to hack it. And the other options would have been to do a diploma in computer science, which I thought

of seriously and which a number of my friends did. But again, I tended to see computer programming as something that was kind of easy and, you know, I was filled with all sorts of questions about, you know, the meaning of life, the universe and everything, and so I decided to spend my final year at Cambridge studying history and philosophy of science. And so that was a complete gear change. I should mention that I'd first learned to program at the Glasgow Schools Computer Centre, because of the influence of our maths teacher, Willy Wilson, who arranged for us to be able to go there one afternoon a week and write programs in Fortan on an old 1104 IBM mainframe. And we had to do that in punch cards, and so it was very much the technology of the 1960s.

Yes.

But still it was fun. And then when I got to Cambridge, part of the IB maths Tripos was learning to program in FOCAL, a local language that was a kind of mash-up of Fortran and ALGOL, and so I duly did a numerical analysis in FOCAL and solved differential equations and so on, but it didn't occur to me at the time that I would want to go and actually make a career in computing. Now, with hindsight, the computer industry ended up devouring me and almost all of my contemporaries, whether they'd been studying maths or physics or geology or computer science or whatever, because it's just where all the jobs were in the 1970s and in the 1980s for kids who had some idea how to program, you could just walk into a well-paid job, and if you didn't like it, you could walk down the road and get a better job for more money. So, that was an enormous attractor. But I didn't go into the computer industry immediately, I wanted to go out and see the world a bit. Now, while I'd been at Cambridge, I'd been in the habit of going abroad to Europe in the summer and busking because I'd discovered that if I, you know, played music on the street in France or Germany or the Netherlands with my bagpipes I would earn as much in an hour as I would earn in a week working in a quarry. And so my habit was to go over to the Continent and spend a few weeks touring around playing here, playing there, staying in youth hostels and cheap hotels and on people's sofas, and saving up a thousand quid or so, which in those days was an awful lot of money. Those days you were only allowed to take fifty... [00:12:53 sound drops out]. So after I had finished my degree, I went and toured round Germany and France for six, eight weeks, I saved up a thousand quid,

and then I left my pipes at home and I set off, as I thought, you know, on the hippy trail to India. Now, I got as far as Istanbul and there started to be a revolution in Iran, and it didn't look like a very good idea to try and go through Iran when everybody's rioting and the police are shooting at them. So I spent three or four months just wandering round the coast of Turkey, and I ended up in Aleppo in Syria, just as the Shah fell, and then I went down to Damascus and the little hotel that I was staying in was full of rich Persians who had just basically fled the country and were waiting around trying to find American visas. So it was all very historical and dramatic. And so I thought, well, okay, so what do I do? So I knew someone who was living and working in Cairo, so I thought why don't I go there next just to have a look around, so I went back to Greece and then got a cheap flight to Cairo, and I stayed with my friend for a month, and then I went down through Sudan and Yemen and Saudi and Jordan, and Israel and Jordan, and Syria and Turkey, and back home. So that's basically how I wasted a year of my life, just wandering around being a tourist.

[00:14:29]

So, before you had this gap year, you also worked for Ferranti as a development engineer. Is it correct?

Yeah, I worked for Ferranti for a year, because it was set up by a friend of my father, Roy Tate, who was a senior person in their inertial systems division in Edinburgh, and he had an interesting project, which was that he'd been responsible for developing the inertial navigation set for the Tornado, and he'd the idea that you could adapt this so that it would be useful in midget submarines in the North Sea. So that was my work for a year, it was messing around with analogue to digital converters and Kalman filters and so on, and in the process, you know, I did an IEE qualification in computer engineering by private study. So, you know, that enabled me to join the IEE as an associate member and, you know, got me a foot on the commercial ladder. And technically, working for Ferranti was fun, but I wasn't very impressed with the way things were set up. And in fact Ferranti went bust while I was there and had to be bailed out by the government. The corporate structure was very oldy-worldy, the engineers were all at the bottom and the engineers' pay scale went up to about where all the salesmen's pay scale started, and I'd also got to know one of the members of

the Ferranti family at Trinity where he was an undergraduate, and I thought well, this is a rum set-up, because, you know, the engineers are like the black man on the plantation here. You know, you got some fancy tools to play with, but the pay is lousy and the promotion prospects are worse, so I got kind of disillusioned with the idea of working for a defence contractor.

Okay, so let's go back to your gap year. So in 1980 at some point, you're back. So, what did you do?

I hung around in London for five or six years. I went home and I'd realised there was nothing much for me in Gourock, so I went down to London and I did for the first year or two just various odd jobs. I did all sorts of things, you know, worked as an ad salesman for a book, I worked as an ad salesman for a newspaper, I worked for a typesetting company on foreign language typesetting. You know, I did this and did that. And then in 1982 out comes the Sinclair Spectrum, so I got one and I started writing software for it. And among other things, I eventually wrote some cryptography software because we just had the early beginnings of email there with things like Prestel, Starlink, CompuServe and so on, it was nothing like as good as today because SMTP email had not been invented to pull all the proprietary email systems together, and so what we needed was something that would take a file and encrypt it in such a way that it would go through Prestel or go through CompuServe or whatever, which meant that it was quite fiddly ASCII-armouring it in an appropriate way. So I had got interested in this because one of my mates from Trinity who was living very close by in London was working as a developer for an estate agency, and the estate agency said we would like an email encryption system so that partners can send messages to each other that their secretaries won't be able to read. And so I said, 'Well, what did you do?' And he said, 'Well, I just called the random number generator again and again and again and XOR'ed it with each byte of the file'. And I thought about that and I said, I suspect that's not very secure, but would have to look into it. So I started digging into it, and I realised that the linear congruential generator that was used for these systems was reasonably easy to reconstruct if you could guess, you know, just a few bytes of plain text. So, I started looking into cryptography and I started reading such of the research literature as was available and there was a book came out by Beker and Piper, called *Cipher Systems*,

which I got my hand on, that was just freshly published, the first textbook on cryptography in English since about the 17th century. And they were proposing a particular type of stream cipher and I suddenly realised that I knew how to solve this. So I wrote a paper on it and sent it off to *Cryptologia*, and in the meantime, a mate of mine, Keith Lockstone and I, sat down and figured out how to produce a better stream cipher. And so we then produced this email software which we managed to sell to one or two people, and at that point in early 1986, I got approached by recruiters for Barclays because they were looking for somebody who knew anything about cryptography and, you know, I was the person, and so I got hired and spent three years with them looking after the security of cash machines and funds to funds transfers and things like that. So that was an experience of a different type of large corporate organisation, you know, not a defence contractor. But Barclays is in some ways a bit like the civil service, only better paid. So I began to understand a bit how bureaucracies work, you know, the twelve layers of managers that sit between the serfs who do the work and the big guy in the big office, and all the games that people play and how these cause stuff to go wrong. So this was, if you like, psychological preparation for doing work on economics of security a dozen years later. But that was, at that time that was still all in the future.

[00:20:52]

And you worked with them for two years, right?

Mm. And after that, I thought, I got assailed by wanderlust, and I went out to Hong Kong because I'd never been there, and thought I'd look around, and so when I was there I spoke to HSBC and Standard Chartered and Standard Chartered also wanted to hire a cryptographer, so I spent some time there designing their security infrastructure and architecture for all their branches in the Far East. So that was, if you like, a little bit of a step up, it was a higher-level job and better paid. And they wanted to hire me, but I [dis?]liked living in Hong Kong because it's just so incredibly cramped, you know, apartments are so expensive and so tiny and their pavements are double-decker and, you know, the pace of work is frenetic and it's very hot and sticky. So although I stuck with it for a few months, I reckoned I probably didn't want to live there long term. Now, one of my cousins had in fact grown up there because his dad had been

an expatriate there, so that was my introduction to Hong Kong circles, but the Hong Kong expatriate lifestyle I reckoned wasn't for me. So, I then got involved in a project to invent prepayment electricity meters for ESCOM, the Electricity Supply Commission of South Africa. It was clear by then that power was going to pass to Nelson Mandela, so what ESCOM was going to have to do was to electrify millions and millions of informal dwellings in the townships, and they started a crash programme to design and build the technology for this. And a guy, Johan Bezuidenhout, who was running that programme, and he got me engaged in it. And so we designed a mechanism whereby you can sell electricity by entering a twenty-digit number into a meter and the lights will come on. And this involved then the hierarchy of vending machines. It uses cryptography in the sense that you have got a twenty-bit number, that's sixty-six bits, so you've got sixty-four bits of cipher text encrypted with a block cipher, and you've then got two bits of plain text, into these sixty-six bits you've got to shoehorn an entire instruction set, such as change the tariff from *x* to *y*, or, you know, dispense so many kilowatt hours, or whatever. And there's an interesting point in that in the near future there's going to be a flag day because the counter is going to roll over, and this means that all the meters in the world – and there's now a hundred million of them in a hundred countries supplied by a hundred vendors – you know, are going to have to have the counter updated by putting in two special tokens which will reset it to zero. And as this process is about fifty per cent complete in South Africa, so that was an engineering lesson learned. Had we thought carefully about it then, we could have decided to make the time clock in the electricity meters eight seconds rather than one second, and then instead of the clock rolling over after thirty-odd years, it would be a couple of centuries and nothing anybody could worry about. But back then we were really pressed for space and we just didn't believe that the meters would still be around after thirty years. So that is the electricity industry's equivalent of Y2K and we propose to write a paper on it some time in the near future with old colleagues from back then with all the lessons that we learnt.

[00:24:57]

Sounds very interesting. So at some point you start to think about going back to academia? Is it correct?

Yeah. You see, I had been for several years basically a security consultant who would go round the world to fix people's problems, and so I ended up in places where there was trouble, right? I was in Hong Kong before and after the Tiananmen massacre, and I was in South Africa when Nelson Mandela made his walk to freedom. And I even once got a lift out of Alex into Soweto in a South African Communist Party staff car. You know, while the South African Defence Forces are standing there in their armoured personnel carriers giving us the evil eye. So this is all very interesting and exciting when you're in your thirties, but when we got to 1991 there was a recession worldwide because the banks had lent too much money on property in London and, you know, to countries like Argentina and so on, so they were not spending on IT any more. And I was also suffering from imposter syndrome because there I was, I'd spent several years advising several banks and utility companies on how to do cryptography, and I'd never actually done a proper university course in it, not that there were many in those days. I'd never been to a crypto or Eurocrypt conference and I thought that some day I should do a PhD in this subject so that I actually know what I'm talking about. And so one day I'm sitting in my office and playing computer games and, you know, I'm paying the rent and I'm paying the secretary and I just said to myself, well, you always said you'd do a PhD one day, looks like today's the day. And so I contacted the various universities, including my alma mater, Cambridge, where Roger Needham said, sure, come round and have a chat. And I went round and had a chat with him and David Wheeler, they were the two full professors doing security and cryptography at the time. David Wheeler had been the first of Maurice Wilkes's research students and he's the guy who actually wrote the world's first computer program, because he crafted the initial orders for the EDSAC, a copy of which we duly gave to Bill Gates when Bill Gates bought us a new building in 2000 or thereabouts. And Roger Needham had also been a student of either Maurice Wilkes or David Wheeler, and he had invented some of the world's first cryptographic protocols when he was working in industry for Xerox at Xerox PARC where, you know, Chuck Thacker invented the modern workstation and Butler Lampson wrote the software for it, there you had workstations on a local area network, ethernet had just been invented and they were using that, and they needed some way of authenticating workstations to resources. So Roger had invented the Needham-Schroeder Protocol, and then somebody had broken it, so they fixed it and

then he worked on something called the BAN logic, the Burrows-Abadi-Needham logic, with Martin Abadi, who was then at Digital Research and is now at Google Research, and Mike Burrows who was then Roger's research student, and ended up writing AltaVista and ended up being at Google. And so Roger was very proud of this and gave me a copy of his tech report on the BAN logic. And as it happened, I had been doing some design work for a company in Johannesburg called Net1 that was also trying to do the transition to black rule in a contract they had with the Permanent Building Society. And they were trying to provide a portable cash card so that if you had people living in remote villages that didn't have any phone service, you would be able to have a bank card that you could use to pay for stuff, even when the whole village was offline. And so the idea was that you had a merchant card and you had a customer card, and the customer card produced a kind of electronic cheque, with two signatures on it, one of which could be verified by a merchant card and the other of which could be verified by a central server. And we were trying to design this system so that if anybody managed to break the tamper resistance on the cards it wouldn't be disastrous. And this particular design ended up being the template for something that was adopted by Visa, because years later it became eventually the GeldKarte in Germany and Proton in the Netherlands, and it went into the patent pool which gave us EMV today. But anyway, the NetCard protocol at the time was something that I'd been helping to work on, and so I went away with the BAN logic and I figured out how to use the BAN logic to verify it, and wrote that up in a paper which duly impressed Roger and David, and I got a research place at Cambridge.

[00:30:12]

So, can you describe your relationship with Roger Needham and David Wheeler?

Well, I was supposed to be a PhD student of David Wheeler's, but I just missed that because he was going to retire within three years of my starting, so I had to become Roger's research student, because Cambridge had a rule that you couldn't take on a PhD student unless you had at least three years to go before the retirement date. And that's a salient thing now, because we now have a big fight against the university to abolish the mandatory retirement date, but that's a separate story that we can come to later. So I ended up being Roger's research student and I was still trying to do stuff

with David Wheeler on cryptography, and I was just digging out all the old papers recently, I had to move office because I was forced to retire and move into a shared office, and I realised that I spent several months of the first year of research basically trying to design better identity-based signature systems. And we started off from one that a couple of guys in the Netherlands had published and David and I worked on various iterations. And I came up with a protocol that seemed to work and I sent it off to Eurocrypt, and it came back with the damning referee's report saying, sorry, this has been already invented, you know, Fiat and Shamir four years ago. And because I hadn't been going to the conference and hadn't read all the conference proceedings I wasn't aware that I was rediscovering this. So I was a bit downhearted and I said to Roger, well, this is harder than it looks, doing public key cryptography. And he said, look, there's a lot of bones in that mountain, you know, ever since Ron Rivest and pals opened it up in the seventies there have been a thousand mathematicians, you know, charging around in that field trying to pick all the low-hanging fruit, and if you find yourself in the end, you know, down on your hands and knees, you know, picking up the crumbs with tweezers that have been left behind by a thousand mathematicians, you're in the wrong place, because good research, he said, is done with a shovel, not tweezers. What you've got to do is to go and find some new problem and tackle it. So, my break came when 2,000 people sued thirteen banks for £2 million that had been stolen from them by means of phantom withdrawals from cash machines. And the lawyers who were running this hired me as an expert witness. Now, I knew a little bit from my work many years previously in banking how cash machines worked, and I was the only person around who wasn't currently on the bank's payroll and thus, you know, conflicted, and so I ended up doing this work and we collected a huge amount of information from various sources about how cash machine frauds were done, and there were quite a lot at the time because cash machines used very simple protocols and magnetic strip cards which were easy to forge. The banks had defeated that class action quite wrongly, they basically went to the High Court and they said, look, you know, here we are, thirteen banks, thirteen [sound drops out 00:33:33-00:33:39] turns to speak. It's going to take years and years and years to get through the preamble, so what has to be done is you must, you know, no doubt there are some people who are genuine fraud victims and no doubt there's other people who are just chancing their arm, so this is all down to individual cases, so what you must do is break this up into 2,000 individual cases in the Small Claims

Court. And the judge unfortunately agreed with them, you know, despite being a Trinity man, he should have known better. And so that class action basically failed and three years later I found myself in Southwark Crown Court being an expert in another trial where the guy who had done most of these frauds was sent down for six and a half years. He had basically developed various tricks for cloning mag strip cards, including parking a furniture van opposite an ATM and having a camera which would watch people enter their PIN, and he would then go to the rubbish basket where the ATM tickets were discarded, and in those days the ATM tickets had the full sixteen-digit account number on them, and there was no card verification value that anybody would check, so if you had an ATM ticket which said account number so-and-so and transaction time is such-and-such, and you have got your furniture van video says the PIN entered at the time such-and-such was 1232, then you're in business.

[00:35:05]

So that particular villain ended up having to serve his whole sentence because the banks lobbied against him getting out in half time on the grounds that he was dangerous, and afterwards he went to live in Thailand and there was a whole bunch of frauds which involved stuff in Thailand, which he may or may not have been involved in, or maybe he just told somebody how to do it, or whatever. Nobody knows. Anyway, so, with that I wrote my first big paper, *Why Cryptosystems Fail*, which looked at all the ways in which real cryptosystems fail, even though the cryptography was okay, the procedures around it, the way in which keys were loaded into hardware security modules then into ATMs, the way key material and cards and PINs were managed in bank branches, the operational security around, you know, dumb things like writing the full sixteen-digits of the account number on the ticket rather than just the last four digits. In other words, the cryptographers at the time were a bunch of idealistic mathematicians with no real world experience. There were some exceptions. Roger organised every year a protocols workshop at Cambridge, which is still kind of going on, and a frequent guest at the protocols workshop was Robert Morris Senior, who was then the Chief Scientist at the National Security Agency. And so he would come in with various gnomic utterances and talk to us about the necessity of getting the OPSEC right and the implementation right. And he was very

keen on the kind of work that I was doing and encouraged it. So, I ended up doing a bunch of stuff during my PhD around cryptography and crypto-protocols, and afterwards I fell in with Eli Biham, who was the inventor of differential cryptoanalysis. And one of the things that we did together, in fact I met him while I was still a PhD student, because one of the things that I noticed was that the theoreticians had taken over the conferences, but if you wanted to get a paper into a crypto conference or a Eurocrypt conference, they were more interested in a paper that provided a security proof of some non-real world protocol than of something that described a real cipher that you could use or a real protocol that you could use it in. And so people who were trying to do classical cryptography, that is breaking existing ciphers and propose new ones had nowhere to send their stuff. So Eli was one of the guys with whom I started a series of workshops on fast software encryption in 1993 so that we could have a place to put this kind of work. And in addition to Eli Biham there was the late Jim Massey of ETH, there was Lars Knudsen, there was various people from KU Leuven, initially Bart Preneel and then Joan Daemen, and Vincent Rijmen who eventually produced the Advanced Encryption Standard Competition. And for a period of time during the nineties there was a number of us working on designing better block ciphers and breaking existing block ciphers, and this is oil out of which the AES competition sprouted, because NIST decided, I think about 1997, that they needed a replacement for DES, because the DES key length was too short. And so this was something that ran through, or parallel with, lots of FSE workshops, and the block cipher that Eli Biham and I came up with, Serpent, you know, was first shown at an FSE workshop, as were the ciphers that became Jim Massey's entrance and the ciphers out of which the Rijndael algorithm that eventually became AES grew. So there was a whole community of us, you know, perhaps a hundred people who were working on block ciphers and stream ciphers. And that came to an end when the AES competition finished. Five of us got through to the second round and then the voting at the final AES conference put Rijndael first and and Serpent was second, and in due course the US government announced Rijndael the winner. And basically we screwed up because we took the brief too literally and the brief said that we want an algorithm that's faster than DES but more secure than Triple DES, and so we went for security, whereas the Rendahl algorithm had fewer rounds and it went full speed. And we reckoned at the time that there would eventually be a certificational break of it and eventually there was, but you know, with hindsight what

we should have done is cut Serpent from thirty-two rounds to sixteen so it ran twice as fast in the second phase of the AES competition and then we might have won it.

[00:40:19]

But anyway, after this competition, I took the view, well, what am I going to do now? I'm not going to spend the rest of my life being the guy who got silver in the encryption Olympics, so what new worlds are there to conquer? Now we had already started working on two other topics, we started working on copyright marking and information hiding. One of my first PhD students. Fabien Petitcolas, was fascinated with this, and so we ended up trying to break all the various schemes that various people had come up with in other information, and we ended up producing some software called StirMark which would do its best to remove copyright marks. So a series of rounds of attack and defence, and again, we set up a series of workshops, the Information Hiding Workshops, which eventually split into the ACM Information Hiding and Multimedia Security Workshop, which is where the police go, the people who do the video forensics. And PATS, the Privacy Enhancing Technology Symposium, which is where all the NGOs and privacy enthusiasts go, the guys who [sound drops out 00:41:40] forensics on your stuff. So that separated out later. So we had these two other lines of work in, basically the signal processing aspect, also the hardware tamper resistance side, because I recruited as a PhD student Markus Kuhn, who had developed some interesting hacks against smartcards whilst still an undergraduate, because you see, *Star Trek* started being encrypted in Germany, so if you were a Trekkie fan, the only way that you could watch your favourite programme was by breaking Rupert Murdoch's cipher. And this motivated a lot of bright young kids to try really, really hard to reverse engineer smartcards. So that became a line of work for us and eventually we had Sergei Skorobogatov, who for a number of years, about twenty years, ran our tamper lab. He's now independent, running his own company. But during the period we invented semi-invasive attacks on smartcards, the idea was that we started off by saying, you know, is it possible to circumvent the tamper protection bit on our microcontroller by just flashing light at it to ionise it. So we went and bought a camera flashgun from Campkins Cameras in King's Parade for twenty quid, and we mounted it on the top of a microscope and we found that yes, we could indeed unlock a microcontroller. So Sergei then went and bought a laser and

mounted that on the microscope and discovered that if you brought along a chip with a laser beam, then the photocurrent that you would generate enabled you to read out the state of the chip, because if a gate was switched off there would be photocurrent where there had been no current before, but if a transistor was switched on, then as it's already conducting, there's no difference. So you end up being able to get a false colour photograph of which flip-flops are at one and which flip-flops are at zero, you know, by scanning a chip with a laser. And this is something that Sergei made his own, semi-invasive attacks, as he called it. Over a period of time, twenty years, got dozens of papers and lots of awards and basically created the whole field of semi-invasive, semi-conductor failure analysis. And as feature sizes shrank, then of course you had to go through infrared and you had to start using electron microscopes and so on and so forth, but that became, you know, a separate line of business at the lab and one that I couldn't personally follow because I, you know, didn't understand, you know, enough of the physics and engineering, but Sergei and Markus made a good go of that.

[00:44:31]

So where were we? Yes, 2001, there was a couple of other things started in 2001. One was what we call API attacks, because in hardware [sound drops out 00:44:45] to generate and manage PINs, Personal Identification Numbers, and over the years these have become really, really complicated because as the banks started networking together, not just local banks, but big networks of banks internationally, and Visa and Mastercard got involved, you went from security modules with a dozen or two dozen transactions to security modules with hundreds. And complexity's the enemy of security, so I got a bright research student, Mike Bond, and I gave him the manual for a hardware security module and I said, 'Mike, nothing this complicated can be secure. Find the bug'. So he sat down and read it for two weeks and he said, 'I've found it!', and that was a false alarm, so he went away with his tail between his legs, but after another week he came back, 'Found it!', and this time it was a vulnerability. And thereafter we found one after another, after another. We found all sorts of ways in which if you submit transaction 164 to a security module and then 493, and then 615, then you know, out pops the master keys. And there's so much complexity that it's a really, really hard job to guard against these feature interaction attacks. And so that

gave us the pleasure of breaking the IBM 4758, which was the only device in the world that was certified to FIPS 140 level four, that is unbreakable by the US Government. So when you've broken a certified unbreakable device, that's a feather in your cap. So we sent the paper off to IBM and we said this is appearing in Oakland IEEE Security and Privacy in ten months' time, so you've got time to figure out how you're going to fix it and ship the software update. And about two weeks before our paper was supposed to appear, I was in another conference in Stuttgart and I sat down at lunch next to the head of IBM's banking services for Europe, Middle East and Africa, and I just said to him casually, over the main course, 'So how are you getting on with the fix for the 4758?' And he said, 'What fix?' And it turns out that the whole disclosure period, the whole ten months had been wasted by IBM because their hardware security people in Raleigh, North Carolina, were arguing with their software security people in Watson labs in New York over whose fault it was. So at the open conference we disclosed a live vulnerability that could be used to exploit the hardware security modules on which thousands of banks depended and there was a torrent of downloads of the paper from our website coming from IBM.com, as you would imagine. And this was another experience of, you know, how the internal political economy of the corporate world gets in the way of security.

[00:47:45]

And the other big thing that we started in 2001 was the economics of information security, because in May of that year I was at the Oakland conference and I met up with Hal Varian who at that time was the professor of economics who was in charge of the Information Management School at Berkeley. Shortly afterwards he joined Google, then a start-up, and became the chief economist and became quite prosperous and famous, but at the time he was just an economics professor. And he'd been consulting for some anti-virus company and he couldn't figure out why fewer people were buying anti-virus software than you might rationally expect, and so I was also curious about why it was that UK banks spent more money on security than American banks, despite the fact that UK banks are very good at blaming their customers for fraud, right? The consumer protection is much stronger in America than it is in Britain, as we had found when we did the test cases over cash machine disputes. And he said, well, you know, that's obviously got economic roots there, it's something to

do with the different incentives that people have, the banks in the UK probably have to be able to pretend that they're doing everything they can to keep the system secure, whereas American banks can take a more risk-based approach. And we also brainstormed a bit about what could be the case with anti-virus software, and with hindsight I think that at the time people didn't take viruses that seriously, because if your PC was infected with a virus, it would typically just go and do a denial-of-service attack on Yahoo or something like that, so it was no real skin off your nose. You know, ransomware hadn't been invented then. So we discussed all this stuff and he gave me a copy of his book, *Information Rules*, which the following year became the best selling business book in Silicon Valley, and this explained things like network effects, how in the IT goods and services markets, the fact that you've got network effects and also that you've got high fixed costs and low marginal costs and also that you've got technical lock-in, means that you tend to have dominant firm markets with one ruling monopolist. And so the best business strategy for a tech start-up is to race into the market and try and get there first and be the monopolist, and then just lock stuff down afterwards. And while you're racing for dominance, I figured out, what you need to do is to appeal to complementors. Now, so when Microsoft is fighting Apple or when [incomp 00:50:43]'s fighting IBM over [incomp 00:50:45] or whatever, you've really got to appeal to the developers, and if you have designed an operating system with the kind of access controls that you have nowadays, you know, in FreeBSD with CHERI, for example, it would be just too complicated to write programs, so somebody else would have won that particular market race. And looking around we realised that this pattern of bargains followed by rip-offs in information goods and services markets is mirrored again and again and again by people building insecure systems which they then lock down later, but not always in ways that benefit the user, typically in ways that benefit the company in terms of increasing its lock-in. So that then started the whole thread of doing research on security economics and Hal and I organised the first workshop on the Economics of Information Security in Berkeley in 2002. And getting that going was hard. I wrote up a paper on it. I wrote my first copy of my *Security Engineering* book and I found that I was using economic discourse to link all the [incomp 00:51:57] stories together and so I pulled out all these pieces of economic argument and put them in a paper, called *Why Information Security is Hard: An Economic Perspective*, and I sent it off to one of the conferences, maybe it was Oakland or the CCS, I can't remember, and a

very snide referee's report came back saying, 'This paper contains no mathematics, send it to a management conference'. But luckily, I had been invited to give a keynote talk at SOSP, the Symposium and Operating System Principles at Banff in Canada, so I went and gave the talk there and it really took off among the assembled operating system security crowd, many of whom had been involved in government work and had been aware that they had failed for, gosh, twenty, thirty years to persuade any of the big tech firms to produce any operating systems that would pass Orange Book evaluation. And thereafter we had the WEIS workshop and we got together, gosh, forty or fifty people at the first WEIS who'd been thinking about similar stuff. You know, Jean Camp had been talking about the need for vulnerability markets, which just emerged round about that time. Bruce Schneier had written about it, once or twice about the role of incentives. And there was a couple of guys from University of Maryland who'd been financial economists, Marty and Larry, had written a paper on the incentives to invest in information security and what's your optimal investment. And from these seeds grew the modern discipline of security economics. So that's been a wild ride. I think it's contributed quite a lot to our understanding of how things break in real life.

[00:53:52]

Yes. So you mentioned your book. So in 2001 you publish your book, Security Engineering: A Guide to Building Dependable Distributed Systems*, and it's now in the third edition, right? Was published… and also translated abroad. So, how did you come up with the idea of writing this textbook? Was it when you were going to become professor in Cambridge? So, what was the…*

I was inspired by Bruce Schneier, whose book, *Applied Cryptography*, had become a runaway bestseller, and I thought there needs to be something similar that looks at security in the round, and I had also been teaching a series of courses at Cambridge, a first-year course in software engineering and I was doing some second-year stuff [incomp 00:54:54] and group projects on e-commerce and whatever, and a third-year course on cryptography and security, so I had a lot of the material and the students wanted write-ups, so I'd written what were the core of, you know, more than a dozen of the chapters already, and I added extra chapters as being case studies of what was

wrong with cash machines, or what was wrong with pre-payment meters. I then wrote fresh chapters on some other things like security printing and seals and so on and shipped the book and it took off and we never looked back.

So a lot about, like in your job in some sense, it's a lot about learning from failures, right?

Absolutely. If you're doing real engineering you have to look at real systems in the real world, you can't do that by just standing at a blackboard and thinking about mathematics.

That's why also in some sense it's important that someone who is developing things in this field doesn't just work in academia but also has experience perhaps in industry. So what's…

Yeah, you see I was aiming the book not just at a PhD student who needs to bring himself up to speed with what's going on the field and, you know, learn the basics across a range of different self-disciplines. I was also aiming it at Dilbert, you know, the random guy sitting in his cubicle somewhere in America or elsewhere who's trying to build systems and suddenly he needs to know a bit about security.

Yeah. So this makes me think a little bit also about how much we rely, you know, on the systems also for health, and I'm talking about not just hospitals, but I'm also thinking about, you know, pacemaker, for instance. So have you had experience in this field of health?

Yes, I did some work for the British Medical Association in 20… sorry, 1995/96, because the government wanted to centralise all Britain's medical records into a big database system, because they wanted to be able to manage the health service a lot more closely and the doctors didn't want that. And so the grounding on which they chose to fight was patient privacy, and whether the network should be encrypted and what the access control rules for the system should be, and so on and so forth. And so I advised them on that for a couple of years and I got to see how governments behave

and misbehave when they're trying to get their way and how they fail to build systems that are actually any use or they're any good.

So, I found in another interview you gave that, you know, at some point you received some piece of advice from Roger Needham in consultancy in, you know, in the defence sector. So I think that was quite interesting, because what he basically told you was about being careful if you wanted to have an academic career, about non-disclosure agreements and that could affect your career, every paper you would publish in the future.

Yeah. Because you see the GCHQ, our signals intelligence service, has got a standard playbook with academics to do relevant research. They invite you to get a security clearance and sign the Official Secrets Act, and they'll then give you some entirely trivial consulting where they tell you some totally unimportant top-secret fact, and they then use this as a means to demand the right of prior review over all your papers forever. Now, Roger Needham himself was warned of this as a young man by the late Donald Davies, who was at the National Physical Laboratory, which also did a lot of the early work on cryptography, and he resisted attempts by GCHQ to bring him within their net and he gave Roger this advice, and Roger passed it on to me. Now Roger did in the end get a security clearance, he took early retirement at, I think, sixty-two, in order to become the boss of Microsoft Research in Europe. And then he got a security clearance and he got a big green safe in his office and he sat on the Defence Science Advisory Board. But while he was a working academic he wouldn't touch them.

[00:59:30]

So talking about your collaborations with industry, in 2011 you were visiting scientist at Google, so can you tell us about your experience there?

Yes, so I spent three very enjoyable months at Google, where my main contribution was being part of the team that designed Android Pay, you know, the mechanism that you use to tap and pay with your mobile phones. The ambition was to get it running in time for the London Olympics in 2012, but although we had the system itself

working in late 2011, moving the banking system was just too hard. There's an awful lot of inertia in payment systems as there are in other technological systems that have network effects. And the big showstopper is how you persuade large numbers of stores to spend hundreds of millions of pounds replacing all the chip and pin terminals. Because, you know, going to phone-based payments means basically going to tap and pay, and this was being done at the same time as a move for ordinary bank cards to go from chip and pin to tap and pay. And so that involved a lot of equipment replacement. And the big store chains only really bought into that once Apple also brought out its own Apple Pay, and it was then clear that this was going to be a future direction. So, you know, the CFOs of the big store chains said fine, you know, let's replace the terminals.

You also had, in 2011 you also had your collaboration with Carnegie Mellon University on cybercrime.

Yes. My sabbatical in 2010, 2011, as well as spending three months at Google, I spent three months at Carnegie Mellon. And at CMU I was working with Alessandro Acquisti and George Loewenstein and Nicolas Christin, and we applied for and got a large Department of Homeland security grant, of which Carnegie Mellon was the lead grant holder. And this was on the behavioural economics of cybercrime. In other words, once you start using economic and behavioural economic ideas, can you understand a bit more about the kind of people who do cybercrime and perhaps how they can be deterred. So this led, again, to work on deterrence of deception which Google helped to fund, and so we started, also hiring a cache of people whose degrees were in psychology rather than just economics or computer science. And in addition to that, the CMU programme also brought in Tyler Moore, one of my former PhD students who'd become a professor at the University of Tulsa. And then in 2015 we got a big grant from the UK Engineering and Physical Sciences Research Council to consolidate this as the Cambridge Cybercrime Centre. Now, this is actually one continuous development, you know, over a period of more than ten years now. And the basic idea is this, that a dozen years ago there was no such thing as scientific research in cybercrime, because somebody would go and get some data, typically by drinking beer with somebody from an antivirus company, and once they'd drunk enough beer they'd be given some data, and their NDA, they'd analyse it, they'd write

their thesis, they'd publish a paper and they'd go and work at Facebook. But anybody who wanted to challenge or to build on that work couldn't do it because the data weren't available and weren't maintained or curated. So we thought, well, we collect a lot of data anyway, and we can also get in data from industrial partners, you know, from firms that run spam filtering services or threat feeds or whatever, who are prepared to let their data go to academia, just provided we don't supply it to anybody who might actually pay them money for it, okay? So we set up inbound and outbound licence… [sound drops out 01:04:10] cybercrime, you can give it to us and we can give it to 350-odd researchers to play with, under an appropriate NDA. And if you're a researcher who wants to do a PhD on cybercrime, well, we've got the data. Or if you want to try and train one of these new machine learning classifiers to spot, you know, bitcoin scams or even to spot hate speech, we've got the data. We started collecting hate speech as well three or four years ago, and so we've got one database, CrimeBB, which is a scrape of over a hundred million messages sent to underground acquisitive cybercrime forums, places like hack forums, where people buy and sell malware and try and recruit kids into their crime gangs and so on and so forth. And so this could be used by social scientists, criminologists, psychologists and so on, to track the evolution of particular crime types, and even to discover new crime types that we didn't know about before. And because we've got this data and because we maintain it as a resource for researchers worldwide, you know, we're a bit like a particle accelerator or a space telescope, you know, this is a shared resource for all researchers.

[01:05:31]

And do you also provide… do you provide analysis and also do you provide solutions for these crimes?

I don't do solutions. Solution is a marketing world and if somebody tries to sell me a solution, I know he's a…

Advice or reflections, what…

No. We write…

… that could be, you know, used in… Let me rephrase. So, if you…

We may sometimes provide tools. Solution is a marketing world and I'd avoid it like the plague.

Yeah. So you offer tools. So what kind of tools, for instance, just to understand…

Well, we are principally providing the data, but we're currently working on a search engine, which we've stood up using Elasticsearch, which enables people to go through our collections of data and look for messages of a particular type. So if you want to look for everything in hack forums in the last twelve years that mention sim swapping, for example, that just becomes a search on our search engine like a Google search. And the reason that we built this is that the majority of our licensees are not actually computer scientists, they are basically humanities and social sciences people. They're lawyers, psychologists, political scientists, criminologists and so on, and if we can provide them with better tools, you know, we get more uptake and we get more use of our data.

So, 2015 was also the year you got the Lovelace Medal from the British Computer Society. In 2009 you became a member of the Royal Society and the Royal Academy of Engineering. So, did you expect all this recognition? Was this coming as a surprise?

Well, to some extent, because when I'd been a research student, Roger Needham had warned me against putting FRS on my list of career ambitions, because it was in those days badly biased against engineers. He said if you want to be an FRS you're better off being a theoretical chemist, because then you've got a chance of getting in as a mathematician and a chance of getting in as a chemist. But if you try and get in as a computer scientist as an engineer, you're competing against all the electrical engineers, chemical engineers, civil engineers, etc. And so I didn't really put it on my list of ambitions, and then, I think 2007, something like that, somebody at Cambridge nominated me and warned me that, look, you know, don't have any high expectations, you might get lucky, but you quite possibly won't. So I just then forgot about it, so it

was a great surprise when it came through. And the Lovelace Medal was kind of similar. I mean that's the top computing award in the UK. I think it's extremely unlikely that I'll get the Turing Prize, which is the top US prize, because that tends to go to people who are Americans, people who are theoreticians, and also to people who've done one really enormous thing, rather than, you know, half a dozen fairly big things, as I've done in my career.

[01:09:04]

Is there parts of your research we haven't covered that you would like to talk about?

Well, over the past four years or so, we've been [sound drops out 01:09:20] on adversarial machine learning. I had one particularly able PhD student, Ilia Shumailov, who has got a [sound drops out] ways of [sound drops out] robots. And he was funded by Bosch who wanted to [sound drops out] the vision systems used in automatic emergency braking and automatic linking, were sufficiently dependable, you know, to be used in a safety environment. Between Ilia and I, we basically looked at the whole supply chain of AI systems, the way the data are collected, the way they're batched, the random number generators that are used, the methods of gradient descent, the architecture. And we looked at all those various ways in which these could be attacked and we came up with a number of attacks that are widely spoken of, and in fact, if you looked at the missed taxonomy of attacks on AI systems there's about half a dozen of ours in there. The two best known are the sponge attack where you craft inputs that will cause the machine learning [sound drops out] or waste energy, and the coding-based attacks where you use, you know, things like the control characters that change the direction of rendering in order to mess around with natural language processing systems. We also discovered what we call model collapse, and the question there is what happens if you've got a machine learning model that trains on its own output generation after generation. So if the internet, the GPT-3 raids the internet and that's used to train GPT-4, and then that raids the internet and that's used to train GPT-5, you know, by the time you get to GPT-7 or GPT-8, what's happening, and the answer is, you've got gibberish. Right? Because the [01:11:15 sound drops out] are truncated more and more until you end up with just getting [incomp 01:11:19] functions, and then the whole output is nonsense. And you could think of it in a

musical context, that if you design a system which will compose music and you train it on Mozart, you'll get something that produces a lot of stuff that sounds a bit like Mozart but doesn't have the sparkle, okay? So you've got a robot that writes Salieri, and now you fill the whole internet up with Salieri and you train another robot to try and imitate Salieri, and by the time you get to the third or fourth generation, it's producing either vapid elevator music or horrible beeps and squawks. And that really matters, because if you think you can keep on scaling up large language models, then there's a certain limit beyond which you can't do it, because, you know, you've already raided the whole internet and what else is…

But if you look at your career, is there anything you would do differently, and why?

Well, with the benefit of hindsight of course we could have got to where we are a lot faster, and there's been a whole series of times in my career where somebody comes out with a great idea in cryptography and you say, oh shit, I wish I'd thought of that. But, you know, apart from that, I think I've had a fairly good run. I've been lucky to be around at the right time, in that when I started publishing, the field was so small. My first serious paper, *Why Crypto Systems Fail*, appeared at the first CCS conference and there were about eighty people there. So in the space of three days I could get to know them all, including, you know, many of the greats in the field like Whit Diffie and Dorothy Denning and Matt Blaze and Carl Landwehr and so on, they were all there. And Roger was there also to introduce me to them. And for someone coming into the field now it's harder work, because if you go to a CCS conference nowadays, there's going to be 1200 people and there's going to be a hundred papers in about six different tracks. And so much of the low-hanging fruit has already been picked.

What is the proudest achievement of your career?

I don't know. I suppose it's for the security economics work that I'm most known, but you know, where I get my kicks is from working on novel problems with bright young people and pushing forward the boundary. I'm not the sort of person who's just going to sit there and look at my old best paper awards from twenty years ago, you know, I'm not that old and decrepit yet.

[01:14:18]

So what advice would you give to one of your students, or what kind of advice you give to your students?

Well, the students basically have to follow their dream, and if they can't think of anything better to do, then of course they must go and work in industry and get some real world experience. But if you've got somebody who goes and works in Google or Facebook or whatever for a year or so, decides they don't like it and goes to academia instead, that's perfectly fine. You know, you don't need a big house and you don't need to fly business class on holidays, what you need more than anything else is to be happy. And on the other hand, there are people who've been in academia, I know a number of people in academia who've left to go and work in industry, because they eventually get bored with academia and they want to build stuff that people will actually use. And, you know, both types of career path are just as valid, it depends on your circumstances and your personality and other factors at the time.

If you think about the future, what do you think are the biggest challenges and opportunities, you know, related to cybersecurity and democracy perhaps, you know?

Well, over the next five years, I would reckon, the thing to be looking at is how artificial intelligence and machine learning will change the cybersecurity landscape. Lots of people are hoping that large language models will make it easier either to do attack or to do defence, and so far that doesn't seem to be happening. We're probably going to see for the most part just incremental changes as AI tools are used as personal productivity assistants by people doing either attack work or defence work. But it's quite possible that you'll see something changing radically. Now, the best analogy that I can give is ransomware. Now, we did a couple of surveys of the cost of cybercrime, one in 2010 and one in 2017, and the amazing thing was, that the pattern of crime hadn't really changed, despite the fact that people had moved from laptops to phones and that they'd moved from on premises service to the cloud, or that they'd moved to social with everything. And so that's showing you that the patterns of cybercrime are not fundamentally technological, they're fundamentally to do with the constraints in the surrounding, you know, legal and economic system. But the one

thing that was changing by 2017 has really changed since then is ransomware, because the invention of bitcoin meant it was possible to collect ransoms, which hadn't been possible with any skill beforehand. And so I think that an awful lot of the growth in the cybersecurity industry over the past few years has been down to the fact that medium-sized firms are now at risk of, you know, being hit by serious ransomware attacks and having to, you know, shell out tens of millions of dollars, or having the embarrassment of seeing their customer data or their internal emails posted on the internet. And so that's, you know, ransomware is bringing winners in the form of the cybersecurity companies. It is quite possible that some application of machine learning will similarly trigger some radical change in the environment sometime in the next few years. And so watching out for that, figuring out what it is and what to do about it, will be one of the big research problems in the years immediately ahead.

If you had the political power of changing something, you know, based on your experience, on your work experience, what would be the first important major change you would enact?

I suppose my focus would be on the standard and policy questions like consumer protection, like competition, and like privacy. These are things that come up again and again and again in the tussle between big firms and small firms, and the tussles between exploitative monopoly firms and defenceless consumers, in the tussle between police and intelligence agencies and tech. But given the nature of things, it's unlikely that any new dispensation would stay uncontested for long. There'll always be somebody coming along trying to lobby for a little bit more of the pie.

Is there anything else you would like to discuss that we haven't covered?

No, I think we've just about covered everything.

Okay. Thank you very much then. That's been lovely talking to you today.

[end of recording]