An Interview with

ROSS ANDERSON

OH 461

Conducted by Jeffrey R. Yost

on

21 May 2015

Computer Security History Project

Cambridge, United Kingdom

Charles Babbage Institute
Center for the History of Information Technology
University of Minnesota, Minneapolis

Ross Anderson Interview

21 May 2015

Oral History 461

Abstract

Computer security pioneer Ross Anderson discusses his education and early career as a computer security consultant (serving banks and other companies) before returning to school to complete a Ph.D. working under Roger Needham at the University of Cambridge. The bulk of the interview focuses on his academic career in the computer security field at the Computer Laboratory, University of Cambridge. Among the topics discussed are cryptography, computer security education, and Anderson's leadership role in launching and providing a substantial infrastructure for the development of the field of computer security economics—including the annual Workshop on the Economics of Information Security (WEIS), which Anderson co-founded in 2002.

This material is based upon work supported by the National Science Foundation under Grant No. 1116862, "Building an Infrastructure for Computer Security History."

Yost: My name is Jeffrey Yost from the Charles Babbage Institute and I'm here at the University of Cambridge with Professor Ross Anderson. This oral history is being funded by the National Science Foundation as part of a CBI project, "Building an Infrastructure for Computer Security History." Ross, can you begin by just giving me some basic biographical information, when and where you were born?

Anderson: I was born on the 15th of September, 1956. I was born in Wallasey in Cheshire and my dad was an academic, who was at the time doing some work for a pharmaceutical lab in the neighborhood. I went to school for the first term or two in Wallasey but then we moved up to Scotland, where I remained up until at the age of 18 I came down to Cambridge. We lived first of all in North Lanarkshire, near a village called Annathill, which was a mining village, and has since been knocked down after the pit became exhausted. Then when I was 11, we moved to the west coast at Gourock, where my dad still lives. Dad was a professor at the University of Strathclyde, and Gourock was as far as you could reasonably commute. It was 40-50 minutes, 33 miles west of Glasgow and also on the seaside. It's where you get off the train and on to the boat.

Yost: What was your father's academic field?

Anderson: Dad was a professor of pharmaceutical technology and he did all sorts of things from isolating and developing new drugs, to developing better machinery for tableting and working on adhesives for stoma bags. My mum had a stoma from a young age and this was a new procedure at the time. Dad spent some time trying to figure out

how to get the gums right. But dad's main achievement was working with oligosaccharides and one big thing was investigating carrageenans for use as an ulcer drug. People in the west coast of Scotland had long used carrageen pudding to settle an upset stomach, and so dad isolated the active ingredient of this and it was used for a few years as a drug in France and Germany. It was eventually overtaken by Tagamet because that could be patented.

Yost: Can you describe yourself as a student in your pre-college days, both in terms of interest and aptitude?

Anderson: I was always one of the cleverest kids in the class but because I didn't have binocular vision – I was born cross-eyed and had a sight correction operation when I was about three – I was completely useless with moving balls, and so I was hopeless at rugby or cricket. And this made my life pretty miserable, because at my school such sports were mandatory, and I only escaped the torture of it when I was about 15 and able to take up rowing instead. Until I was 16 I was being steered towards a career with law or medicine, with my folks preferring medicine and the school headmaster, who was a classicist, preferring law. But then I came across a book in the local library, which got me going; it was by Felix Klein. It was *Elementary Mathematics from an Advanced Standpoint* and Klein was one of the great mathematicians of about 100 years ago. This is a book of lectures that he got together for students who were about to be maths teachers explaining how they could use ideas from what was then research mathematics, to inspire

schoolkids. And with that I was just hooked, and I became determined to be a mathematician.

Yost: Can you tell me about your decision to attend Trinity College at the University of Cambridge?

Anderson: In Scotland it was normal for kids to leave school at 16 and go on a four-year university degree, rather than leaving at 18 and doing a three-year course, as is the norm in England. And so I went to Glasgow University to study mathematics. I was actually supposed to study medicine and I even got a place to study medicine, but I decided at the last minute to do maths instead, which the medics thought was completely inexplicable and they kept on sending me all the spam and paperwork for medical students until I was about halfway through my first year. But when I arrived at Glasgow, I fairly rapidly observed that all the professors had been lecturers at Cambridge, and all the lecturers had been researchers at Cambridge, and I realized that I wasn't at the mother ship. So within a few weeks of arriving at Glasgow I got the paperwork for Cambridge admissions, filled that out, and at the end of my first term at Glasgow I came down to Cambridge, did an admissions interview, and must have impressed the people I spoke to; Jeffrey Goldstone, if memory serves, because I got an offer of a place. And so aged 18, I arrived at Cambridge having already done a year at Glasgow.

Yost: And you had firmly decided at that point that you would do mathematics?

Anderson: Absolutely. This was my mission in life, to be a professor of pure mathematics.

Yost: How did you become interested in the history and philosophy of science?

Anderson: My undergraduate years were fairly turbulent. I started off doing the second year of Cambridge Tripos in my first year, which is something traditionally done by bright students and I thought that as I'd had a year of maths at Glasgow, I could have a go at that. I did, but it was very, very hard work. As a result of an end of term prank that got a bit out of hand, I was advised to spend a year away, at the end of my first year. We had a culture among undergraduate mathematicians whereby we would burglarize each other's rooms and play pranks, turning the bookcases upside down or whatever. One of my colleagues, who is now a professor of mathematics at Bath, had gone away for a weekend and left a note at his door as a challenge. And so I jolly well got into his room and then a thunder flash went off between the inner and outer doors. The bomb squad was eventually called and the whole thing got played up out of all proportion. The chap from special branch had come in and then went into the room and turned on the light switch, which caused a secondary explosion because this had also been wired up to another thunder flash. And with that, the whole machinery of state security ground into action.

Anyway, to cut a long story short, my friend and I were invited to spend a year off, and he spent it working for a software company while I spent it working for Ferranti in Edinburgh, where I was involved in designing inertial navigation equipment. My task

there was to take the inertial navigation set for the Tornado, which was about then to serve as Britain's front-line fighter bomber, and redesign it so that it could be used on submersibles. This meant taking a piece of kit with a top speed of 1500 miles an hour and a gyro drift of about one mile an hour, and repurposing it so it would work for a top speed of about 10 miles an hour and a drift of hopefully only a few yards per hour, because the mission was to take divers to the right wellhead in the North Sea. So I managed that within a year and we even took the thing out and tested it on a North Sea supply ship. That was a lot of fun and taught me an awful lot.

Yost: Was there any computing work involved with that?

Anderson: Absolutely. I built a computer, and in those days you didn't just get a microcontroller because those were just a glint in Gordon Moore's eye at the time. The computer that we used was made with 54 series logic soldered to a board. There are about 30 logic chips in a CPU, and about the same again in our support unit. I had quite a bit of difficulty getting a suitably sensitive analogue to digital converter to work. I had a colleague in this who wrote the software and yes, it was a nontrivial hardware project.

Yost: And was that your introduction to computing or had you used computers at Cambridge previously?

Anderson: Well my introduction to electronics came in the Boy Scouts when, at age 11, a local guy, Ian Simpson, volunteered to run an amateur radio club for Boy Scouts. He

was in our village, in Gourock, and he was a keen radio ham and he had a shack with all sorts of wonderful toys in it. We used to go there and solder things together. Now, when I was 15, the Glasgow school system opened up a computer center, which had a small IBM mainframe in it. So we kids would go and write FORTRAN on punch cards, and stand in line to feed our cards into the machine and get our output out from fan-fold printers. And that was the high experience for me. Once you have got the means of controlling the flow of instructions you could get it to do whatever you wanted. You could get it to print out a book of multiplication tables or a book of log tables, and these were the first things that I tried. And then you could get it to play computer games and that was rapidly what the other kids were interested in. So, yes, I was privileged to have an introduction to programming in the 1970s, the early 1970s, when most kids didn't have that access.

Yost: And did that play into your interest in mathematics at all?

Anderson: It existed alongside it because the guy who ran the school computing thing, Willie Wilson, was also the head maths teacher and was my maths teacher my last year in school. He gave us some formal lessons in FORTRAN. He even had one of the strips in his rolling blackboard replaced with a FORTRAN coding form so that he could write up programs in the class and explain them to us.

So anyway, I went back to Cambridge, age 20, to do my second year of college, somewhat more grown-up I think. I also, while I was at Ferranti, qualified as an electrical engineer. I did the Council of Engineering Institution's exam. And being good at maths, I found this fairly easy and, of course, I'd experienced working in the real world. What I

found in my second year at college was that pure maths didn't have the appeal that it used to and I was put off by a number of factors. Some of the other students were just way better at it than I was; you know, they would live and breathe group theory and algebraic number theory and I couldn't really see the point of these things at the time because cryptography, public key cryptography hadn't really come along yet. I had some interest in applied maths and analytic number theory but my interests in maths gradually waned and so I found myself with the need to do something in my third year of college, having already done the finals. The options for what you can do after a mathematical Tripos at Cambridge were a diploma in computer science, which I seriously considered but didn't find inspiring at the time; I could do what's now the master's in mathematics, that's the one-year post grad course in maths, which is absolutely brutally hard. It's a competitive exam for research places and I didn't face the prospect of that with any enthusiasm. The other possibility was doing history and philosophy of science, and I pumped for that just for want of something better to do and because one of my mates, Simon Schaffer (who is now the professor of history of science here) was doing it and found it really interesting. So I did that for a year but it didn't lead to postgraduate work or a research place because back in the late 1970s, Britain's economy was in a bad way and there was very very little research student funding. So I just wandered off into the world to make a living. And after a gap year when I traveled around Europe, Africa, and the Middle East, I did various odd jobs for a while but eventually fell into computing because it was the obvious way to earn a living.

Yost: When you were an undergraduate, did you have any courses with Roger Needham or did you encounter him?

Anderson: I didn't know Roger Needham or David Wheeler when I was an undergraduate. The only one of the computing pioneers that I got to know was J.C.P. Miller because I went to his graduate courses on complex polytopes. We did have some computing classes as maths undergraduates. We were taught to program in a language called FOCAL, which is a mixture of FORTRAN and ALGOL on PDP-8s, which were kept in the basement of applied maths. That focused on numerical analysis more than anything else, so at least the programming skills didn't get entirely rusty. Then, of course, in the early 1980s, when home computing burst on the scene, there was suddenly a huge frenzy of activity. I started writing software for home computers and doing things like that, which was great fun.

Yost: Were there other computing courses you took besides that one?

Anderson: No, on the computing side of things I was largely self-taught, so I came back to do a Ph.D. in my early thirties. How I got interested in cryptography; well, there was a friend of mine, Ross Baldwin, who was working as a programmer for an estate agency, if memory serves. And the estate agent had told him that they wanted encrypted e-mail for the partners so they could send secret notes to each other. Ross thought about this for about five minutes and came up with a scheme that many programmers invented independently, which is that he used the password to initialize the random number

generator, and then just called it and x-or'ed it with each byte of the file in succession in order to scramble it. And I took a look at this and I said "Now hang on a minute, surely this isn't right; surely I can figure out how to get back to the password!" And having figured that, I then started digging around, and I started looking in the research literature, and there was just at that time beginning to be a research literature in cryptography. The first crypto conference, I think, was in 1982. And by 1985, Henry Beker and Fred Piper's book on cipher systems came out.

It was in this window in here that I started to get interested in cryptography, and I dug around a lot in the relevant literature with a friend, Keith Lockstone, who was at that time working for Marconi, now British Aerospace. We put together an e-mail encryption program called Ciphernet that we sold to one or two people. That was basically an improved stream cipher. Now how that came about was that the center of research activity at that time was Royal Holloway and the cipher they were promoting was a multiplex shift register generator, which one of their research students, Sylvia Jennings, had come up with and had managed to prove some nice results about in her thesis. Now I took a look at this and managed to find a divide-and-conquer attack on it, which gave me the self confidence that if I can break a cipher that's reckoned to be academically leading then I must have some talent for this business.

So we put together an improved random number generator based on shift registers, multiplexers and other bits and pieces, which inherited the good properties of the multiplex generator but without its vulnerabilities [to] divide-and-conquer attack. And we sold it to a number of firms who were selling software to banks. This was at the same time as writing home computer software.

Yost: Can you tell me what time frame you were doing this?

Anderson: This was 1984-85. And then I got a job with a bank. Barclay's Bank was looking for someone to look after the security of cash machines, and funds transfer, points of sale, and so on; in other words, somebody who understood cryptography and could join their information systems team. So I worked there for three years and that gave me an insight into the corporate world and how one turns up for work every morning wearing a suit and all that ritual around that, which I must say I didn't really take to. The bank was almost a civil service bureaucracy with a dozen different layers of management between the people who did the work and the guy with a very large salary.
After I worked for Barclays I worked for Standard Chartered for a while in 1989; Standard Chartered was at the time refurbishing its branch network in Asia and wanted somebody to put together a security policy and strategy to protect all the new systems. Standard Chartered, like Barclay's, was involved in moving all its banking systems from non-IBM to IBM mainframes, plus a product called Hogan, which was initially run by a company in Texas called Hogan, but was then bought by IBM and became IBM's standard bank-in-a-box offering. So as I knew a bit about this by then, I spent some quite interesting times in Hong Kong putting together what became Standard Chartered's branch network system for use in 23 countries in Asia.

Yost: I assume these were larger group projects, if so, how many people worked on them?

Anderson: The Barclays project was several hundred people and the Hong Kong project was about 80 people. They were substantial projects costing hundreds of millions, and that also taught me there are good ways and bad ways to do big IT projects. Standard Chartered was notably better at running things. They had a can-do attitude to technology. This is something that has continued to interest me. Now that I'm an academic with an interest among other things of how governments use IT, it's notable that governments tend to be very bad at large IT projects. They often fail and this is something we even teach in a course to masters of public policy students. So I suppose I first got blooded in the realities of that back in the 1980s.

After I had left Standard Chartered, I started doing consulting work of various kinds. One of these projects was for ESCOM, the Electricity Supply Commission in South Africa, which with the change of regime there was interested in electrifying millions and millions of black African households. How do you do this when people don't even have addresses, let alone credit ratings? The answer is prepayment meters. And so there was a big project to do electricity prepayment meters. There were a number of flavors being tried out at the time. Some involved things like bus tickets that went into a magnetic strip reader in the meter. The technology that eventually won out used cryptography in that you got a 20-digit number which magically would top up your meter, by instructing your meter to credit so many kilowatt hours or so many rands worth of electricity.

These had a number of fairly interesting failure modes. The first one or two times you ran a design you never get it right. But that was a real success; it was the project that enabled

Nelson Mandela to deliver on his election promise to electrify two million homes. So I visited South Africa a number of times getting that together.

Around about 1991, I was beginning to get fed up with the life of being an IT security consultant and there were a number of reasons for this. The first was that you keep on doing the same thing over and over and over again. I was applying the same basic ideas — essentially ATM technology — to one application after another; point of sale devices, prepayment electricity meters, payment cards, you know when the first smart payment cards came along. The second thing was that I felt a bit of imposter syndrome. I'd never done a computer science degree and I had never done serious systems programming. I'd maintain programs even in assembler if I had to, but I was aware that there were gaps in my knowledge. Here I was representing myself as, you know, the great expert. I'd never even been to a crypto conference, so I had it in the back of my mind that it would be a good thing to go do a Ph.D., perhaps in this field. And then in 1990-91, business started to be real slow because there were stock market and property crashes in 1989, and this meant that in 1990 the banks stopped spending money. There wasn't any budget for new projects and so people who made a living as IT consultants were beginning to tighten our belts a bit. But by then I'd saved up enough money I could be independent for a few years and so I just decided that "I'd always said I'd do a Ph.D. one day, and it looks like today's the day." So I started calling and seeing what the options were.

Yost: How did you go about marketing yourself as a consultant and was it difficult to land opportunities?

Anderson: It was all word of mouth. It all ran from people that I'd got to know initially while working at Barclays and then at Standard Chartered. You see, the banking IT industry's a fairly small industry. First the banks have got bank security and standards committees; you go to meet your opposite numbers from all the other banks. Second, all the salesmen from the companies come in and try to sell you stuff, so you got to know the salesmen, and then the CEOs with all the supplier companies inviting you to rugby matches or the opera or whatever. That's just part of the marketing cycle so once you've worked in an area like that for a while, you get to know the relevant people. I suppose I must have impressed people by actually understanding the mathematics that underlay all that. At a typical bank standards committee meeting there might be 20 people at the table and about three of us knew what we were talking about. The rest of them would be, you know, managers who had joined in the branch network and just proceeded up the management track, rather than people who came from a computer science background.

Yost: . . . and had no idea of the mathematics at the time.

Anderson: So I suppose this enabled me to get the recognition as somebody who had some clue of what he was talking about.

Yost: When you decide you want to go back to school, did you consider a number of different schools or was it always Cambridge at the top of your list?

Anderson: Cambridge was always on the short list. I thought about perhaps going to the Weizmann but I was put off by the fact that I'd have the additional hassle of having to learn Hebrew. I did speak to Ron Rivest at MIT; I called him up. But it turned out I'd missed the deadline for MIT admissions that year by about two weeks. By contrast, when I called the computer lab at Cambridge, I got put through to Roger Needham, who was the head of department and also happened to be a security guy. He said fine, come around and talk to us. So I did; I went around and talked to him and David Wheeler, and we got on fine.

Roger had at that time just produced something called the BAN logic, the Burrows-Abadi-Needham logic. I think this interview must have been about January 1991, something like that. Anyway, the Burrows-Abadi-Needham logic enables you to verify an authentication protocol by tracking the beliefs of the parties. This is something that Roger and his research student, Mike Burrows, had done with Martín Abadi, who was a researcher at the DEC Systems Research Center, and went on to become one of the most highly cited papers in computer science during the mid-1990s. I was, at the time, doing some work for a company producing smart card payment system, NetCard, and this went on to become a VISA product; it was one of the precursors of EMV. So I took the BAN logic away and thought about it for a bit, and applied it to the protocols that NetCard was designing and found ways to improve the protocols. And [I] showed this to the company who used it to improve the product. I then produced notes on how this logic could be applied to a problem of real engineering interest in the payments industry, and I think that must have impressed Roger because I got an offer of a research place.

Yost: Did that project move forward?

Anderson: Oh yes. The NetCard people sold off-line smartcard-based electronic cash systems in a number of countries, mostly in less developed countries, or countries with bad phone networks — Brazil, South Africa, Russia — and the thing led in 1995 to a patent lawsuit between VISA and MasterCard. VISA had bought the NetCard products and MasterCard had bought something called Mondex, which was set up by the NatWest Bank basically as a knockoff of NetCard. So there was a big case in the European patent court of who infringed on whose ideas, and eventually the lawyers got together in a huddle and fixed up what the patent pool would be, what percentage of EMV card royalties would go to VISA and MasterCard, as one does in such circumstances.
It was rather interesting because I was the expert witness for VISA and Haroon Ahmed, a professor of physics, was the expert witness for MasterCard. The lawyers would say well, Cambridge says "X" and the other lawyer would say no, our guy from Cambridge says "Y", an interesting issue of what happens when university brands are used by contenders in a case.

Yost: Can you describe the laboratory when you arrived? How large was it and what was the place of computer security research within the broader field of computer science research there?

Anderson: The computer lab was still in downtown. It was where the original lab had been set up in the 1930s, but in a newer building. It was in the Aruo Tower, as it was

called, which was built in the 1960s and is currently being refurbished. It was a dreadful 1960s building, just hot in the summer, cold in the winter; it leaked, there was condensation; substandard accommodation. But then much of the best science done at Cambridge has been done in substandard accommodation. That didn't really bother anybody.

At the time there were about 20 university teaching officers. There were two professors, Roger Needham and David Wheeler, both of whom were interested in computer security. They were joined around about then by Mike Gordon, who was interested in formal verification. Numbers have increased basically by about one faculty member per year on average since. The security group at our first meeting consisted of Roger, David, and Mike Roe, who is still a research associate here; Bill Harbison, who's since retired; he was an ICL guy who was made redundant in his fifties and did a Ph.D just out of curiosity; and Mark Lomas, who did a Ph.D. in crypto implementation and later went on the become the information security guy at Goldman Sachs. So I was the sixth person in the group and the group grew fairly steadily after that. We've now got 20 or 25. Although Roger and David have both passed away, we now have four university teaching officers that are putting most of their efforts in the computer security group, and it's been a matter of steady growth, really.

The research students arrived at the rate of one or two a year and many of them in the 1990s were looking at crypto protocols one way or another. That was the hot topic; that's where the maths meets the metal; that's where the crypto meets the computer science; that's how you take trust from where it exists to where it is needed. Surprisingly, crypto protocols have remained an interesting topic; it's a mine that has never been mined out.

In fact, I'm working right now on finalizing a protocols workshop paper with one of my research students, so it's still going strong.

The big thing that hit me in my first year was that there was a court case. There was a lawyer in Liverpool who got together 2,000 people who'd been victims of ATM fraud and they sued 13 banks for £2 million. There had been a wave of ATM fraud, which we later learned was the fault of a chap who had learned how to duplicate magnetic strip cards while working for a building entry control company down on the south coast. This chap's *modus operandi* was to either take a bank card and replace the account number on it and use the encrypted PIN, whose clear text value he already knew, to take money out of this other account number, that he'd learned perhaps by picking up a discarded ATM ticket. And that worked when the ATMs were offline and PIN verification were done with respect to an encrypted PIN on the card track. Alternatively, once the banks blocked that particular hole, what he did was to observe people at ATMs. He would shoulder-surf your PIN number; the banks at that time printed the full account number on the ticket, and with the account number and the PIN you could make up a card. And that's precisely what he did. What he would do was park a furniture van next to an ATM and set up a video camera to shoulder surf all the people using the ATM; and he would then go pick up all the discarded tickets from the cash machine's waste bin or the council waste bin next to it; and would match the account numbers with the PINs.

The banks' response to this was extraordinarily cruel; they basically said to customers who complained, "Our systems are secure, you must be mistaken or lying, go away!" So I got engaged by this company to be an expert witness in their court case and I was

basically the only person who knew how ATMs worked who basically wasn't in the pay of a bank or a bank supplier.

Yost: Was it your opinion that the banks knew that fraud had occurred but just didn't want to pay out or . . . ?

Anderson: I reckoned that a number of people in the banks knew perfectly well what was going on.

Yost: Was the practice different in the U.S. with regard to the banks paying out?

Anderson: Yes it was, and the reason for this was that in the U.S.A., the first court case was won by the bank customer rather than by the bank. There was a lady called Dorothy Judd who had got hit for a few hundred dollars of phantom withdrawals from her CitiBank account in New York. I think it was 1976, some time like that. Anyway, she went to the small claims court in Manhattan as a litigant and filed suit on CitiBank. CitiBank turned up with its experts to say its systems were secure, but the judge wouldn't hear it. He said "If I accept your argument that your systems are secure, that puts an unreasonable burden of proof on the plaintiff, which is wrong in law", and she got her money back. And that, although it was not a Supreme Court judgement, had advisory force when the Fed then crafted Reg E and Reg Z, which govern credit cards and debit cards in the U.S.A. And in the U.S.A., basically, if you're a victim of a phantom withdrawal, you just call up the bank and tell them to take it off your account. In Britain,

unfortunately, the court case that I helped in went the other way, because the 13 banks turned up with their solicitors, their junior counsel, their senior counsel, you know, 40-odd lawyers in court; and they persuaded the high court judge that it was completely infeasible to run a trial this way. They said "Surely there's no common factor in all of these complaints. They should be sent to the small claims court where perhaps the plaintiffs will win some of them and probably we'll manage to defend most of them successfully. But that can be done very conveniently and at low cost."

Unfortunately, the high court judge agreed. He was completely wrong, of course, because the following year in 1994, the perp got caught and got sent down for six and a half years at Southwark Crown Court. I was again involved in that trial as an expert witness, which is how we got to figure out a lot of what went on.

Now the academic payoff for this is I wrote a paper "Why Cryptosystems Fail" documenting all the different failures of ATM systems that have come out as we collected evidence for the case. There are a number of people who previously worked for different banks who came to us and told us the various vulnerabilities they had observed, disclosed to the management, and we also collected various case histories where ATM frauds have been solved and we knew with certainty what actually had gone wrong. And so this paper appeared at ACM CCS, the first ACM conference on computers and communication security in November 1993. And it's that that really put me on the map as an academic, because here you have this new subject, cryptography, which had been an academic subject for only 10 years, and people had been largely playing around with theoretical ideas: here's how you do a digital signature; here's how you do a blind

signature; here's how you might make an anonymous e-mail. All good stuff, but how's it going to fail in practice? What's the real engineering?

I think it's fair to say that "Why Cryptosystems Fail" is seminal because it directed people's attention to all the things that go wrong with implementation and operations, which are very easy for designers to forget. With security systems, things are different because it's hard to tell whether they're working or not, especially if you don't know that anybody's attacking you. And if somebody is attacking you, if all they're interested in is reading your e-mail — say the NSA, for example — then if they keep very quiet about what they've learned then you've got no indication that your privacy has been compromised. And so testing security systems is really difficult, and this has been a thread in my work ever since.

Yost: So it's probably at that point that you're thinking of the concept of security engineering.

Anderson: Formulating security engineering as a mission was something that came later, when I was beginning to write my security engineering book, right about 1999. The idea came out of discussions or was at least prompted by discussions with Robert Morris Sr., who was the chief scientist at the NSA and used to come to our protocols workshop once a year. He was our token person with a clearance at the workshop. He was always somewhat enigmatic but he often pointed us at interesting things to research.

Yost: Can you talk about how you came to decide on a dissertation topic and briefly describe your dissertation?

Anderson: My dissertation was just a book that contains chapters with the research papers that I'd written up to the time. It wasn't something that I put online. Everything in it appeared in a more polished form elsewhere. It was something that I wrote during the last of my terms to satisfy the requirements for a degree. I started off —

Yost: Was it unusual for students at Cambridge University to publish as much as you did as a graduate student before graduating?

Anderson: Well, in addition to the "Why Cryptosystems Fail" paper, I continued my interest in stream ciphers. I published a number of papers with attacks on stream ciphers that various people had proposed, and proposing improved versions of my own. This is just continuing the interest of 10 years previously in the early 1980s, and that was an interest that I fostered by going to conferences such as Crypto and Eurocrypt, where I met the other people who were engaged in this. And so my thesis contained some of the crypto robustness stuff that we had learned from cash machines and some of the mathematical cryptanalysis stuff that I'd been doing on the stream cipher track. And it also contained some stuff on cryptographic protocols. I tied the whole thing together by making the argument, as my thesis, that robustness in cryptographic protocols is all about explicitness, about being very careful about what you rely on, and what checks have to be

done before you'll accept the outcome of a cryptographic computation as being correct. Which is a bit of a pastiche but you know, it served the purpose.

And the other thing that I did when I was a graduate student that had some consequences was that I set up the workshop on fast software encryption, because those of us that were interested in real cryptography – that is creating and attacking real ciphers, block ciphers, or stream ciphers – really didn't have a venue to go to. When I came up with a new attack on somebody's cipher, if I sent the paper to someone like Eurocrypt or Crypto it would be rejected because by that time, the people at Crypto and Eurocrypt were mostly mathematicians who wanted papers with theorem/proof, theorem/proof, theorem/proof. And so it was great if you were trying to contribute to the complexity theoretic structure whereby Ron Rivest, Shafi Goldwasser, Silvio Micali and others sought to make public-key cryptography intellectually respectable. But if you're interested in making and breaking new ciphers it wasn't much good.

And so I got together with a number of people who I realized felt as I did, Jim Massey, Eli Biham, and others; and we set up the workshop on fast software encryption, the first instance of which happened when I was still a grad student. That started a very productive thread of research, which led ultimately to — for me — to the advanced encryption standard competition. Eli Biham, Lars Knudsen, and I had a block cipher we designed called Serpent that became a finalist in that competition. Had we designed it with half the number of rounds, the advanced encryption standard today would probably be Serpent rather than Rijndael, but I think we misjudged the mood of the community in going for a more conservative design than our competitors did. But that nonetheless was a really exciting thread of research for the best part of a decade.

After we failed to win gold at the crypto Olympics I decided to turn my attention to other things instead because, you know, there's lots more to do; there's no point in moping and feeling sorry for yourself. But later when there was a competition for the hash function, there weren't just 15 teams that participated but over 60. So certainly we built up a community of people doing practical cryptography, which is now fairly substantial.

Yost: When you were doing research in cryptography in these years were you also following what was going on in other areas of computer security research, i.e., access control and security models like Bell-LaPadula, and what was going on in the U.S., starting the National Computer Security Center, the DoD center at NSA?

Anderson: Bell-LaPadula was something that I taught but not something that I experimented on. Roger Needham had done work on the CAP, a capability computer here at Cambridge. And I have a colleague, Robert Watson, who has revised that work and is now doing a project, building a CPU with capability support. But that was never really one of my things, although I was involved in the beginning of Robert's project.
The technical threads that I started while I was a lecturer during the 1990s had to do with signal processing. They had to do, on the one hand, with attacks on hardware; whether you can get keys out of smart cards by means of probing, by means of glitching, by means of timing attacks, and of course the biggest of these was in 1998 when differential power analysis came along. But we were doing stuff even before then on things like probing. This was relevant to real applications, because people were at the time pirating pay TV cards.

One of the biggest mistakes that the industry made was making Star Trek not available in Germany because that meant that all the Trekkies in Germany had to get out their probing stations, or oscilloscopes, or whatever, and break Mr. Murdoch's smart cards, otherwise they couldn't get their weekly fix. I ended up getting a student, now a lecturer here, Markus Kuhn, who was really good at breaking into smart cards and tamper proof CPUs. With Markus and with Fabien Petitcolas, who came to us from France, we did a whole lot of work on information hiding, steganography, and copyright marking. One of the things we did there was StirMark, a program that enabled you to take information that had information hidden in it, such as an image or an audio file that had a hidden copyright mark, and then distort it in subtle ways to see if you could make the mark not detectable anymore. And this became an industry standard for testing copyright marking schemes. The interest in steganography, in covert communication, and also in copyright marking led us to start the Information Hiding Workshop in 1996.

One piece of background to that was, of course, the whole key escrow debate in America, because at one of the first cabinet meetings that Bill Clinton held, the NSA came in and sold him on the idea that he should have a key escrow policy, the Clipper chip. The idea was that no one in the world should be allowed to use cryptography unless the NSA has got a back-door key, and that basically caused the world of crypto security to explode in anger. It kind of fell to me as the only young guy who knew about these things and didn't have a security clearance, and was not somehow beholden to a government for funding, so I ended up working with people in America who were also opposed to this. People like Whit Diffie, and Matt Blaze, and John Gilmore, and so on — Ron Rivest — we produced a paper on the risks and costs of key escrow which became very highly cited. We kept on

running into each other as we went and spoke at various technology policy meetings and trying to elucidate the debate between privacy and security, which is of course, something a little bit more complex than you might think on the surface.

And it was during that time that we pointed out the important differences between content and metadata. Very often, if you wiretap somebody's phone all you'll hear is "Right, Fred, see you in the usual place in 20 minutes." The real take that you get is from the metadata that you get on who called who and when. And that's very much harder to protect using cryptography. Nowadays, we've got things like Tor, but Tor is at the end of a long series of evolution from David Chaum's original anonymous e-mail in the early 1980s and peer-to-peer stuff that we were playing around with a bit in the 1990s, and so on and so forth. So there was a whole ferment of ideas in the mid-1990s about broadening the scope of information security. Instead of just thinking about government mainframe systems, confidentiality, integrity and availability thereof – how does information security apply to a much broader and richer and more complex world of people with personal computers, where you find computers embedded in everyday objects such as electricity meters, such as ATMs, where you've got multiple competing stakeholders, where you've got companies trying to build monopolies, and other companies trying to break them?

It was in 1994, for example, that Xerox started using cryptography to tie printer cartridges to printers, so that you could subsidize printers from the sales of ink. And of course, everybody had to follow suit because otherwise it would narrow the marketplace. Ron Rivest put this neatly in the late 1990s; he said that cryptography has become like duct tape; it's just used everywhere to hold stuff together. And for the NSA to have

sovereignty over all this stuff is just simply crazy. So it was great to be around in the 1990s where all this was being worked out on a technical level in terms of stuff that appeared in crypto conferences and new algorithms and protocols, at an engineering level where people are building stuff and are looking for real-world failures in it, and also at a policy level, as we're trying to fend off the spooks that get in everybody's way.

Yost: Did the British intelligence agencies weigh in on the crypto war debate as it was unfolding?

Anderson: There have been various attempts to interfere with stuff. For example, I tried to organize a session at the Isaac Newton Institute here in Cambridge — it was an institute for the advanced study of mathematics — on cryptography, coding theory, and computer security to be held in early 1996. And the previous year when our application for this was just in for refereeing, I was visiting the Isaac Newton Institute with Roger Needham and what should happen but the director of the institute, Peter Goddard, who later became a director of the Institute of Advanced Study at Princeton, came up to us and he had in tow Michael Atiyah, who was President of the Royal Society, and Peter Swinnerton-Dyer, who is a former Vice Chancellor of the University. Peter informed us in grave tones that Dr. Covey Crump, the chief mathematician of GCHQ, had come to see him and had said "Look, Peter, I hear you're thinking of running a session on cryptography at the Isaac Newton Institute. I really don't think you should do this because there's nothing interesting going on in cryptography, and Her Majesty's government would like this state of affairs to continue." He then offered Peter £50,000 –

not as a personal bribe but as a scholarship for the Institute to use for some bright student or other. To his great credit, Peter turned Covey Crump down flat and promptly informed the university and the Royal Society of the attempt, and came and told us about it. And so GCHQ did me a really big favor there, because they absolutely saw to it that the committee would accept our proposal for a session on cryptography. And that was perhaps the second thing that helped me to get established academically, because by that time, I was tenured, I was a lecturer, and here we had the use of a whole lot of space, office space, seminar space within the Isaac Newton Institute for six months. And we could bring researchers from all over the world to work on subjects in crypto and security. And in many cases we were able to get their expenses paid either by the Newton Institute or by the various other institutions that supported it. And so that was a really formative experience, and hundreds of research papers came out of that Newton session.

Yost: Were there scholars at different career stages that were coming or was it more senior people?

Anderson: The most senior people we had were Robert Morris, who I think just retired as chief scientist of the NSA; and Gus Simmons, who was probably just about to retire as the chief mathematician at Sandia. He was the guy who designed the football, the nuclear firing codes that are carried around behind the President. And Gus was actually made a visiting fellow of Trinity, the Rothschild Professorship of Trinity. And then we had a number of academics with varying degrees of seniority and at the bottom we even had a couple of undergraduates; Markus Kuhn, who was then an undergraduate at Erlangen and

is now a faculty member here; and David Wagner, who was then an undergraduate at Berkeley, and is now a full professor there. So basically we just picked all the people with ideas and energy and enthusiasm that we'd met at either in the crypto track at Crypto or Eurocrypt, or on the security track at conferences like Oakland and CCS.

Yost: When did you first start attending the IEEE Security and Privacy Symposium, the Oakland Conference?

Anderson: Gosh, I think it must've been 1993 or 1994. I'd have to check. I suspect it was 1994 because the first major such conference that I went to in America was CCS in 1993. I previously had been to Esorics here, I think, in my first graduate year. I went to Esorics, which was France, but after 1994 I became a fairly regular attendee at Oakland, and even nowadays, despite all the extra stuff I've got to do, I manage to get on there about every two or three years. I didn't manage to get there this week but we've got a lot of people and stuff there. My student Laurent Simon and I have a paper which appears today at the MOST workshop, that's the mobile security workshop at Oakland, which is basically about how factory reset in Android usually doesn't work very well. Good practical paper. So, yes, Oakland is important. There are, of course, four big conferences now; roughly one in each quarter of the year to which you can send a good paper if you've got it, so things are much improved over the early 1990s.

Yost: This is something you mentioned at lunch — that you wanted to stay out of the secret/intelligence agency community and you were recruited. Can you provide a bit

more information about the first attempt to recruit you and what your thought process and response was?

Anderson: When I was I suppose in my first term as a grad student, I bumped into somebody from one of the agencies who said, wouldn't you be interested in perhaps doing a bit of stuff blah blah blah. It was informal, deniable, I suppose as such things are designed to be. And I spoke to Roger, who said you'd be well advised to turn them down. Roger had had that advice from Donald Davies at the National Physical Lab, who ran a section in security in the 1970s and 1980s and then retired in the early 1980s and continued as a private consultant. Donald was involved in a whole lot of early stuff, early cash machine stuff, early pay TV stuff, early block cipher and hash function stuff. His advice through Roger was that if you give the spooks a chance, they'll get you to sign the Official Secrets Act; they'll tell you a top secret fact that's of no consequence; then they will use that to demand the right of prior review of all your publications forever. So if you dance with that particular devil then don't expect to make a lot of earth-shattering publications in the security and crypto fields. This made an awful lot of sense to me and so I always took that view. I suppose I did do some classified work when I was working with Ferranti on inertial navigation sets but that's an entirely different subject and an entirely different universe. I have friends and colleagues here who have done classified stuff and got clearances. One of our engineering professors, for example, used to work for GCHQ, and he told me that when he came here he decided to become an audio processing and machine learning person so that there wouldn't be any clash with the

things that he could write on only with great care and with great difficulty. It was clearly the right decision to make.

And so one of the things that concerns me is the push recently by GCHQ to direct security research in the UK. We now have a scheme of "Accredited centers of excellence in cybersecurity research" for which institutions can sign up, and this gives them some privileged access to various almost inconsequential funding streams. But there is a push on people to come within the tent to come and sign NDAs.

My observation is this: that those people who have signed up often don't publish anything of consequence. Now there are circumstances in which this may be okay. My mentor, Roger Needham, for example, retired after 45 years at Cambridge and became head of Microsoft Research at Cambridge for the last five years of his life. And he also became a member of the Defense Science Advisory Board. So he got his security clearance, he got his big green safe in his office, everything that went with that. But by then he'd already notched up a substantial number of publications and was focused on other things, namely setting up a computer science research lab for the second time in his life. So it's understandable for someone to go over to the dark side once they're sufficiently senior, but to do it when you're junior basically precludes a proper academic career.

Yost: One thing that I meant to ask you earlier is to describe Roger Needham as a thesis advisor and what he meant to you as a mentor.

Anderson: It was absolutely transformative. He was almost like a second father to me. He didn't have set office hours, set ways of doing things by means of formal supervisions. But we'd forever be bumping into each other in the corridor or at tea; the lab had tea and coffee mid-morning and mid-afternoon and that's where people go socially. We'd have security group meetings every Friday afternoon, which we still have, along Quaker lines as Roger was from a Quaker household in Sheffield. And there was the pub. Roger liked his pint and so very often around about twelve or twelve thirty there'd be a knock on my office door. 'Shall we go over to the Eagle?' We'd go over to the Eagle and we'd have a pint and a pie for lunch and talk about whatever interesting stuff was around or whatever had come up. This was just his standard way of operating. I've mirrored this in the way I mentor my research students. I'm typically sitting down for a sandwich with them about three days a week. I might go to college one day a week and do something else one day a week, but having lunch one day a week with one's students, as far as I'm concerned, is just a standard way of operating.

What else can one say about Roger? He was extremely bright about the mathematical and systems side of things. He started off as a mathematician and like me, failed to make it as a pure mathematician but found his métier in computer science. He said in his first year at Cambridge he got a first, in the second he got a 2.1; in his third he got a 2.2 and it was a good job he switched to a diploma in computer science and found that he was good in programming or he would've ended up with an undistinguished career. Roger distinguished himself, I think, by emerging as one of the two leading programmers on the big projects in the lab, the Atlas and the Titan. These were early time-shared operating systems. He also went over every summer to DEC Systems Research Center and before

that, to PARC, where he worked with the scientists on all sorts of cool new stuff that came out. So he was a hands-on guy.

Despite being a math and CS person he was also a real sweetie pie. He was sociable, extrovert, gentle; he had all the social skills that many computer scientists lack, and he had a life-long interest in politics. He joined the Labour Party just before Tony Blair was born. He wasn't able to be a councillor for the Labour Party because he lived west of Cambridge in a rural constituency, so he served for about 30 years as an independent councillor. And what else can one say? He had a great way of coming up with pithy sayings. He was great with language, and he would go and think about a problem for two or three days and then just come up with the phrase that summarized it. For example, when we were looking at some cryptographic protocol that failed because a check wasn't done, he said "Well, this protocol has been optimized and as you know," he says, "optimization consists of taking something that works and replacing it with something that almost works but is cheaper." People remember dozens, hundreds of Roger's sayings of this kind.

He followed Maurice Wilkes as head of the computer laboratory, then he became Chair of the School of Technology (the equivalent of a dean), then he became pro-vice-chancellor (like an American university provost), and then he retired and ran Microsoft Research. So he had a successful academic career, and he could've been vice chancellor if he had wished it when Alec Broers was being hired as vice chancellor about 20 years ago. Alec suggested that Roger should have the job and Roger said Alec should have the job. Perhaps he got his way.

Yost: Did he enjoy upper academic administration?

Anderson: No he didn't. When he moved from being dean to be pro-vice-chancellor he remarked to me over pints in the Eagle that it was just a matter of going from shoveling cow manure to shoveling horse manure. I think he took the view that he owed a huge amount to Cambridge because he showed up at the age of 19 and never left, and it had offered him this glittering career that enabled him to make a contribution to both science and technology; to become a Fellow of the Royal Society; to become a pro-vice-chancellor; and en route he made a decent amount of money, I believe from involvement in various companies and things that happened in the 1960s and 1970s. So he felt that it was his duty to pay it back.

Yost: You moved very quickly to a high echelon in the computer security field but I'm wondering if in those early years there were any other important mentors in addition to Roger.

Anderson: Well, David Wheeler was also important, and you've got his oral history. David was a research student when the EDSAC, the world's first proper computer, was turned on here in May 1949, and so he had the distinction of being the world's first programmer. David was a guy who remained interested only in the math and the tech. He wasn't interested in administration. He didn't have Roger's extrovert personality or gift for networking; he just sort of stayed in his office and wrote programs and had ideas and he'd come around and discuss some idea with you. He was relatively poor at explaining

his ideas. We always used to say that when he spoke it was both compressed and encrypted. [Laughs.] But it was worth persevering. And yes, he was a great support in the initial years when I was getting ramped up to do work on conventional cryptography with block ciphers and stream ciphers. He would come to the fast software encryption workshops and he would just point out well, couldn't you just do this, couldn't you do that? David had lots of things he never got around to publishing. Perhaps the biggest example of this was the fast Fourier transform. Somebody would come to him with a problem, he'd think about it for a minute or two and tell them the solution, and then just wander on. He wasn't hungry for fame and recognition so his career had a fairly flat trajectory, but those who knew him really appreciated his work.

In addition to that, I got a lot of support from peer colleagues, from people who were research students at the same time as me or were my research students shortly afterwards, and people with whom I collaborated – because I've always adopted Roger's philosophy, which is the Bell Labs philosophy, which is the way to get good research done is you find the brightest people you can and then let them do whatever turns them on. So that's the algorithm that the lab was always trying to follow in admitting research students, and it's the algorithm that I've tried to follow.

And this is why I've done so much stuff in different fields. You get a new research student coming along; Markus Kuhn comes along, he wants to drill into chips and break into hardware, so that's fine, let's do that. Later Sergei Skorobogatov did the same thing with lasers, so that's great. It means that as a supervisor you're constantly running to catch up with your research students. Suddenly you've got to go read a book about signal

processing or whatever, but I like this as a lifestyle because it stops me from getting in a rut.

There's lots of people in academia who are still doing what they did for their thesis and you meet somebody and you say "Well, John, what are you doing?" "I'm still doing superconductors." "You were doing that 30 years ago!" "Yes, I'm still doing superconductors." Well, hey, come on! When I was working in industry I would normally change jobs every year, six months, two years, and then the three-year stint with Barclays was the longest I ever did. That was probably too long; I was getting stir crazy towards the end. In academia, if you're working in a problem space, then as your problem keeps changing you keep having to find or create or forge new tools to deal with it, and that's been the story of my research career.

If on the other hand you're driven by technology push or theory push, if you're someone who sits on a large pile of theorems about higher order logics, or category theory, or whatever, then all that you can do with your life is to add more theorems to the pile and try and look for more applications, which often don't come or are rather artificial. So I think that the style of engineering research where you let yourself be driven by real problems is the way to do it. This was also Roger's view. One of his sayings was, "Good research comes from real problems."

Another piece of guidance he gave us was: "Good research is done with a shovel, not with tweezers." He gave me this advice when I messing around trying to find yet another variant of some public key scheme or other. And he said "Look, when you find yourself down on your hands and knees with a tweezers picking up the crumbs left by 200 mathematicians that trampled the place flat already, you're in the wrong place. Leave that

to the guys from the University of Mudflats and go and find a big pile of muck, a big pile of steaming muck and drive a shovel into it." And so that's what I tried to do throughout my career by tackling problems like copyright marking, hardware tamper resistance, the economics of security – because every so often you just get the idea that hey, there's a new problem area here that people haven't even thought about. Let's have a go, read the relevant subject books, let's go and hire an expert. Let's see if we can find some large game animal in that bush.

Yost: The economics of computer security is one problem area that you have drawn on theory from many different fields. Can you tell me when you first started thinking about the economics of computer security and really made a decision to tackle it and really, in essence, founded or co-founded this research specialty?

Anderson: I'd come up across economic factors once or twice before and there's some mention in the Why Cryptosystems Fail paper of the institutional-economics side of how people behave in organizations like banks. But the moment for me was an Oakland conference. I think it was Oakland 2000 or maybe 2001; I can check that because I wrote a paper on this which appeared at ACSAC last year. What happened was that I met Hal Varian, and Hal at that time, as you know, had just written a book *Information Rules*, which explained in great detail how network effects were shaping the landscape of our industry and the innovation environments in Silicon Valley. So I met Hal – of course, Berkeley's local to Oakland – and we started talking, and we just kept on talking, because he was interested in why it was that people didn't buy enough anti-virus software.

Perhaps he was consulting with one of the companies, I don't recall; and I explained to him that in the old days, people would be prepared to spend £10 and get Dr Solomon's software to stop their own hard disk being trashed. But since about 1997 or 1998, if someone infected your PC, they would use it to attack Amazon's web server or to do a denial-of-service attack against Panix in New York, or whatever. So why should you spend money to stop your PC doing something that didn't harm you all that much? Once viruses were doing real work it was not in their interest to harm the host. We started talking about this and he described the ideas he had, the work he had done on network effects. I was vaguely aware of network effects because Bob Metcalfe was in town here at the lab for a few months in the mid-1990s, and I had met him.

But one of the things I'd had niggling at the back of my brain was why was it that in the U.K., banks spent more on security and fraud put together than banks in the U.S., even with more banks in the U.S.A. are more exposed in liability terms? Reg E and Reg Z don't allow them to dump fraud liability on to their customers the way banks can do here. If economics were simple, you would expect that it would be the other way around; that banks in America, who bear the full liability for fraud, would end up spending more money on it. This led to a fascinating discussion on adverse selection, moral hazard, and so on, and basically I was hooked. Each of us had realized that the other had parts of the answer to the problem. So we started a long and fruitful exchange of e-mails. Hal gave me a copy of his books; the *Information Rules* book and the *Intermediate Microeconomics* book, which I read and then read again carefully. And I was in the process of starting to write – in fact I was about halfway through the process of writing – *Security Engineering*.

Now this book of mine has as its genesis the lectures that I was teaching here, so the first part of the book is roughly speaking my second year course, and the second part of the book is roughly speaking my third year course. The last part of the book is roughly speaking the stuff that I've written about policy in the context of the key escrow debate. So I basically put all the stuff together and wrote text to cover the gaps and hammer this into a cohesive whole, so that people could have a guide to the subject as I saw it at the time. And I saw it as being important to reach out not just to students but to practitioners, to Dilbert in his cubicle that actually wants to know how the hell to fulfill this latest mad mission that he's got from his pointy-haired boss.

Yost: Did you see the book as equally for both groups—students and practitioners—or a bit more for one than the other?

Anderson: Yes. Some back history there which I can come to in a sec. Anyway, so at the time, the stage I was at with the book was writing all the glue text, turning it into a cohesive story from a bundle of bits and pieces of text that had come from individual security lectures or research publications. And I found as I worked through it and did successive passes of polishing it and reorganizing it, and turning it into a book, that more and more the glue text that I was using was text about incentives. And this, as I've come to realize, is absolutely central in, for example, how payment systems fail. So you've got a system that involves 20,000 banks and a hundred big suppliers, and several million merchants, and a couple of billion card holders, and it takes 20 years to change it. And although VISA and MasterCard may think that they're setting standards, they're as much

buffeted by the storm as anybody else. The only way that a system like that becomes secure is if the incentives are such that all the banks, merchants, users and so on get to an equilibrium which you can live with. And if the equilibrium is something you can't live with then you've got a disaster on your hands, as with CB radio, where everybody used it and so it became unusable.

Once you've seen that, you start to look at the second-order stuff as well. For example, we did some work afterwards on why it is that certification of payment terminals is often perverse. This is about ten years later. In 2010, 2009, sometime like that — anyway — some bad guys put a dodgy terminal in the BP garage in Girton near Cambridge and over 200 people locally found that money was being taken out of their accounts through cash machines in Thailand. The banks initially said "Our systems are secure so it's your fault; you must have gone to Thailand when we weren't looking or you must have colluded with the criminals." And eventually there was an outcry and the banks relented, and paid people what they had to. So we investigated what had happened with terminals and it turned out the terminals were trivial to tamper with. They had been certified as supposedly tamper proof, but the certification process was a sham; it was an imitation of common criteria certification rather than the real thing, and was being run by VISA at the behest of the banks. Think of the incentives facing a manager say at the Royal Bank of Scotland as he considers whether to buy a million terminals for his merchants, which are secure and cost £100 each, or a million terminals which are optimized — as Roger put it — and cost £80 each. So you're going to spend £20 million of your budget decreasing card fraud in the U.K? So fine, how much is the bank going to benefit? Well, if it's going to cut card fraud over the relevant period by £100 million, then how much will that

benefit the Royal Bank of Scotland? Well, they've got eight percent or so of the issued cards, that's £8 million, so it was a very simple calculation. You use the cheap terminals and you let the cost of the fraud fall on the other banks and merchants and card holders. But it's actually worse than that, because if you're the guy in charge of Streamline, the Royal Bank of Scotland's merchant acquisition program, you're spending that £20 million more to save money on the budget of your bitter rival, the general manager who's in charge of card services, with whom you are competing for the retail managing director's job next time it comes up. So the institutional economics are also against doing the trade. The more we study this the more we come to the conclusion that to a first approximation, all big failures of big complex sociotechnical systems are about people getting the incentives wrong.

And that was something that we just started to grasp in the Claremont car park 15 years ago, as Hal and I were sitting there. We talked and talked and talked and we missed most of the Oakland reception. I was vaguely aware that I should go and have a glass of wine and say hi to all the people in my field, and Hal was vaguely aware that he should go home to his family and have dinner, but we just sat there for it must have been over an hour in his car just talking all these things through and realizing, you know, wow, yes this fits, then that fits, the next fits.

And so what happened then was that I carved out the security economics text from my book and turned it into the first security economics paper ["Why Information Security is Hard – An Economic Perspective"], which then really got legs and got the field going. And then the following year, I went and did some sabbatical time at Berkeley and we held the first workshop on the economics of information security. So unlike many good

ideas in academia, I can tie down to that particular meeting sitting with Hal in his car in the car park of the Claremont at Berkeley.

Yost: Had you taken any courses in economics in school or were those discussions with Hal an introduction to a number of these topics, like moral hazard and game theory?

Anderson: When I worked for Barclays I went to night school for my banking exams for a year or so and economics was one of the things that was taught, but it wasn't economics that was relevant; it was macro. It was all about national accounting and IS/LM and all that sort of stuff. So I didn't come across game theory and network externalities, the things that are actually relevant to the field, until I read first Hal's popular book and then second, his economics textbook. When I did so, it was a complete eye-opener, and of course if you know math, then you should be able to learn economics pretty quickly. It's like physics, you know, it's a bunch of equations and a bunch of experiments that confirm them. If you don't find equations difficult then you just roll your sleeves up and read it.

Yost: Do you want to take a quick break?

Anderson: Yes, why don't we have coffee?

[BREAK]

Yost: When we left off, you were just starting to talk about the early years of your work in the economics of computer security. At what point did you decide to form a conference in this area and can you tell me about the context and back story with that?

Anderson: It just occurred to me, as I said, while I was writing my book that this was a really important missing piece of the picture. All through the 1990s we had thought that the Internet was insecure because it didn't have enough features, it didn't have enough crypto, or authentication, or filtering, or firewalls, or blah. And so a whole bunch of us went off and developed more crypto and everything. That didn't change anything; the Internet kept on being insecure. So hey guys, maybe we're solving the wrong problem, maybe the real problem lies elsewhere! And so once I realized that security economics was something that people needed to explore and understand, I thought "Right, I'll just make this, to the greatest possible extent, a research focus." Now I already had commitments, of course, had other research students doing other things, but I did have a year of sabbatical coming up because I was first appointed as a university lecturer in October 1995, so you're allowed one year off every seven at Cambridge and that meant that from September 2001 to September 2002 I was entitled to go off and do what the hell I liked. So I did. The lab was just moving into this building at the time and I reckoned it would be a pain getting through fixing all the snags and debugging everything, so it was an opportune time to take sabbatical.

So I arranged to make four trips during my year off and because of family reasons I couldn't just go away for the whole year. So what typically happened was I'd go away for a period of two months, my wife would come out for two or three weeks in the

middle, and we'd arrange dog sitters and stuff like that. But anyway to cut a long story short, I did a stint in Berkeley in the first term, that is, in the fall and early winter of 2001. The purpose of that was to work with Hal on thinking through what security economics should be about. And then I did a stint in MIT in late winter of 2002. I did a short trip to Singapore just for a couple of weeks. And then I did another long stint at Berkeley basically in the spring of 2002, or early summer, which ended with the workshop on the economics of information security, which was held in Berkeley in 2002.

On my first trip to Berkeley, Hal and I started plotting to run the workshop. As dean of SIMS, the information school at Berkeley, he could lay on the auditorium and all that sort of stuff so we could do it at almost a zero cost basis – invite people to turn up and pay their own traveling and accommodation. We did make offers of expenses, which Hal paid for out of his department slush funds, to one or two people we wanted to get there. And so I spent time speaking to and e-mailing with various people I knew in the research community who I thought might be open to starting to think about this and Hal similarly started shaking down economists.

So we got together an initial hard core of people who thought that yes, this was important and overdue; Andrew Odlyzko, whom you know; Bruce Schneier, you've probably heard of; there's also Jean Camp, who was at that time at Harvard and she had independently come up with the idea that internet insecurity was like environmental pollution and was starting to talk about how do you deal with it with an equivalent of carbon tax or an equivalent of cap and trade. She also was talking about the need for markets in vulnerabilities; that led in fairly short order to the establishment of iDefense and Tipping Point, companies that actually traded in vulnerabilities. And the people that Hal brought

in included Marty Loeb and Larry Gordon from the University of Maryland, who are basically professors of accounting. Their background is institutional economics, conflict of interest, that sort of stuff; and they're interested in using models from financial theory to model investments in security. We also, just by calling around the Bay Area and speaking to everybody we knew, managed to get hold of a number of people from assorted companies, and research students from the Bay Area who were vaguely interested in this. We put together, gosh, it must've been something like 45 people for the first WEIS [Workshop on the Economics of Information Security] and that was a real fun event. We realized that we had a winner and that it would become an annual event.

Yost: At that first event, besides computer security scientists and economists, were there also psychologists and other social scientists?

Anderson: The psychology came later. The track that led to that was Alessandro Acquisti, who at that time was Hal Varian's research student, who started out as a straightforward economist but over time became more and more interested in behavioral economics. Alessandro was also interested in privacy. He's become the leading writer on economics of privacy worldwide. Many of the ways in which privacy goes wrong require the behavioral aspect for a convincing explanation. Why is it, for example, that if you ask random people in the street in London or New York, now or 10 or even 20 years ago, about a third will answer that we don't care about privacy; a third will answer that they'll give you some information in return for some benefit; and a third will say you'll never get my information, you know, "You'll get my PGP key from my cold dead fingers?" Yet

this distribution of privacy preferences that people state, isn't the distribution that's revealed by people's online behavior, where almost everybody will give you their medical records for the free cheeseburger. So why do people say one thing and do another? This has been one of the research problems that people have worked on off and on for the past 15 years. There's all sorts of theories. Alessandro has done an awful lot of work about its being highly context sensitive, where you make privacy salient to people when you ask the questions or do the experiment. I reckon that a fair part of it is just knowledge and experience. People like me who have been online since before most people heard that online existed, and who understand how computers work, are very, very leery about using some online services. I've always, for example, considered everything I've put on Facebook to be published. But people who come online for the first time don't necessarily make these assumptions and so there's a long learning curve. Now, again, this is something that Alessandro has done research on and he's done fascinating graphs where he has graphed the privacy preferences set by CMU students over the past seven or eight years. Facebook, because Facebook resets and re-engineers its privacy settings every year or so, when too many people opt out they change all the privacy settings opting you back into advertising, and you have to go back in and learn how to use the new privacy settings and opt yourself out again. And yet the graph of how many people opt out of stuff is an interesting sawtooth waveform where the proportion of opt-out keeps on rising despite the frequent resets. And this basically says to me that despite Facebook's best efforts, they are like as not training up a population of users to be suspicious of their privacy guarantees, with the effect that a greater proportion of their

users every year make some effort to protect their privacy than the previous year. And this is just, if you like, knowledge and experience catching up with reality.
But no doubt there are many other factors in play, as well, because it is strongly in the interest of online firms to deceive people in order to get more personal information out of them. So we see, for example, when we look at the various social networking sites that none of them has a privacy policy on the front page. Why? Because if you make privacy salient then people will be wary. But if you go to a website that says "Come share your photos with your friends," then people don't stop to think who else you're sharing it with.

Yost: Can you talk about the reception of the broad computer security research community to the seminal article you wrote and the launch of this event that became annual? Were people ready for this to become a main area of computer security research or were a number of people either skeptical or critical?

Anderson: My first security economics paper, the one that came out of my security engineering book, was first given as an invited talk at SOSP in Banff and there the audience was 200 or 300 people and, the program committee having invited me, I had them in my hand for 40 minutes or whatever it was. I got comments of interest from a number of people afterwards. The paper had also been accepted as a refereed paper at ACSAC, the application security conference that December. ACSAC was a multi-track conference and I ended up in a small track in a small room in a hotel. I think it was in Phoenix; you could look it up and check. And I think there was something like 16 people listening to the talk, and I thought that was rather disappointing, you know, this is the

subject I'm going to be organizing the first international workshop on in six months' time and here only 16 people turn up. (In the end, we did better than 16 for WEIS; we did over 40.) But, yes, starting a new idea or a new discipline is always hard. I've been involved in this several times, with the beginnings of fast software encryption, with the information hiding workshop, and then with WEIS, and more recently with SHB, the workshop on security and human behaviour. But we've actually been lucky when you look back on history and see how long it's taken some other people. Faraday invents electromagnetism in the 1820s and it's the 1880s before it grows to industrial scale; 60 years. James Clerk Maxwell does his equations in 1861 and nobody pays any attention until 1885 with Heinrich Hertz' experiment. And then in 1907 when Lord Kelvin dies, he still doesn't believe in Maxwell's equations 46 years after the event. So the world seems to be getting better, in that it doesn't take you a whole generation to get a new idea across the way it used to. That's perhaps one thing for which we can thank the Internet, social media, mobile phones, jet travel, a more competitive academic environment, and large numbers of bright young Ph.D. students looking for something to do. You know, we're blessed in comparison with previous generations.

Yost: Have funding sources been receptive for this area of research?

Anderson: Yes and no. Most of the funding that I've got through my academic life has been accidental. Companies come along and offer money, or research students turn up who've got funding attached to them for scholarships from their home country governments. I've only had about three or four big research grants in my life. The

security economics community did get some big U.S. government funding about five years ago, after NITRD declared a "leap year" in security research, which meant that there should be a leap forward, and security economics was one of the topics that they chose. I was a beneficiary of that somewhat indirectly because one of the big grants went to CMU and we became a subcontractor on that, on the behavioral economics of cybercrime. That's still paying a couple of my post-docs and should be running an extra year from now. The U.K. funding has been a bit patchier. One of the research councils funded a second center in security economics at Bath and Bristol, which didn't go so well because the key guy, David Pym, decided he wanted to move to Aberdeen instead. He's now at UCL. We have other people in Europe who do security economics work. As well as David there's Angela Sasse at UCL, Michel van Eeten at Delft who's hosting WEIS this year, there's Rainer Böhme at Innsbruck. But by and large we've only got a very small security economics community in Europe. The great bulk of it is in the U.S.A., which is of course, the mother ship as far as computer science research generally is concerned. But America also has the advantage that you have many business schools that specialize in IT. It's not just SIMS at Berkeley, where Hal Varian was, there must be something like 30 or 40 universities now have got a business school dedicated to the computer industry, and this tends to provide a natural home for people who work in security economics.

Yost: Do you know if NSF's Trustworthy Computing, the prior name of SATC, was funding much in this area?

Anderson: No idea.

Yost: Have many legal scholars also become interested in security economics?

Anderson: I've had a number of interactions with lawyers over the years, and these mostly took place in the context of technology policy stuff. During the crypto wars, for example, I was one of a number of people who set up the Foundation for Information Policy Research as a tech policy think tank, and for the first five years we actually had a paid director who was doing lobbying on issues such as what became the Regulation of Investigatory Powers Act in Britain, and the Export Control Act, and the IP Enforcement Directive in Europe. The landscape has since changed and there are other organizations doing that heavy lifting, but FIPR.org remains an active mailing list for people, including technologists and lawyers, who are interested in these policy issues.

As we come around again to a new conservative government which wants to increase surveillance powers, we get all the arguments that are not just the old technology arguments but updated for changes in technology in the last 20 years, but also legal arguments that can now draw, for example, on America's experience in wrestling with the Snowden revelations, with PCLOB's denunciation of section 215 mass traffic data collection in the U.S.A. as unconstitutional, and so on and so forth. Maintaining a conversation between technologists and lawyers is important but difficult. One of my oldest allies in this space, Nick Bohm, who's now retired but was a partner in a big city law firm, once coined a beautiful phrase: that "the arguments of lawyers and engineers go through each other like angry ghosts." Trying to find ways in which lawyers and

computer scientists can communicate in ways that are constructive and bring both parties forward is a never-ending job.

A few weeks ago I was over at Princeton for an event at which Ed Snowden attended remotely by means of a telepresence robot on the first day. On the second day we had one of the ladies from PCLOB there putting, if you like, the view of the White House on all this. And it struck me that the lawyers present, who were most of the audience, had forgotten all the arguments from the crypto wars. Perhaps they hadn't been there at the time, or perhaps they'd been studying other things. But note that these issues tend to come up again and we face the same task that we did 20 years ago in educating lawyers, lawmakers, special advisors to ministers, and so on in terms of what's practical and what's plain stupid when it comes to defining the possible frontiers between technological innovation and sensible regulation.

Yost: Can you talk about how WEIS has evolved over the years, both in terms of the growth and the number of people attending, the range of fields, and what are, in your opinion, some of the most important areas that have evolved in the last half decade or so?

Anderson: The second WEIS was at Maryland and we saw the Gordon-Loeb model with financial economics being brought to the fray. There was then one — I think that was 2004, you can check on the website – at your University of Minnesota, chaired by Andrew Odlyzko, where the big debate was between open and closed models. Is open source software more likely to end up being robust than proprietary software and why? What can you say about reliability of both models? This is something on which I'd done

some work. We'd shown that under certain circumstances you expect closed and open systems to be equally reliable, and so if you're going to make a case that one or the other is better then you have to show the assumptions behind this proof do not hold in the case of a particular product or service. So we had a good ding-dong argument, and after that I got a student, Andy Ozment, whose background was CS and international relations, who went and collected the data. He looked at bugs in BSD over a number of years and tried to figure out whether software is more like milk or more like wine. Does it get better or worse with age? So that was the early 2000s.

From the middle of the 2000s we saw a couple of things emerging. First of all, we had interaction with real policy people who were making real measurement. At WEIS 2006, if memory serves — again, check this with the website — we had Ben Edelman give a wonderful paper on the fact that websites which use the "TRUSTe" seal of approval for having a privacy policy were twice as likely to try and infect your computer with malware as systems that didn't. In this case, clearly, the adverse selection was such that the outcome of a certification scheme was entirely perverse, because if you're a respectable firm, if you're Barclays or Marks & Spencer, you don't need some third party's seal of approval on your website, whereas if you're a scammer working from a dingy apartment in Kazakhstan, then why not buy a seal of approval because you're not using your own credit card anyway?

So once we started getting real data on real scams that were going on, it was a beginning of what we would now call econometrics of information security, which is something that we're investing in more and more here at Cambridge. The other thing that started to appear in the mid-2000s was the behavioral economics of security and privacy. I've

already mentioned Alessandro Acquisti. He and a number of people, many of whom had been colleagues at iSchool, guys like Jens Grossklags for example, started producing a series of papers about the behavioral economics of privacy that caused people to sit up and take notice.

This eventually led us to start yet another workshop, Security and Human Behavior, because we'd observed that so much good interesting stuff had come out of bringing security engineers together with economists, we said why don't we try to do this properly and at scale with psychologists, and anthropologists, and philosophers, and other people from the social sciences and humanities? Now, security usability was something that I'd done some work on in the early 1990s, when we did an experiment on what's the best way to give people advice about choosing passwords and we actually got a psychologist on board there, we designed the experiment properly, blah blah blah blah, using randomized trials and so on. Then there had arisen a workshop called SOUPS, the Symposium on Usable Privacy and Security, which was looking at stuff like that. How do you make security products more usable? And there were some notable results about non-usability. For example, there's a famous paper by Alma Whitten and Doug Tygar, "Why Johnny Can't Encrypt," which showed the typical CMU undergraduates couldn't use PGP to encrypt an e-mail. That sort of thing got people looking; that was the 1990s back history of security usability. We began to realize it was a problem.

Then, with the behavioral economics stuff we began to realize that psychology was a problem also, in terms of understanding what people wanted, or helping to remind them about what they said they wanted. Why is it that people say they want privacy but then click otherwise? Is it because of hyperbolic discounting, that is, inability to anticipate the

future rationally enough? Is it because of contextual factors? Is it because of miscuing? Is it because they're deliberately misled by website designers acting greedily in their own self-interest? There's a whole bundle of issues around here that you should add to that. And then the next thing that we've begun to think about was deception. I got to know a guy here, Nicholas Humphrey, who is one of our distinguished psychologists. He's retired now but still lives in Cambridge, and in the 1970s he came up with what's known as the Machiavellian Brain Hypothesis. Now, until then, people had assumed that *homo sapiens* developed our high intelligence in order to use tools better. But when you look at the archaeological record this doesn't make any sense, because flint axes remain essentially the same from about two million years B.C., when we were definitely not human – we were halfway between ape and human – and the beginnings of the middle stone age near a hundred thousand years ago, by which time we most definitely are human. So we developed intelligence first and then we figured out how to make better stone axes second. So what was it that caused humans to become bright? Well the Machiavellian Brain Hypothesis, also known as the social brain hypothesis, is that as Africa dried up more than one-and-a-half million years ago, we moved from the trees to the plains and we started living in bigger social groups. This meant that we needed more intelligence in order to keep track of more complex social relationships; the whole social game of "he thinks that she thinks that he thinks," and so on. There's been quite a lot of evidence for all this in terms of the Dunbar number, for example, the correlation between primate brain size and social group size, which indicates that humans in our current form evolved for a social group size of about 150 people, which is about how many friends as you maybe have on Facebook. And so there's a whole lot of research stuff that

psychologists and anthropologists are working on, which tells about the nature of deception and which also starts to talk about self-deception — a very controversial topic, by the way — and which is stuff that we should be paying attention to in our field.

So there was a conversation, I think it took place at one of the financial crypto conferences, between me and Bruce Schneier, and Alessandro Aquisti; and I think it was at a UREC workshop, a usability workshop at one of the financial crypto conferences. We just sat around with a beer and said look, this isn't anything like broad enough, we need a much broader event where security engineers come together with a full range of psychologists who can talk interesting stuff about all these questions. And we set out to make it happen. We set up the first workshop, Security and Human Behavior, at MIT in 2008 and we invited as wide a range of interesting people as we could, and this has continued every year since then. It's great, it really has people sparking ideas off each other. We have museum curators who've written learned books about violent behavior in primitive society; we have anthropologists who go on peace negotiations and can tell you a lot about violence in modern society; we have the people at the core of the deception versus self-deception debate. It gives lots and lots of new perspectives on human wickedness, and I think these are going to become more important because as technological protections become better so the scammers rely on manipulating people. The security economics tell us that you're not always going to be able to fix this by relying on big service companies to fix it because the big service companies have different incentives from you and me. They're trying to sell advertising. The banks want to dump liability. There's all sorts of holes that stuff is going to fall through and there's

always going to be the incentive to provide people with user interfaces that will cause them to disclose more than they want and that, in turn, makes them more vulnerable.

Yost: What are some of the most successful attempts to rework incentives to have positive outcomes?

Anderson: Within the mainstream industry the general view is that if you want to fix incentive problems you should use auctions because if you use a second price auction, that is strategy proof; nobody is getting incentives to lie. So if it were practical you would solve all sorts of problems using auctions and this was a big area of research about 10 years ago. People looked, for example, at whether you could use auctions to do Internet routing. The answer is you can, but the overhead's a bit too high. Of course, we're familiar with auctions in the form of eBay auctions for your junk; Amazon auctions are somewhat more controlled, venders are selling you stuff; Google's ad auctions, right? Google is primarily the dominant auctioneer of advertising space, which also has a search engine attached, and a mail service attached, and a map service attached, and a mobile phone service attached, and so on. But fundamentally, they work the network effects of being the world's biggest ad auction. And so it goes. Now, that is not something that comes out of the security economics thing *per se*, but of course, within the world of security you have seen vulnerability options, which are now well established as a means of dealing with at least those vulnerabilities that aren't of interest to the intelligence agencies.

Yost: You wrote that you continually return to cryptography every couple years, can you talk a bit about that and how your research has evolved, and different types of things you've done in cryptography in the past decade or decade and a half?

Anderson: After the AES competition was won by Rijndael, and our candidate Serpent came in second, I thought "Well right, that's it, let's go do something new," and so security economics was the thing. But from time to time we keep on coming up with interesting protocol problems and regularly, every couple of years, we get another vulnerability in a payment system which turns out to be a protocol failure.
So four or five years ago we got complaints from a number of people that their stolen chip and PIN cards had been used, although they couldn't have compromised the PIN, and the banks said they must have done it, therefore it was their fault. So we investigated and we found that with an appropriate protocol manipulation you could use a stolen chip and PIN card without knowing the PIN. You put in a middleperson, a piece of hardware, which convinces the card that it was doing a chip-and-signature transaction, and convinces the terminal that the card actually accepted the PIN that was offered. Unless the background checking is very, very careful and looks for subtle differences between the card's account of events and the terminal's account of events, then the transaction will be allowed. And that was an exploit that the bad guys discovered and they stole hundreds of thousands, and people have been prosecuted and sent to jail for it. But the banks were not prepared to talk about it until we figured out what was going on and wrote about it.

More recently, we came across attacks in which the bad guys program a chip and PIN terminal so that what you see isn't what you get. The terminal can say you're paying £30 and the card is told you are now paying £3,000, and so generates an authentification code on a transaction you haven't authorized. And this is becoming an issue in dodgy businesses like clubs, lap-dancing clubs, strip clubs, and so forth. Customers are often unwilling to complain if they get charged 10 times or 100 times what they thought they were being charged, because the owners can just say to the bank 'Well, he had five girls all night and £4,000 is what it costs in our establishment to do that, and let him complain and we'll send a copy of the complaint to his wife"; a very, very dodgy sort of thing.

We came across this in a hard luck case. A British sailor went into a bar in the main drag in Barcelona and bought a round of drinks for 33 Euros, and he passed out and woke up the next morning with a headache. They'd slipped some kind of mickey finn into his drink. What had happened was that the terminal had booked 10 transactions of 3,300 Euros each, one per hour for the 10 hours he'd been out cold, and Lloyds' Bank had paid the lot. Now he sued the bank and he got us in as expert witnesses, and we pointed out multiple evidence of fraud, and Lloyd's Bank rather grudgingly paid up. And we recently found the same sort of thing has been going on in Bournemouth with a dodgy lap-dancing club there.

Finding these kinds of attacks and explaining them is important; in fact, before you came in I was just explaining them to three people from the Metropolitan Police, because the issue is if you get somebody who owns a nightclub who's in the habit of not only ripping off his customers but putting chloral hydrate or rohypnol in their drinks – so that they don't go out to the next nightclub and make transactions that would break the chain and

stop him booking the dodgy ones he's just made – then eventually, it's going to be a murder case. You're anesthetizing people who are drunk or got full stomach contents, sort of lying there on a bench without medical supervision. That sort of thing should put the nightclub owner in jail. This is, if you like, practical cryptography. That's understanding how crypto protocols fail in real life.

We also have an interest in anonymous communications. One of my post-docs for some time was Steven Murdoch, who was one of the maintainers of Tor, The Onion Router, and the thing that we're working on at the moment is how do you go about incentivizing people to provide Tor bridges. These are private Tor entry nodes whose IP addresses can be communicated to victims of censorship in places like China, which blocks access to the Tor network using national firewalling. Therefore, to use anonymous communications you have to find a private IP address of somebody who'll provide you an entry to the Tor network. And that is something that also involves us in thinking about tweaks to cryptographic protocols.

So given the ubiquity of crypto protocols in everything from payment systems to anonymous communications, anybody who's working as a security engineer comes up against this again and again and again, even if it's no longer a main research focus.

Yost: In 2007 you received a contract to do work for the European Commission on cyber crime failures. Can you tell me about both the context of that effort and also assess the impact of that?

Anderson: ENISA, the European Network and Information Security Agency, had just been set up, I think, three years previously, and their ambition is to be a European NSA. The reality is still a bit short of that; it's a small office with only a few people; some lawyers rather than engineers. One of the things that they do is bring in hired help to write authoritative reports on stuff that they want to sell through the policy machinery in Brussels. And that particular report was asking, "What is there that Brussels could and should do about cybercrime, given that its mandate is the single market, that is, promoting trade between the Member States of the European Union?" The answer is: there's quite a lot that we can do. The sort of things that we were promoting included security breach disclosure laws, which you already have in most U.S. states but which are fairly primitive and fragmentary in Europe. So if a bank discovers a skimmer on its cash machine in the States, it has to write to everybody who used that cash machine to tell them to check their account and send in a claim if there have been phantom withdrawals. But in the U.K. a bank won't do that because it doesn't have a duty to do it. When the police discover a skimmer in a cash machine they basically have to put an article in the local paper saying "If you used the cash machine at Tesco's last Tuesday please check your statements and call us." That's not how it should be.

There are all sorts of other things around software liability, and one of the points we made was it took something like 60 years after the invention of the motor car before you had Ralph Nader coming along in the States and the product liability directive coming along in the European Union. Until that time, cars were absolutely bloody lethal, right? Because it wasn't in the car industry's interest to do anything except lard them with lots of sharp lethal bits of chromium on the outside to kill pedestrians, because it made them

pretty. You know, it can take a long time to get a new technology properly sorted, whether you do it by tort laws, like in America, or whether you do it by regulation, as in Europe.

So we analyzed the parallels between product liability laws in the U.S.A. and Europe, and what's happening in software. In Europe it's very fragmented. If a device has got software in it and that device kills somebody, then you're liable. If it does damage to an individual person then you're liable. Regardless of how many times your car navigation forces you to press the "Don't Sue Me" button, that has no force or effect. If it gets your car stuck in a narrow lane in Cornwall and you have to pay a crane to get you out you can still sue them. Most people in software don't realize this. Laying all that out and laying out a path for the future about how we could get to a more sensible world, where product liability and service liability were brought into equilibrium, was one of the things that the report was about.

You see, if as a private individual you take your caravan down a small lane in Cornwall and you get stuck, and you need a crane to get you out, you can sue Mr. Garmin for the money. But if as a businessman you're towing a trailer of prawns and it goes down the same road in Cornwall and gets stuck, and your prawns go off, you can't sue because that's a business. And if instead of using Garmin you've been using Google maps, you can't sue Google because that's not a product, that's a service. So there are all sorts of strange gotchas in the way existing European law applies to information security topics and to related consumer protection. And that was something that we were trying to tease out in that report.

Curiously enough, one of the ways in which you could most conclusively push back on cybercrime is if you have good financial consumer protection, because when something wrong happens online, usually the end result is that you see a deduction that you don't recognize in your bank account or your credit card statement. So if consumer protection is robust, at least the costs are falling on somebody else, on your bank, or the merchant, or whatever, who as in a better position to do something about it than the consumer is. There are many complex issues that spread over a number of areas of government activity and coordinating these is hard.

Yost: I imagine companies have lobbied fiercely against certain regulations for computer security. Is that the case in the U.K.?

Anderson: We've never been able to make any serious progress in lobbying for better financial consumer protection. The banks are so powerful a lobby in the U.K. because they are very few, they're very large, they're very concentrated, and they give vast amounts of money to the party that happens to be currently in power. So we've not been able to lay a glove on them. They've also managed to get the U.K. government to lobby on their behalf in Brussels, and Brussels [is] very opaque. Unless you've got monopoly profits, it's difficult to even know what's about to happen there, let alone to influence it. But the NGO world finally does have EDRi, European Digital Rights, which is paying some attention to what's going on there and beginning to get some results.

Yost: Can you talk a bit about your research on protocol attacks?

Anderson: We've done a lot of protocol attacks over the years. I've already mentioned the things that go wrong with payment protocols, which have been an interest off and on for 20 years, and then this was part of the work of Why Cryptosystems Fail, and recent papers on the no-PIN attack and the preplay attack are squarely in that field. In the middle, in the early 2000s, we started a really interesting line of research because in banking the keys that are used to generate and manage your PIN, your Personal Identification Number for your bank card, are kept in Hardware Security Modules. Now these were designed by VISA, IBM, etcetera in the 1970s, in the early days of ATM networks, and I worked with them when I was in banking. It occurred to me in about 2000 that many security modules had become so complex that you could attack them by looking for feature interactions. So if you've got a security module with 500 different transactions that you can do, would it be possible for example, to discover that if you did transaction 174, followed by transaction 316, followed by transaction 46, a clear key would pop out, or you'd be able to work out a PIN on somebody's account number? It turned out the answer was yes.

How we found this out was I'd got a new research student, Mike Bond, and I said "Here is the manual for the IBM 4758, which the U.S. government in the form of NIST certifies as unbreakable. Nothing with this big and complex a manual could be unbreakable; find the bug!" So Mike went away and a couple of weeks later he came back and said, "Found it!" That turned out to be a false alarm and he turned away rather crestfallen and looked at the manual even harder. After another week or two he came back and this time he actually had a bug that we could exploit. We found one attack after another on security

modules. I got another research student, Jolyon Clulow, who was interested in this, and between Mike, and Jol, and me, we found attacks that forced the redesign of basically all hardware security modules on the market.

It was fascinating to interact with industry. So, for example, we broke the IBM 4758 and, as we do, we did responsible disclosure. We wrote to IBM and we said "We found an attack on your system; this is how it works, X and Y and Z." And to your senior people, crypto people at IBM — because I used to meet them at crypto conferences and stuff — I said "We will be disclosing this at the Oakland conference in 10 months' time. I trust this is enough time to ship an upgrade to all your customers." So fine, we wrote the paper, we submitted it to Oakland, and it got accepted. Then a month before I was due to go and give it at Oakland, I was at a conference somewhere in Germany and happened to be sitting for lunch next to IBM's head of banking for Europe, the Middle East, and Africa. I said "Well, how have you got on with fixing the 4758 bug?" And he said, "What bug?" It turned out that IBM had done nothing because the 10-month grace period we'd given them had been entirely spent with the hardware crypto people in Watson Labs arguing with the software crypto people in Raleigh, North Carolina, over whose fault it was. It's only after our paper appeared at Oakland that all of a sudden, there was a large number of downloads of the paper from our website coming from various places in ibm.com. Obviously somebody had kicked some butt and told them get the hell on and fix this. [Laughs.]

Yost: How soon was it fixed after that?

Anderson: Oh, it took months. In fact, the problem hasn't been definitively fixed and the reason for this is that VISA keeps introducing new bugs, and MasterCard, because even once the vendors reckon they've got their API sorted out, VISA will come along and say "Here is a new transaction which all vendors must support." We have several examples of this. They've come out with a new transaction, which will enable a bank to load a new cryptographic key into a chip and PIN card. The way they designed it, there was a variable length text field before the field that held the actual cryptographic key itself, and this meant that you could do a brute force attack on the key one byte at a time by giving it a certain amount of length for padding, and then trying the last byte of the free text entry field to go through all possible bytes, and see when there would be a collision with what happened when you shifted it one step along, and so on and so forth. There was no way that that transaction could be implemented faithfully by any vendor without opening up a significant security vulnerability. So we wrote a paper on this, and the vendors then went and waved it at VISA. VISA was completely uncompromising: "Management has decided, and therefore it must be."

And so anyway, my student, Mike Bond, then went and got a job at a company called Cryptomatic, down the road, who do crypto stuff for banks. And he then came up with an API firewall, a black box that you put between your normal servers and your hardware security modules, which filters all the traffic going through to prevent API attacks. So no matter how dumb VISA are in defining transactions that break the hardware security module's protection policy, you can intercept and deal with all this in the software that Mike wrote – which completely defeats the whole purpose of there being a hardware security module in the first place. It's now Mike's software that in effect is what the

banks trust. But this is the culture in banking: the audit checklist says we must have a hardware security module; we've got one; tick. As for the fact that it's no longer doing very much useful work, hey!

Yost: Is there a standard etiquette — I think you mentioned you gave 10 months for the computer security research community to inform companies of vulnerabilities — if there is, has that been widely observed or has it caused some problems with people publishing things as soon as they discover them?

Anderson: This was one of the debates we had at WEIS, responsible disclosure, and there's been a whole series of papers, many of them by people at CMU, Rahul Telang and colleagues, which have convinced the community that the optimal outcome is responsible disclosure. If you've got no disclosure at all, vendors won't fix bugs. If you get too rapid disclosure then you'll have too much exposure against stuff that isn't fixable yet. And so CERT's policy of giving vendors 46 days, or 90 days, or whatever according to circumstance, appears to be an optimal one. If vendors are going to ship new product, new software versions monthly, something like 90 days is about reasonable.
Where the wheels have fallen off is, for example, the recent Volkswagen case, where researchers at Birmingham and at Nijmegen, Holland, discovered a flaw with Volkswagen's keyless entry system. They informed Thales, the vendor, of this and just before they were due to publish their paper, Thales informed Volkswagen, who panicked and had it dealt with by their law office rather than their engineers. As a result of this, there was an injunction against the University of Birmingham which prevented the

researchers from publishing this particular paper, and that caused a lot of anger and resentment in the community just from the sheer unprofessionalism of the industry. Now, the advice that I give to people is, when you're disclosing a vulnerability, never disclose it to the company; disclose to its regulator. I've had one or two brushes with lawyers as well — with bank lawyers — so nowadays when we discover a vulnerability in banking systems there's an absolutely straightforward path to take, because I disclose to the Financial Combat Authority, the U.K. regulator, I disclose to the European Central Bank, the European regulator; I disclose to the Federal Reserve; I disclose to the U.K. police. If there's a third party vendor involved, then we'll disclose to the key vendor. If it's a vulnerability in an Android phone, we'll tell the people we know at Google, but we won't necessarily tell everybody at Sony, and HTC, and so on because there's too many of them and they're too difficult to get hold of. So that means the banks who are going to be affected by this — RBS, Barclays, Lloyds, whatever — learn about it from their regulator rather than from me. And therefore, the likelihood that they'll tell their lawyers to go and come after me is pretty well zero, because it would be slapping their own regulator in the face, which they don't want to do. So my advice for people like University of Birmingham is next time you hack Mr. Volkswagen's system, don't tell Volkswagen, tell TüV, which is the technical regulator in Germany, where Volkswagen is based.

I should mention that my wife's coming around at quarter past five. You said you're only going to take a couple of hours or so.

Yost: I'll wrap it up.

Anderson: It's just that we're taking our kid out to a show tonight.

Yost: Sure.

Anderson: You're welcome to come around tomorrow morning if you want.

Yost: No, I'll be able to get through. What do you see as the most important lessons for students today that learn about the history of computer security that they aren't currently learning about?

Anderson: The most important thing to learn that most people don't learn, is the trick of adversarial thinking. How can you think like the bad guy; how can you think two or three moves ahead. Anybody who plays chess or go or any game like that should learn this, because to be a nontrivial player you have to be able to think up good moves for your opponent rather than just dumb moves. Most people can't do that. How do you develop this skill? I don't know.

For a researcher the main skill is creativity. How do you spot that? I don't know. You admit four research students, all of whom have beautiful CVs; they've won every star and prize in sight; and one of them will be truly creative and come up with hundreds of research papers over their early research career and end up a professor. And two will kind of coast along; you know, they'll do good work where you tell them where to look but they'll never be academic leaders. They'll find a good position in industry and be very

competent engineers. And one will be just unable to do innovative stuff; they'll just do what they're told. Finding creativity's the real challenge. And, of course, that's a challenge for industry, and also for the bad guys; how can you come up with new criminal business models that the FBI is not in a position to block?

Yost: What do you see as the greatest challenges to computer security moving forward?

Anderson: What's happening is pretty well everything we do as a species is moving online. And this means that the tussles are moving online as well, the competition and the conflict. If you write a successful system, as Mark Zuckerberg did for example with Facebook, then all of a sudden it's no longer about figuring out who's popular in your Harvard dorm and who gets to sleep with whom. All of a sudden you get hundreds of millions of users and the Indians and the Pakistanis are having a go at each other; the Israelis and Palestinians are having a go at each other; and you have companies that are bitter commercial rivals who are similarly trying to undermine each other. How do you deal with this? How do you set up structures where that can be accommodated and if necessary, policed? How do you deal with this in a globalized world where national governments are 15 years behind the curve, at least, where they're mostly technophobic, and where police forces are usually unwilling to do anything that involves foreigners because invoking mutual legal assistance or phoning up Interpol is just too hard?

Yost: With this project, as I mentioned over coffee, we have under-sampled interviewing people on cryptography and it also has had a U.S. focus. This is the only trip I took

overseas on this project. Who are the computer security pioneers from the 1980s and 1990s that you feel would be most important if we expanded our next project to include the U.K. and all of Europe?

Anderson: The main U.K. pioneers I'm afraid have all passed on: Donald Davis, David Wheeler, Roger Needham. They were the big hitters of the men of my father's generation. On the cryptography side there are some very strong people in Israel; Adi Shamir's no doubt the first person you want to talk to there, although there's others such as Eli Biham, with whom I worked on crypto. On the computer security side, most of that is on the industrial side, people working at places like Checkpoint, NDS, and so on. On the academic crypto side, you want to speak to Bart Preneel at Leuven; Christof Paar at Bochum, who's both crypto and security – in fact, Bart is as well; Christof is interesting because he's *the* expert on security of vehicles. I've done a bit of that in tachographs, but he's the guy who really understands all these remote key entry systems, and smart car mutual vehicle avoidance systems, and so on that are being designed. In EPFL [École Polytechnique Fédérale de Lausanne], I suppose you may want to speak to Arjen Lenstra, who's one of the pioneers of computational number theory; of factoring larger and larger and larger numbers. The top guy in France is probably Jacques Stern, at the Ecole Normale Supérieure. He's more a cryptographer than a security guy. I'm just trying to think who else there is in the security space in Europe. Angela Sasse at University College London is one of the leading people who works in security usability. She's done amazing stuff on, for example, the security budgets in organizations. The average person is prepared to spend only so much, so many hours per year on compliance. And how do

you see to it that that is used effectively, and how do you ensure that you don't displace useful stuff with whichever fashion has just grabbed your chief financial officer's ear from his accountant? How people behave in big organizations is important, and it's not studied enough. So she's a person to speak to. And then there's a number of people in specific areas; I know a number of people doing medical confidentiality but that's perhaps too specialized. Overall I'd say the overwhelming majority of first division security researchers are in the U.S.A.

Yost: And finally, are there any topics I haven't covered you want to discuss before we close?

Anderson: Wow. We've covered the crypto stuff, the crypto engineering stuff, signal processing, tamper resistant security, economics, psychology. That's a fair canter through what I've been doing in the past 20 years. I suppose one outside crazy thing I'm doing just because it's one of those things that's low probability but very high payoff if it does work, is looking at the foundations of quantum computing, because we see enormous amounts of money spent on the promise of building quantum computers and they don't happen. The Snowden revelations tell us that even with the Pentagon's budget you can't make quantum computing happen. The idea that I have is that this might actually be a signal to us that people have misunderstood something about the foundations of quantum mechanics. I've been working with a physicist here, Robert Brady, on looking at whether we can explain the apparently anomalous Bell test results, which were used to justify the whole rhetoric about quantum entanglement, by looking instead for a different

foundation, namely, that there's a long-range emergent order in the quantum vacuum. Robert's thesis was on superconductors, where things are dominated by the order parameter, you know, R cos S where R is the amplitude and S is the phase, which in effect coordinates the wave functions – for example, in a Josephson junction. And if you have similar behavior in the quantum vacuum, so that there's pre-existing order, then you can get the observed correlation of cogenerated photons without having to assume stuff that travels backwards in time or faster than light. We've recently been doing some work on that. I'm also looking at how you can explain the amazing phenomenon whereby, if you get a vibrating fluid bath, then droplets bouncing on that fluid bath obey analogs of Schrödinger's equation and Maxwell's equations.

When I worked at Google we had this thing that you'd always have a 20 percent interest, something that you do one day a week just on the off-chance that it would pay off; and for the last couple of years the quantum foundation stuff has been my 20 percent interest.

Yost: Great, thank you so much.