A. M. Turing Award Oral History Interview with

Whitfield Diffie

by Hugh Williams

Sept. 15, 2017, Portola Valley CA

Williams: Hello. My name is Hugh Williams. It is the 15th day of September, 2017, and we are in the conference room of Almaz Capital in Portola Valley, California. I am here to interview Whitfield Diffie for the ACM Turing Award Winners project.

Williams: Let's start with some questions about your early life. Tell me a bit about your ancestors and where they came from up to the time you were born.

Diffie: My mother's family arrived here somewhere in time in the 1600s. She's told me 1631, but I was in contact 10 or 20 years ago with family. People who were studying the history said they couldn't find any sign of the Whitfields before 1679. The important event is that the Whitfields fought on the winning side in the revolution and received 10,000 acres in Tennessee and moved there to raise corn and cotton and whip their slaves. My father's family I know less about. I don't know about them any earlier than in the Louisiana Bottoms 1825 or something like that.

And, get back to more recent generations, my father's father was a judge and had served I think in both the Oklahoma and Texas legislatures. My mother's father was a stockbroker. My father grew up in Detroit, Texas. They say "*De*troit" down there. My mother grew up in Clarksville, Tennessee. The two of them met in Madrid. Both of them were adventurous.

My mother joined the Foreign Service. You could do that at the time without having graduated from college. She went a couple of years to Transylvania College near Knoxville. I don't think she graduated. Joined the Foreign Service, was posted to Porto in the consulate, where among other things she worked as a spy. That is, she slipped things for *The New York Times* into the diplomatic bag and some agent in the US slipped them back out. So they moved things around. I don't know anything more about it than that.

My father signed onto a freighter, jumped ship in Barcelona, took the train to Madrid, and matriculated as a graduate student in history.

My mother learned, I believe through church circles… My stepmother once told me the name of the person who introduced them, but I don't remember it and her memory's not any good anymore. But they both belonged to a small sect called

the Campbellites or Disciples of Christ. There's actually a Campbellite church in Cambridge — *(not)* in Cambridge — in *Berkeley*. I'm guessing as I say it, that though they grew up a thousand miles apart or something, they had social connections through the church. At any event, my mother learned that my father was in Madrid, put in for a transfer to go there, because I assume she maybe was interested in marrying a Yankee, didn't want to *(unintelligible)* something. She would probably be roughly 25 at the time.

They met. They got married in Paris in 1928. They couldn't be married in Madrid because they weren't Catholics. She wrote a very funny article that appeared in *Harper's* in the mid-1930s about how difficult it is to get married abroad. I remember her remark that they were taken aback because they were asked a long series of questions. They were asked if they'd ever been married before and the answer was no, and later they were asked if they had any children, which ran across their Southern sensibilities of the era.

And they were barely able to make it. My father went to Paris for a month. I don't know if it had to be the man who had a month's residence, but my father was a graduate student. He presumably could work usefully in the libraries in Paris, whereas my mother had a day job. She took a week off and for the last week of that went to Paris. She had only been there it turned out six days or something, but it was alright because the prefect could waive that requirement, but if my father had only been there 29 days, they would have had to wait a day and show they had tickets back.

In any event, in 1930 a job opened up at City College of New York. My father thought that he was hired because it was time to hire a goy. He wrote a wonderful essay called "A WASP in a Jewish Beehive" about his years at City College. They travelled as much as they possibly could from then until World War II. They moved. They had bought a house in Jamaica Estates. Donald *(Trump)*, he grew up a block and a half from me. I suppose they rented that, but they moved to Washington. My mother once showed me the hotel in which I was conceived, but I don't remember it. But my interpretation of what happened is my mother had nothing to do in Washington. My father was working on the blockade of Spain. They were there maybe two years or something. She didn't have any particular circle of friends in Washington, so she needed something to do, so she got pregnant, and I was the result.

Williams: What year were you born?

Diffie: June 5th, 1944, the day before D-Day. I like to think… I believe… Lots of people believe in astrology, that stars, celestial things have an influence on people's lives. I've conceived of something I call "terrastrology" – you really have to look around for the events that occurred at the moment of your birth here on the Earth. I, for example, was born just at about the time the first SIGINT

deception against the Germans started in preparation for the D-Day invasion. I figure that must be what shaped the course of my life.

Williams: What did your parents do?

Diffie: They were both in effect historians. My father was a historian of the Iberian countries and their colonies. His most famous … the book that established his reputation … is called *Latin-American Civilization: Colonial Period*. At his memorial, I learned that he is responsible for what would now be called the multicultural view of Latin America, that you couldn't understand it without understanding the mixture of the native civilization, the invading European civilization, and the imported African civilization.

My father always seemed to me exceedingly… That was a great surprise to me because my father seemed… That I think of as progressive... These multicultural points of view are associated with progressives… and my father was very conservative by the time I knew him. He was 42 when I was born, so I knew him in later life.

My mother was at the time maybe the foremost, most accomplished, not necessarily best-known, scholar of a woman named Madame de Sévigné, whose letters to her daughter Madame de Grignan from the courts of Louis XIII and XIV are the basic social history of that era. She died fairly young. She died at 58 of breast cancer, so she never got to publish on that subject. She had an article that would have been published in *The New Yorker* which might have drawn her some notice, but the editor changed. I don't remember who came, whether Ross came or went. It's probably between Ross and White. In any event, the new editor wasn't interested. He paid her for it but didn't run it.

Williams: What are your earliest memories?

Diffie: That of course is always hard to know. I'll tell you the first thing I don't remember, which is the bombing of Hiroshima. I was 13 months old when that happened, and that's the kind of thing that interested me and I don't remember that.

I have a memory of being carried on my father's shoulders when my parents were going to vote. My father insists I couldn't possibly remember that because I was two at the time. I rather think it occurred again sometime later and I remember it, but he doesn't remember the occasion which it occurred. But it's also entirely possible that I've imagined it.

I'm pretty concrete memory from when I was three of standing in a yellow raincoat out in the front yard in the rain.

And then my memories become solid [0:10:00] from the time I was four, because we went to Portugal. So I remember the big box that was packed that had our stuff in it that was sitting in the driveway. We didn't have a car, so that was alright. And I remember going to the boat and I remember sailing over on a freighter called the *Marine Shark*. I think that's right. I can't remember which one's over. I think came back on one, something called the *Santa Maria*, a small liner called the *Santa Maria*, but I could have the names backwards. I remember being shown the engine rooms. When the engine's not running, you can actually go inside it. When it is running, you can't go inside it but you can look through a glass window and see oil splashing all over the place and pistons running and things of that kind. I remember that. I remember they put a leash on me when we were out on deck in bad weather, so I couldn't fall overboard. And that's probably most of my memory of that.

I have a very distinct memory of arriving in Lisbon. We went to lunch with somebody whose name I also don't remember. He was one of my father's colleagues who was living there. I was thrilled to be back on dry land. I remember the palm trees. They came back to mind somewhat later. And I remember this wonderful lunch out on a veranda, semi-outdoors on a veranda of some kind or other.

Then I don't actually… I can't say I remember the train trip up to Porto, but we lived then for a year, roughly speaking, in Porto. I have lots of memories of individual things from that. We would go out. My mother and I would walk to, so to speak… We were in Foz do Douro, which is the mouth of the Douro river, and there were trollies and there was… just up the street from the Marie Castro which is where they'd been staying since the '20s. Maybe it's where my mother lived before she met my father. I don't know that offhand. But there was a washing place, I imagine natural water flowing over stones, and women were always scrubbing wash in that place.

There was sort of a drugstore. I remember going to get ice cream bars. One of the features of our walks is I'd get a little ice cream bar at some sort of a shop. I have a feeling the proprietor may have spoken English, but I don't know. At any event, I learned to speak Portuguese natively that year, but I forgot it immediately because I wouldn't speak to my parents or to anybody else if I knew they spoke English. I only spoke to pure Portuguese in Portuguese, and coming back to New York, I could have spoken Portuguese to lots of people. My parents had lots of friends and colleagues who spoke Portuguese. But I think I must have wanted to be a normal American child, so I wouldn't speak Portuguese to people.

I am very embarrassed at the memory, when I think I was seven, of walking in the Bronx Zoo and asking my father, "Are those boys talking in code?" and he said to me sharply, "No, they're speaking Spanish." I had become so disconnected from a big piece of life that I didn't… That I should have thought that way struck me as very strange.

Williams: Did you have any siblings?

Diffie: No, I have no siblings.

Williams: Tell me a bit about growing up. What did you do as a kid?

Diffie: [laughs] How am I supposed to know that?

Williams: Okay. What do you remember?

Diffie: I did lots of things as a kid. Let me say things I didn't do. I wasn't much of a sportsman. I tried, but I never… just basically didn't… I mean I tried to play baseball. I did like riding a bicycle. I really liked walking. I walked a tremendous amount. I was, as I always have been, sort of an indifferent student. I worked very hard to study things that interested me, but I wasn't good at working to study what was needed for somebody else's agenda. So all the way through school, I had lousy grades because I didn't work on the stuff that was at hand.

We lived on Kruger Road in Jamaica Estates. As I said, if – Kruger Road's one block long – it ran through the next block to touch Midland Parkway, it would come out opposite Donald Trump's house. My closest friend played on a Little League team with him in 1957 and had the strange observation I thought from my friend that Trump arrived in a limousine. Now I'm guessing that Bob rode his bike, but I could have imagined his arriving in a limousine too because his father was a doctor and employed a full-time chauffeur and had a big black Chrysler. So, as I say, that struck me as an odd observation for Bob to make, but that's what he said.

My growing up lasts I suppose… that period of my life, except for the punctuation of going to Portugal, lasts until I'm 17 when I left to go to MIT. I did a variety of things, but the kind of thing that may concern you, at about 12 I became convinced I was a mathematician. The earliest influence I can remember is Heinlein's book *The Rolling Stones* in which the very influential grandmother believes mathematics is the key to at least all scientific knowledge.

One of the things that I've come to think about lately is… Then while I was in high school… I learned half of the mathematics I know by the time I entered MIT. That other thing I learned was a very mistaken view of mathematical culture, because most of my source was *Men of Mathematics*, which is about 19th… it ends very early 20th century. Roughly speaking, I developed a vision of being a 19th-century academic mathematician. So I didn't know anything about the real sociology, either the way the universities worked in my part of the 20th century or the way they came to work by the time I would have had a professorship, or the other… I had never heard of NSA or RAND or Sperry or any of these people as people who were hired mathematicians for various different ways. Probably I had

absorbed… Though I'd read *A Mathematician's Apology*, which now seems a very silly book in lots of ways, but I had absorbed Hardy's on-the-record view of mathematics. I also studied his analysis book. A great deal of the mathematics I knew came from studying that the summer before I entered MIT.

Williams: You went to MIT as you just said. Were there others in your family that went on to higher education?

Diffie: Well, you've heard the enumeration of my family, right? I have no siblings. My father had a doctorate. My mother I think didn't finish at… got bored and went in the Foreign Service as I said and didn't finish at Transylvania. It's a women's college. I've seen it. I don't know if it's still there. It's outside of Knoxville. She was a very literature and very erudite person certainly, but not by formal education.

The rest of my family, I mean I basically have nobody left I think. I have two cousins I think on my mother's side who may still be alive, but I have not seen them since I was a child. I don't know their names. My mother had two sisters, and one of them… My mother was the eldest of three sisters, and her sisters, one moved off to Texas and one moved to Luxembourg. I remember the child of one of them, whom I found a real brat, but I don't know his name and I haven't seen him since I was eight or something of that kind. The next people alive, Joe Diffie, country singer, is my first cousin once removed, grandson of my father's elder brother Putnam. My father was the youngest of five brothers. My mother was the eldest of three sisters.

Williams: What was your favorite subject in school prior to university?

Diffie: I'm sure I would have said it was mathematics. But [0:20:00] loosely speaking. I mean scientific subjects were what I identified with, although in high school I read a whole lot of poetry and did other things. But I thought of myself as a scientist.

Williams: What was your least favorite?

Diffie: Probably whatever's required at the moment. I don't know I have a clear least favorite. I don't remember. I mean I may… probably various… it probably was different at different times.

Williams: Who were the teachers you liked the most and how did they help or inspire you?

Diffie: Well, I mean there's one. May have been dead by the time I began looking for her, although she was not… I conjecture she was 35 when I was 10, so she'll be 25… She's surely dead by now, but I began looking for her 15 to 20 years ago without success. Her name is Mary Collins. She was my fifth grade teacher.

I remember she read to the class. A great deal of my religious education comes from her reading, not directly any version of the Bible, but Bible stories to the class, something that would now be forbidden I'm sure. I considered it very important when I looked back on it, because otherwise I wouldn't have known anything about the subject. My parents had become… My father was too chicken to say he was an atheist. He called himself an agnostic. My mother called herself an atheist. They raised me as an atheist. And I wouldn't have known as much as I do about Judeo-Christian tradition without that.

The other thing she did was she taught us about cryptography. This ran a day and a half I think in the spring of the year. I got very interested in the subject very briefly. My father brought me a stack of books from the library at City College. I wasn't able to read Gaines' *Cryptanalysis*, which was one of the books. I've never read it. He didn't bring Wolfe. I'm pretty sure. There's a set of lecture notes that were done during the war. A whole lot of during-the-war events, of which the most important in our field is Claude Shannon's very clear… the paper on cryptography, which precedes … the internal memo precedes the published paper on information theory … is clearly a "Here, what can I do for the war?" piece of work, at any event. It was a couple of weeks' study of all of the several children's books. I concluded that a transposed… Now I don't remember. I think a transposed Vigenère cipher was as good as it got or something like that. Then the whole matter was dormant in my mind till I was nearly 30.

I liked several of my teachers. In junior high, I had a math teacher named Bernstein. I liked him very much. Hard… I don't have the patience to go… [laughs]

Williams: Did you have any mentors when you first started your higher education?

Diffie: That's a good question. I don't know. I don't think I ever thought of it that way. My advisor was Warren Ambrose, who was I think as good an analyst as they had in that department. And Ambrose – of course I didn't know any of this at the time – Ambrose went to school with Dick Leibler and… Blackman? … the black statistician at Berkeley. There's a cluster of very interesting people who come out of the University of Illinois at Champaign in the late '30s. The one I later came to know best was Dick Leibler.

But no, I don't remember. I think for some reason I've always been seen as promising or something like that, and so people put up with me. But I don't remember having anybody who I can call a mentor.

Williams: Do you recall any textbooks or other books that were important to you when you were in university?

Diffie: Oh, of course, by the time you get that far. I already… Let's go back further than that. Coming in high school, I said I read *Men of Mathematics*, and I read several… The head of the department looked at me down his nose. I read Lancelot Hogben's *Mathematics for the Million* and… I'm trying to think of somebody else's name. Who were the inventors of elliptic-curve cryptography? Who's the one at CRD in Princeton?

Williams: I… Oh.

Diffie: I mean it's just my Alzheimer's kicking in.

Williams: Mine too. [laughs] It'll come to me in a moment.

Diffie: Okay. When we come back to it, when it comes to you, you can blurt it out.

Williams: Alright.

Diffie: For some reason, I read a bit of Courant, but maybe I didn't read it until I was a freshman. I don't know. I certainly was aware of it. And I tried to read Bourbaki as a freshman at MIT and didn't read a great deal of it. But the book that I studied hardest was Hardy's *Course of Pure Mathematics*. So I came into MIT knowing a good deal of analysis. I think I took analysis as a freshman. I took the junior course in analysis from G.B. Thomas as a freshman.

I was a big acquirer of books. Always have been. But have I read very many, studied very many all the way through the way I did with Hardy? No.

Williams: Tell me about your current family, your children, your spouse, how and where you met, etc.

Diffie: That is well covered in… Where Mary and I met and our lives take up a good part of Steven Levy's book *Crypto*. We met in a hardware store at about 1000 Massachusetts Avenue – the place no longer exists – in the spring of 1969. Mary had a squirrel in her pocket and I think I was buying cage wire. We were both involved… Both of us, I and my partner of that time and she and her husband were both all interested in wild animals and kept exotic animals. We struck up a relationship immediately. But I am embarrassed to say… I … didn't dislike Mary. I merely saw her as a minor appendage of her husband, who was a mathematics graduate student at Harvard. But at any event, the four of us were associated, hung out together that spring and summer and fall.

In the fall, there was a very important event. I had stopped into the place of my longest-standing colleague, a woman named Harriet Fell, who's a retired professor of computer science at Northeastern. She and I went to high school together and to MIT together. I call myself her advisor on her Westinghouse

[Westinghouse Science Talent Search] project. That is to say I proposed the problem. I didn't have either the good taste or probably the ability to have done as good a job on the problem as she did. It was looking at a… the paper is called… a linear algebra with 16 units, and that subject, as the dimension goes up, properties keep falling off. That made her… She probably couldn't have afforded to go to MIT without that. That had got her scholarships and so forth.

In any event, I was at her place and I called Mary and Richard's house. Mary answered and I said, "Hi, is Richard there?" Mary said, "You know, I may not be as smart as Richard or Richard Schroeppel or you, [0:30:00] but you could be polite to me." [laughs] I was just utterly humiliated. I didn't get to see her again for another two years. I left town a couple of days later to go to Stanford, to work at Stanford.

But when I went back to the east coast, one of the things I did was I visited a mathematician named Bill Gosper, who was known for his work on the Game of Life and things like that. His family came from New Jersey. What had happened to Mary and Richard was that Richard had discovered he wasn't interested in mathematics and that he was interested in being a veterinarian. I've forgotten the role… I think Mary pointed that out to him, but Andy Gleason helped him along [laughs] getting away. He was quite good but probably not at all suited to being a mathematician. So he went to veterinary school at University of Pennsylvania and they were living in Willingboro.

So I went to see her and I was on my best behavior. That night we just went out. The three of us, four of us, five of us went out to dinner at some place in Philadelphia. And I was back in the summer of '73 and I was still on my best behavior and I got more than I bargained for and she ran off with me.

That is in the two stages how we met. We were together after that for 43 years. She died the 1st of April this year of longstanding breast cancer. We had no children.

Williams: Do you have any hobbies or other activities that you enjoy?

Diffie: Well, I don't think of it that way. If you ask me what I've been doing, usually all I can remember is staring into space. I mean I collect miscellaneous stuff. I find things look pretty or one reason or another and I get them. That's something like a hobby. But no, I don't play any organized sports, I don't have a collection in the sense of stamps or coins or something that's mounted anywhere. So I think the answer, the short answer to that is no.

Williams: My final question on this section is do you still carry your own personal chopsticks with you?

Diffie: I have them in my briefcase, yes. But the short answer is I picked up carrying them again fairly recently. It has to do with the containers. I came across a scissors case in something of mine that I used to use to hold chopsticks and I slipped them back into that. In between, I had had them in a lovely but too large bamboo box. So I left them on my dressing table and picked them up when I knew I was going to an appropriate dinner.

Williams: Do you want a break or something right now?

Diffie: No, wait a minute … but you asked … Ten years ago I discovered my credentials as a degenerate were better than my credentials as an outdoorsman, so I shifted from carrying a pocketknife to carrying a corkscrew.

Williams: [laughs] Okay.

Diffie: So that's my culinary implement that goes with me everywhere.

Williams: Let's move on then to your accomplishments. What was your first job?

Diffie: Well, I don't know whether you want to count my high school girlfriend's father having given me a job in maintenance in the garment workers' co-ops in Lower New York, but that was a perfect example of my social ineptitude, because he clearly did that so I'd have money to take her out and instead I bought math books with it. [laughs]

But my first so to speak… Oh no. Then I had another non-intellectual job the summer of… I had this feeling that I needed to work, that I needed to get… that maybe my life would be improved if I learned what working was like or something like that. So the summer of 1963, I lived with my girlfriend and three other women that I'm still extremely close to. I mean I've seen them all within the last year or so. No, haven't seen one in two or three years. But we lived up near Columbia just off of Broadway. I worked for Pepsi-Cola in Brooklyn, a job with the aforementioned Bob who played Little League with Donald *(Trump)*. His father was the doctor for Pepsi-Cola and he got me this job. I was a bottler, which means that you tend this giant bottling machine. That's what I… So first time, only time maybe I've ever had something approximating an honest job in my life.

The next summer… Summers in college, there were effectively three of them. First one I went to Berkeley, second one we stayed in New York, third one Vicky nd I went to Berkeley together. Then following graduation, I didn't get into graduate school. I had lousy grades. I was trying to dodge the draft, so I got a job at MITRE Corporation, a systems engineering consultant to the Air Force. I did that for four years before I moved to Stanford.

Williams: Can you describe what the computing field was like when you first entered it?

Diffie: Well, that's a funny way to put the question for somebody who's first entering it. I will try. One, I mean I had a particularly I think ill-formed viewpoint, so that the major influence on my thinking about it were the group that called themselves "the hackers" at the MIT AI Lab. Bill Gosper is one of those, but *(Richard)* Greenblatt, *(Jack)* Holloway, who lives around here now, *(Stewart)* Nelson. These people were very concerned with super-optimized programming, things of that sort. I didn't form until much later any clear understanding and a sense … a more sensible view of organizing the task of programming existed already then, but I didn't form it.

At the same time, almost everything I did… I learned to program on the PDP-6 in machine code, but I never really did that for real. Most of what I did was in Lisp and the main thing I worked on was a program called MATHLAB that was… My boss was a man named Carl Engelman. It was a MITRE independent research project. We lost it to MIT and finagle about how much money the Air Force was allowed to give MITRE or something like that, and Joel Moses at MIT got hold of that project and it was renamed Macsyma. It may continue to exist, I don't know.

But in the process of working on that and in particular on working on the Lisp compiler, I got to the viewpoint that proof of correctness of programs was the most important problem in contemporary engineering. That's very at odds in some sense with the kind of work that was around me and so forth. But I find I learned all sorts of things at that time. Margaret Minsky, who is editing a book on her father, asked me about having read his book *Finite and Infinite Machines*. I said no, I had never read it. I had learned that stuff I thought from looking at papers written at the time and listening to lectures at the AI Lab. So I was learning a good deal of computer science at the same time that my day job was doing things in a very narrow sort of hack programming–oriented style.

But in any event, as I say, one, the project I was on was going away, and two, I had become convinced that proof of correctness was critical. So when John McCarthy came through town in the summer, I talked to him and that started a process where he hired me to work at the AI Lab at Stanford and I moved out here.

Williams: What was your first computer?

Diffie: Oh, I took a course as a freshman that was using the *[IBM]* 709, if that's what you mean by "first computer." The computer that I did most of my work on was PDP-6, PDP-10.

Williams: You mostly answered this question I think, but were there any other projects that you worked on in the early part of your career prior to you going to Stanford? [0:40:00]

Diffie: I sort of gravitated… I never really contributed a great deal to MATHLAB. What I did, it got me into various levels of system programming. So I worked … before I came to Stanford, I began working on the Lisp compiler. I had a view that would now be called programming methodology. My belief was the reason that programs were hard to prove correct is we didn't know how to program. I formed this notion, never adequately published, somewhat worked out in my own programs, of what I guess is now called "primitive-based," the noti-… take it for the compiler, I had a ring of… if you didn't touch this set of programs, then you were going to get syntactically correct object code. That didn't mean it would be correct, but it would have the right syntax. Then in the ring outside of that, if you didn't touch, delve within that ring, then you would always have stack discipline, because these things guar-… I had this notion of having parts of the program that guaranteed different properties that you needed. And so that accounts … basically writing things at that level, assembler and compiler, was a lot of what I did in that period.

Williams: Why did you decide to go to Stanford?

Diffie: Well, I have sort of answered that question, but I decided to go to Stanford because the project I was on at MITRE was losing its money. I don't know. I never looked at the issue of going directly to MIT. I might conceivably have done that. But I wanted, I probably wanted to move west anyway. I don't remember that that clearly. I didn't want to go to Stanford. I wanted to go to Berkeley. But when I went to Berkeley in the summer of 1962, in six weeks I loved it as much as I loved New York or Paris or the south of France, which are my three favorite places. I mentioned arriving in Lisbon and the palm trees. I had the same confronted with tall… I got to Berkeley and there were tall palm trees in the same way. The place immediately felt like home. I mean nobody served me lunch on a nice veranda, I'm sorry about that, but…

I probably… I think I wanted… All the time I felt trapped in Boston and I wanted to get back out here. Stanford was a place that in some ways… McCarthy, thought the same … "Well, maybe it would have been better if Berkeley had offered me a job" or something like that. Berkeley seemed to us to be a fundamentally more interesting town at that time. But I ended up down at Stanford.

By the time I'd been here about six weeks, I understood why *The Daily Cal* called it "down on the farm." It seemed to me to be a magnificent imitation of a great university. Somehow it's gotten a lot better in the 40 years since, or I've gotten different, but…

Williams: Well, let me switch now to your interest in cryptology.

Diffie: Alright.

Williams: While a student at Stanford, you were very much a pioneer in studying this. At the time, it was a taboo subject.

Diffie: OK, so it dates back… My being a student is really… A quick summary of that period at Stanford is I was there four years, I took four courses, and I wrote four papers. Nobody knows what the courses were anymore. I can remember one of them on discrete mathematics, but I've been making a living off the four papers for the rest of my life.

In the '60s, it goes back to the period at the AI Lab at MIT, because we were in the same building that had Project MAC and had the Multics project, the Multics project being the major activity of Project MAC. Multics is the most ambitious, most important operating system project I believe of all time. It's the parent of Unix. That's the only piece of it survives I think doing any real operations. But Multics was tremendously ambitious about storage management, about scheduling, about security. I was there for hearing conversation about security issues, and my response was… Remember I was a hippy. I was very countercultural. I regarded the police as my enemy, not my friend. So my response to the discussion of file protection was "Well, what good is that? Because they'll serve a subpoena on the system operators and they'll sell you out. They're not going to go to jail to protect your files." So I had this view that now looks technically naive, I mean it turns out to be harder than I expected, but I thought, "Well, cryptography is the only mechanism you can have that will really protect you. Once you get something encrypted correctly, then they have to come to you and try to get the key out of you. You don't have other people who can sell you out." Now that became the basis for a big political argument and a lot of my career as well.

But I thought that what I was working on… I thought and still think in a sense that what I was working on at the time, sort of proof of correctness, was more important, so I didn't want to work on cryptography myself. I tried to convince various people – Harriet Fell for example – who now wish they had been convinced, that it was a good thing to work on. My vision of it at the time didn't run yet to public-key and that sort of thing. I was persuaded, I said, "Look, this is an interdisciplinary field in which you're doing statistical and probabilistic analysis of algebraic phenomena." Harriet was an algebraist, but she wasn't convinced. Had she been, she might well have done… there's a whole lot of good work was exhibited largely at EUROCRYPT in the '80s that's about that sort of sequence generator issues.

I got to Stanford. I still… I don't know. I may have talked to people about it occasionally. But in the summer of 1972… And if you don't mind I may go back

in a little bit and fill in some points. But in the summer of 1972, there was a momentous event for my life. I remember it as follows. I put it that way because John McCarthy is dead and Larry Roberts doesn't remem-… it wasn't that important in his life. But I believe I formed the view, probably from what McCarthy told me because I didn't talk to Roberts till much later… Larry Roberts went up to NSA to talk to Howard Rosenblum, who was the Deputy Director for either COMSEC or Research at the time. He held both jobs. I don't know what he was at that moment. I believe they had to have agreed that roughly Roberts said to him, "I have a $100 million a year military communications research budget. We should look at security." They must have agreed on that. I don't see they can have thought that wasn't important … But the trouble is that Roberts in that section of ARPA didn't want to support any secret work and Rosenblum didn't want to do anything else.

So Rosenblum wouldn't accept money from Roberts to have NSA people work on ARPANET security, so Roberts goes back to his office in Rosslyn, and Roberts has a great job, right? Roberts, his PIs come by with their hats in their hands and have to listen to whatever he wants to talk about. So that week, we'd gotten fired up about network security and he talked to them about network security. At that time, none of us distinguished that from cryptography. Now we'd see them as one is just a piece of the other. So John McCarthy came by and talked to him, got interested in network security, and John McCarthy comes back out to the lab and he chats us up about network security. And a few people started working on cryptography for a little while.

The most interesting one other than me is Hans Moravec, who is later the head of the robotics lab at CMU. Hans Moravec wrote a cryptographic system that was… As also done later, that one was never published, but it was called a "shrinking generator" when it was done later at IBM. It consists of two simple processors, IBM used linear shift registers, but John used congruential gen-… linear cong-… affine congruential generators, and one of them picks bits out of the other one. It's very important of this that it's not purely combinatorial. It drops some of the bits as well, otherwise there are very simple solutions to it.

At any event, John McCarthy had gotten the idea from a Russian machine called the BESM-6, [0:50:00] which has two cryptanalytic operations in it called "spread" and "gather." "Spread"… Well, "gather" takes a mask and a word and packs together the bits picked by the mask, and "spread" takes a mask and a packed set of bits and spreads them out into the word. If you're doing shift registers and things for cryptanalytic purposes, these are very attractive. So in effect, he was using the "gather" instruction for this generator.

In any event, Hans Moravec coded it, but Hans Moravec was a graduate student of John McCarthy and he thought, "If my advisor was going to encrypt something, I might be interested in what it is." It's the first instance I know of key escrow. He

had the program stash the keys somewhere that he could find. But he got back to robotics work after that.

Six months later I was still thinking mostly about cryptography and writing programs about cryptographic systems. In a way that parallels the amount of effort I spent studying *Men of Mathematics* in high school, I was reading very slowly through David Kahn's comparable-length book about cryptography. I had missed seeing that book when it came out because my late friend Paul Heckel asked me if I'd seen it and I thought I had because I'd seen another book. It's a book, it's called something about Magic. I don't remember its name, but it's about the Japanese problem and those systems. They both appeared in '67. He asked if I'd seen this wonderful new book about cryptography and I said yes, I'd seen it, and I hadn't seen the other one. At any event, in fall of '72, I began studying Kahn.

Spring of '73, John McCarthy is a bit fed up because I haven't done any work. I was working on proof checking and proving systems and things like that as well as on my own approach to proof of correctness, which would now be called programming methodology. It was about how to organize programs. This was an embarrassment to McCarthy because I was being supported by under-the-table money from NSA to work on the proof of correctness stuff. He figured NSA would not be pleased if they discovered that I was spending their money competing on their turf, which they didn't like at all.

We came to a friendly separation. I set off to travel around the world. I started out driving around the US, looking for anybody who was willing to talk about this subject, digging up rare manuscripts in libraries, and sitting around thinking about it. In this process… This is the third time or something, the second time after the conversation in Cambridge years ago, that I met Mary. I call her my first discovery. Without her, I wouldn't have accomplished anything else I'm confident. We ran off and began travelling around together. And by the summer… We were back here in I guess the late fall of '73 and through into early '74. We stayed with Les Lamport up in Oakland.

One of the things that interests me about this business is sort of how close together all these people are packed. It may be a function of computer science having been a very small field at that time.

But we went back to the east coast in the summer of '74 to collect Mary's things and hired a truck and brought them back out here. But one of the things we did was we went to IBM Watson to see a man named Alan Tritter, of whom I'd been hearing for years because Alan Tritter was an incredibly flamboyant character. Called himself the biggest man in computer science. Weighed 500 pounds. Bigger than Joel Moses. [laughs] Alan Tritter introduced us to his boss, Alan Konheim, with whom I'll still in contact. He's retired from UCSB. And Konheim was very secretive, barely told me anything. Since then he told me one thing – he

wishes he hadn't said that. He said, "Two people can work on a subject better than one. When you get back out to Stanford, you should look up my old friend Marty Hellman."

So we got back out to Stanford and I made a phone call from Oakland down to Marty's and said, "Alan Konheim said I should look you up. I've been interested in cryptography." Marty graciously granted me half an hour of his time, I think from 4:30 to 5. We went down there and Mary took off with the car and went to do something. When she called back around 5:30 or 6, when she phoned to see what was happening, we were still talking and Marty invited us to dinner. It turned out then as families we got along very well, because Marty's mother-in-law was a dog breeder and his wife liked dogs, and Mary could recognize 300 breeds of dog at a glance. It just all went… Then we worked together for four years and we became a great pain in Konheim's tush because we opposed the cryptosystem that his lab had played a large role in designing.

Williams: What is the Data Encryption Standard, DES, and why was it so important during the '70s?

Diffie: And sometimes still important. Well, let me hold forth on a much broader sort of issue. World War II, the best mechanical cryptosystem… well, the best one was called SIGABA. It's a cryptosystem of a type called a rotor machine in which rotating wheels serve as lookup tables. The reason they're built, they're organized as wheels, is it means you can rotate them and, in modern computer science terms, this amounts to an indexed lookup instruction. If you imagine a routine that's "Lookup in a table, index the result, lookup in a table, index the result," then these indices are the keys. And so all the famous… All but one of the famous World War II machines… No, two. Hagelin machines aren't rotor machines and the Japanese systems Orange, Red, and Purple are not rotor machines. I'm not actually sure about Orange, but Purple is certainly not a rotor machine. It uses stepping switches.

Now that's a very bad… that has a lot of disadvantages cryptographically. It'll only run at about 10 or 15 characters a second – it's mechanical. It's difficult to wire the rotors. You have in effect to have Top Secret clearances for the people who wire the rotors, where it's merely a simple job, but you're handling the most sensitive material. That makes it expensive. It's very hard to destroy them. If you get captured and you have instructions say, "Destroy your keying material," the rotors, it's easy to make them so they won't work but very hard to destroy them so the information about what the keys were is gone and a forensic laboratory can't find it.

So they had a whole bunch of disadvantages and at the same time, computers were coming along. In particular on this side of the Atlantic, there's a not much appreciated ... Everybody sort of knows about Colossus, everybody interested in this stuff knows about Colossus, which was a monster correlator and can be

seen as the proto-computer. It wasn't a stored-program computer but it pioneered the electronic architecture of computing machines.

On this side of the Atlantic, it's a machine called SIGSALY, which is a voice encryptor. It actually had a crypto principle, an alternate mechanism that may never have been used in anger, but the main thing was it had keying material on 16-inch long-playing records. Sixteen-inch records were the radio, the professional standard of records. Most of this is not about the cryptography. Most of it is about the 2,400-bit-per-second vocoder, which is the same thing you have in a contemporary or fairly contemporary secure phone, except the current one fit on my desk and SIGSALY is thirty 7-foot-tall racks of equipment. It filled up a room larger than these offices we're in. But the technique for designing something with thousands of tubes that way and managing timing and all of that that was going to be necessary for building both computers and cryptographic equipment were discovered there.

That led to a long series of [1:00:00] NSA systems that are sequence generators, that generate a sequence of zeros and ones that you then XOR with the plaintext. So in some sense, you can say … shrink the alphabet down to size two and you have the same basic technique you're using in World War II, but now you can do it at tens of kilobits a second and eventually at megabits.

At the same time another problem came up, which is that aircraft identification friend or foe systems were basically password systems through the Second World War. It takes a while, nine of them, because they had to keep changing them because the opponents would figure out how they worked. The other critical point was that in essence, the first one that's digital is the Mark X. The Mark X, effectively you're asked, "What is your password?" and you send a transmission that's your password. Just as with typed passwords, it's vulnerable to shoulder surfing.

So the Air Force came up with the notion circa 1950 of building a system in which the basic mechanism is the fire-control radar challenges you and it says in effect, "Show me that you can encrypt this correctly so that I'll know you're a friend of mine." The challenge is transient, it's a time or something like that, and the receiving system encrypts it and sends it back, and the challenger encrypts what returned and compares it with… I'm sorry, takes what returns and encrypts the original and compares the two. A very cute thing, they do it… Now you just have something displayed, a "Yes" or a "No." What they had was they'd translate it into the offsets of a dot on a screen. You either had one dot… The dots merged if it was giving you the right answers, and otherwise you got a cluster of dots.

You can't do that sort of encryption with a sequence generator. You really need to encrypt blocks of text. There was some history of that. There's a German system in World War I called ADFGVX that encrypts a block of about six characters, and there were systems… I can't remember … in the 19th century

there were a couple of things that do two characters or maybe three at a time. But you really can't… There are a whole lot of things in cryptography that only flower in the 20th century or the late 20th century because you can't really do them by hand. I mean these sequence generators operate on principles that were known in 1500 – you have table lookups and you have arithmetic – but you have to do much more of it than you can do and get correct by hand. It's not only a lot of circumstances you couldn't do it fast enough, you couldn't do it without errors. I mean try to do a six-fold Vigenère, which is what the not-very-good Enigma system is, and you're just never going to get any traffic through. Just too many errors occur.

In any event, Horst Feistel was head of a group at the Air Force Cambridge Research Center. It discovered this problem. There are actually two problems going on. The other one has to do with SAGE, the Semi-Automatic Ground Environment. They needed to secure communications over modems between remote radars in the computing centers. I have not totally gotten sorted out what the groups were, but Feistel discovered the IFF problem and thought that the system that had been designed had not been well enough vetted. His group began working on the IFF problem. Most of what ultimately emerges in DES was developed at that time. But at the time, transistors were a dollar a piece or something like that, so they couldn't do very… the things they did were very, very tiny by comparison with DES.

Then he worked… he's a little like me – a one trick pony. He worked on one problem through most of his career. That group at AFCRC made the mistake that cryptographers keep making. It solved its problem, but it used a system called Cadmus, which is in Mark XII IFF to this day. It's not the only one now, there's some others, but that… So '58, he had to move off to Lincoln Lab and Oliver Selfridge hired him. And this very interesting paper I can't find by Roland Silver on using computers to solve some simple puzzle sort of cryptography, but it's fascinating, the distribution list includes Horst Feistel and Marvin Minsky. That was a much more interesting group than you would have expected.

Then he went on to MITRE. He overlapped with me at MITRE, but I didn't know him at the time. He wasn't allowed to do cryptographic work there either. Then he got hired by IBM. IBM was not directly dependent on Air Force money, just at a level of selling them products, and he got to work on the contract to provide ATM security for Lloyds Bank London. That ultimately leads to DES.

It's a system that basically combines the simplest of such systems, combined transpositions… They unfortunately call them "permutations," but in cryptography a "transposition" is a permutation of the bits of characters or bits of the message as opposed to of the underlying alphabet. This consists of a proper mixture of permutations of the alphabet with permutations of the bits iterated a number of times. That's the type of system that DES is.

Williams: Why was it so important?

Diffie: Well, because it was the first US system, maybe in some sense the first system in the world of any quality at least, ever adopted by a government as a public cryptographic standard. US cryptographic standards, there are… I suppose NSA had by this point pretty much come across along into the notion of having standard cryptosystems. In the '50s, they designed a new system for each radio, because all of it was evolving so fast. None of the radios would talk to each other anyway, so there's no particular reason to have interoperability at the level of the crypto when you didn't have interoperability at the level of the modulation and the spreading or any of the rest of these things.

In the late '60s, they adopt the first cryptosystem that's used across a number of pieces of equipment. So they'd had this notion, but the adoption of DES is the first time that that approach is taken so publicly. Alright, NIST will publish a standard. We will have a group that qualifies things as meeting the standard. They will be known to meet the standard because there's a set of published tests that are run on them and they have to encrypt all these things exactly correctly. And the tests are organized so the failure of the device is likely to make it fail the test – they exercise every lookup within the device, every data path within the device.

It was very important because basically what's now so evident, which is the digitization of business, was just beginning then. But it wasn't present at a retail level, but somewhere in those arguments, no later than around 1980, I was told that Fedwire, the check-clearing service, was processing… it's at least $50 billion a year, maybe it's $50 trillion. I mean the velocity of money is high. That means that relative to the total amount of money, the amount of money moving around is huge. I remember saying, somebody saying at a 1978 National Computer Conference, a banker saying the largest single EFT he knew about – the limit was later lowered – was $28 billion.

Now what this means… I found this when I thought about it very interesting, because NSA's attitude toward this was "This cryptosystem isn't good enough for us, but you children don't … what you're doing." I came across a Navy regulation about how much protection is given to different things. It starts – you can't argue with this at the top – SIOP-ESI, the nuclear end-of-the-world plans. Alright. Those are the most …. And it graduates down, Top Secret, Secret, Confidential. The next thing under that is, quote, "very large amounts of money." Well, when you're talking about a billion dollars, there's something wrong with this thinking, because NSA's budget is a few billion dollars. Somebody [1:10:00] walked into the director's office and say, "Hey, I want to buy a Top Secret document. I'm willing to pay a billion dollars for it." I mean the situation is set up so they can't have that conversation with the Director. The Director would be thinking, "Oh, is that document really worth a billion dollars to us? I mean think what we could do

with 20% more in our budget" or something. But nonetheless, that was the attitude. That changed, has changed since.

So cryptographic systems were obviously needed. There were some cryptographic banking standards. I remember somebody, head of the standards group saying it was almost as secure as a Captain Cody decoder ring, about the authentication system used in one of the older things. It's roughly speaking like taking a check digit by summing the digits you have.

That is one thing about it. It was designed… There are a lot of funny things about this. Let me just mention that if you look at physical security standards, if you look at a safe for holding Top Secret documents, the ones that were current at that time were resistant to 10 minutes' forced entry. Well, "forced entry" means that you have a 25-pound bag of tools. You don't have any torches, you don't have any explosives, you don't have what's called a magnetic drill press which clamps on, you have to hand hold it or something. As I say, you're limited to a bag of tools that's judged to be something you could get in and out without being noticed through the guards.

You go to an ordinary grocery store and it probably has what's called a TRTL-30 safe. That is, it will resist torches and tools for 30 minutes. Maybe it has a TXTL-30 or 60 safe, which means it will resist explosives as well. The point is people don't want the secret documents. Secret documents are drug in the market. They're hard to sell. People want the money. [laughs]

In any event, a long explanation I've been practicing for years about why we needed a cryptographic system.

The aforementioned Howard Rosenblum was a key member of the… Director of Research at NSA and Director of COMSEC at another time … was important getting this. It was seen as important to fulfill I guess the Privacy Act of 1974 and some other… There was a Computer Security Act and a Privacy Act, all of whom seemed to require technical methods for protecting information. One of those was a cryptographic system. I imagine NSA convinced itself it would be better off if there were one standardized than if everybody, independent groups were working on them.

Williams: Can you tell us about your work on DES?

Diffie: Well, I had known… Butler Lampson told me at a restaurant called Sinon where I ran into him in the fall of 1974 that they were going… I mentioned the IBM system. He said, "You know they're going to standardize that," which I hadn't known. When it appeared in February or March '75, it was very much an IBM-style system. When I anticipated, I'm saying, "How in the world can they do this? I mean they can't publish one they can't break" – that was my view of their view –

“On the other hand, if it can be broken, they risk a tremendous embarrassment if they publish one that can be broken and it’s broken.”

That’s what led me to think of the notion of a trap-door cryptosystem. Going along with my proof-of-correctness line of thought, I thought, “Now imagine that we have a space of cryptosystems with the property that only a tiny number of them, it’s very sparse, are good ones. Pick something out of this space at random, it’s not going to be any good. Furthermore, they have the property that their parameters… it’s evolved by parameters, and if you know those parameters, then you will know how to break it, but they’re no longer evident in the final system.” So I had that notion when I encountered it in the late winter of ’75.

But when I began studying it, the first thing that occurred to me was just it seemed to me, one, it had a 56-bit key that it was trying to disguise as a 64-bit key. Fifty-six bits just didn’t seem to me like enough. I outlined to Marty a proposal for something… Ultimately we decided it could be done much cheaper. I thought it was… I had a proposal for roughly a half-a-billion-dollar machine that could break it in one day or a few hours or something of that kind. That seemed to me sufficient, and it is from a COMSEC viewpoint, to say, “This is not any good. You can imagine building a machine that will search through it.”

Anyway, that’s what… Then Marty and I, we both worked on some actual cryptanalysis. I think it’s my fault that we missed linear cryptanalysis at that time. I was so convinced that NSA had that approach down and wouldn’t possibly have approved anything that would succumb to linear cryptanalysis that I resisted Marty’s proposal to work on linear approximations. It is true we might or might not have found them, because you have to do it just right. They did linear approximation work at IBM that they thought of as showing the systems were secure, correlations of small numbers of bits with small numbers of key bits or something. You have to do it. The notion of a round characteristic and so forth that came up a bit later are not obvious.

But in any event, we worked on DES. In particular there was a meeting over near Stanford at Paul Baran’s offices on Welch Road in January… I imagine January of ’76. There’s a recording of some of it available on the Internet. But the meeting was originally to discuss some arcane points of some IEEE publications. The DES subject was added to the contents of the meeting. All of a sudden it was transformed and we were graced with a visit from Washington by three people, of whom the most important was Arthur Levenson, who had been head of a group called P1, which was an interdisciplinary… It was described to me as a brain trust for offensive cryptanalysis at NSA. The organization has changed since then. But Arthur Levenson was a really, really senior NSA cryptanalyst. He brought with him Dennis Branstad and a man named Doug Hogan. Levenson and Hogan have died since then.

At any event, it was Levenson… I think it of as a "between the boroughs" argument. Levenson's a Brooklyn Jew, Marty Hellman's a Bronx Jew. They had a New York style of argument, and Levenson tried to wave off our arguments about… By then, we'd refined the argument of "build a machine" down from my notion of half a billion. Marty had conceived it could be built for around $20 million, $25 million. He was just arrogantly dismissive and he pissed Marty off. Marty was… [laughs] He was going to rub his nose in the feasibility of building this machine.

That's what led to the 1977 paper, which… The design is in retrospect, not… For one thing, it uses silicon on sapphire, which is something that never went anywhere. But it was evident by 1990 or so when John Gilmore funded the development of enough of the machine. It was a quarter-million dollars' worth of it at that point. You could have built more of it, but that was enough to solve the problem in a week or so, and it did that.

And it turned on somebody at Northern, one of my colleagues. His name is slipping my mind. He wrote a very nice form of the paper in the '80s that was effectively a specialized DMS-100 that didn't switch calls, it broke DES.

Williams: During the '70s, the National Security Agency, who you have referred to several times, considered matters cryptologic to be part of its own private preserve and would certainly have regarded your work as an intrusion. What inhibiting effect did this have on your interest in cryptology?

Diffie: Well, it didn't have much effect on mine. I've been told incidentally that if I thought "New Directions" produced a stir in the outside world, that was nothing compared to the stir it produced at NSA. That seems to me entirely plausible. But NSA had a number of sort of interlocking attitudes. One, it liked to claim it knew everything about cryptography and we knew nothing [1:20:00] and we should stop wasting our time, and, something possibly somewhat in contradiction, that we would endanger the country if we went around working on cryptography.

I was less directly exposed to that at that time. Marty talked directly to NSA people. Having a professorship at Stanford sort of put him in a more visible position. I talked to peripheral people. One I particularly remember is Jim Simons, now the best-paid mathematician in the world, who had been a consultant for IBM. Jim Simons owes his career to being fired by my friend Dick Leibler. In the late '60s, Simons said rather too publically, "I'm not doing any more secret work till the war is over." Loosely speaking, Dick Leibler said, "We don't do anything here but secret work. Get out."

Simons got out. He immediately became head of the math department at Stony Brook and formed a company. The first one he did predicted the prime interest rate. Then he went through … he backed venture for a while and now he's doing

– what's it called? – a hedge fund, market manipulation sort of stuff. I don't know any details.

But the other wonderful thing he did, one of the French Impressionists whose name I can never remember said, "Life being what it is, one dreams of revenge." Jim Simons had his revenge in grand style. The first person he hired away from Dick Leibler was Lenny Baum, whom Leibler thought might be the best cryptanalyst in the country and is responsible for what's hidden Markov variable analysis, which later infested a whole lot of fields. Then subsequently he hired somebody I haven't met that I'm told many people think is the smartest person they've ever met. His name, Nick Patterson. He's a British import, came from GCHQ, and went to Princeton.

At any event, I had only sort of secondhand knowledge of what these people thought. But yes, I think they had been… Basically they react the way anybody else who thought he had a monopoly does. Cryptography is their product line and all of a sudden their monopoly on their product line was challenged. If you look at their actions, they can be seen in that light.

The next one is in the '80s, they formed… they expan-… they had this plan… I'm trying… Industrial COMSEC Approval Program or some acronym like that. The basic phenomenon was to change the way COMSEC equipment was purchased. But incidental to that, it's a market expansion program. They intended to make two classes of equipment, what are called Type I for military use and Type II, which was, quote, "good enough for everything except classified information." There were a few pieces of equipment built, but they never… the whole program just didn't work very well. I don't think they ever solved the problem of exporting the Type II ones and it re-emerged as the key escrow program in the early '90s, and that one never flew very far either.

But both of these, they're all attempts by NSA to recapture this market that was getting away from it. Probably somebody should do a business school thesis, business case study in losing your market.

Williams: Did NSA ever make any attempt to get you behind the fence?

Diffie: Well, not seriously. That is to say on at least one occasion, somebody has said to me, "Hey, why don't you come down and do some real work?" But nobody… I think they've probably made propositions to other people more forcefully, more energetically than that. They didn't. Exactly why I don't know.

I'm not convinced I would have been that good for NSA, because there are much better mathematicians in the world and they've hired quite a lot of those. I'm imaginative, but whether… I always think it's probably good I didn't work for them. Suppose I'd joined them instead of going off and working on my own. I would probably just have been buried in that vast… There's a huge culture. I

would just have probably been absorbed into their lines of thought. In any event, whatever reasons, no, they've never seriously courted me.

Williams: The citation for your A.M. Turing Award is – and I'm quoting – "For inventing and promulgating both asymmetric public-key cryptography, including its application to digital signatures, and a practical cryptographic key exchange method." What is public-key cryptography and why is it so important?

Diffie: The paradigm of conventional cryptography is that you and I share a small amount of secret information we call the "key," a few hundred bits, and we can use that to protect gigabits of information. The problem with that is that we have in some sense already to know each other fairly well. We don't literally have to know each other, but we have to be within a structure that can… The keys are secret, and so something has to convey the same key to you and to me securely before we can use cryptography in our communications.

This works fine in DoD. DoD is a very large organization. It has more than a million employees. But it has a very rigid structure and everybody knows what it is. Authority flows from the President to the Secretary of Defense to the Joint Chiefs of Staff to the four-star commands and so forth. So it has the luxury of appointing what's called the Executive Agent for Communication Security, which is NSA, and then everybody is in a position of trusting NSA to run the key management system, which starts with something called the Central Facility. The Central Facility manufactures keys, which these days are largely electronic, but they involve those rotors I discussed and pieces of paper tape and IBM cards, and they had giant printing presses. They claimed they had the largest printing press in the world at that… This would be going into the '70s. They're saying they published more key than *The New York Times* published newspaper or something like that.

Now the trouble is that does not suit the world we were going into. If you look, that would not be at all good for the Internet, in which what is so wonderful is that people with no connection to each other – they don't know each other and they don't have any national connection, any business connection, etc. – they find each other and begin doing business.

So public-key cryptography is a scheme whereby – and this is not how I originally conceived of it, but this viewpoint is very important now and is what underlies the Diffie–Hellman–Merkle–Williams key exchange mechanism – two of us can negotiate in public and everything we say is heard by all of the observers, but at the end of our negotiation we come up with a secret that both of us know and none of the observers know. That underlies the big Internet security mechanism, which is called Transport Layer Security, in which I as a client call a server somewhere and we enter into a negotiation that ends up with a cryptographic key that's then used to protect all of our communications between ourselves.

Without public-key cryptography, you couldn't do this without building a whole lot of centralized infrastructure that would amount to the world having to trust something – you know, the UN, the Swiss banks, American Express. There are candidates for doing it. But it makes it much easier… Security is the science of minimizing trust. Public-key cryptography made a great contribution to minimizing the trust among people.

Williams: Was there a eureka moment when you sort of realized that you could actually do this?

Diffie: Yeah, there was. It's a particularly satisfying moment because there were two problems. In 1955… Sorry, 1965. In 1965, somebody I'd been a freshman with at MIT named Bill Mann – still a friend of mine, long-term influence on my life – he told me mistakenly that NSA encrypted the telephones within its own building. Now that's barely true now and it was even more barely true then. What they actually do is they have two separate sets of phone wires, [1:30:00] one of them at least run in shielded conduits. If you walk down a corridor, you see secure telephone rooms and I don't know what the others are called. So they have an internal… everybody's desk has an internal phone. People who deal with the outside world also have an external phone. So you just plain have separate phones, have separate sets of wires.

But I believed him of course. I had no reason not to believe them. He was working with NSA at the time. I thought, "I can understand how you could do it" – I didn't know any details or hadn't thought about the problems of vocoding or any of that, but I can understand you could encrypt telephone calls. But I had a very countercultural viewpoint and it was very anti–central authority. I didn't see what good it would do you, because I could understand how you could have some central authority delivering keys to people, but I didn't see how… I thought a phone call is secure if only the two people on the ends of the line can understand it and nobody else can possibly understand it. So I put away in my mind for years, and I thought of this, "What could you do that would give you security for such a phone system?"

Then in 1970, I was just arrived at Stanford and John McCarthy went off and gave a talk in Bordeaux about what he called something like "buying and selling through home terminals." I do not know whether that contributes to the evolution of Minitel, but what he was talking is very much like the French Minitel system. Because John McCarthy not quite I think to the end of his life, for a long time called distributed computing "the Xerox heresy." He thought it was a mistake to think people would do a significant amount of computing on their own machines. In some sense it may be that his point of view has returned. That is to say the big data computers are doing vastly more computing than you do in your desktop computer, which has more computing power than any computing center of the '70s.

But at any event, he thought of it in very much Minitel terms. He came back and he told us about that, talked about that. I began thinking about electronic offices. What I didn't see was what you would do for signatures in an electronic office, because written signatures depend on their uniqueness for their virtues and digital documents are always perfectly reproducible. I didn't see how you could have a digital signature and I didn't see how you could run an office without signed memos, signed checks, signed directives, etc. So I began thinking about that.

In the spring of 1975, I wasn't having a lot of success. What I really wanted to do in cryptography was I wanted to solve the old problem so to speak. Every mathematician who's ever started working in cryptography has thought, [laughs] "We got to clean this subject up. We need clear proofs about why systems are secure. Let's study the way systems are designed and we'll…" I wasn't making any progress with that, which isn't surprising. Nobody's made any progress with it.

I also had a list of what I called "problems for an ambitious theory of cryptography," because… Therein among these was trying to combine two things. One of them was… I mentioned Horst Feistel and identification friend or foe. IFF systems protect you against what in an office context would be called "shoulder surfing." Somebody's looking over your shoulder and sees you type the password. As the password stays the same days, weeks, or months, after you go away that person could sit down at your console or some other console in the same system and type your password and get in and pretend to be you. In the IFF context, bombers in particular are probably coming in and if they are just giving passwords… The reason you do cryptographic IFF is so that the intercept radars, intercept radios can't hear what the aircraft said to the fire-control radars and know how to do it again. That's one problem.

Unix – this is traced back to earlier systems, it was popular in Unix at that time, probably goes back to Maurice Wilkes at Cambridge – had a system called one-way ciphers. They got a great deal of mileage, it unfortunately dissolved later, but the password table was not only not secret, it was outright public. The password table listed everybody who had access to the system together with a, quote, "one-way enciphered version" of the password. That protected you, not just protected you against compromise of the password table. It actually made administering the system much easier, because if you have somebody who has an account on this system and you say, "Oh, we want to open an account for that person on the other system," just move that entry of the password table over.

Now what do you do to combine the two things? I was trying to think… And I envisioned some protocol that would combine the two things. Eventually I realized that this… I came to what are now called "digital signatures." I thought the mere fact that you could recognize a correct solution to a problem, which the

password-accepter could do, doesn't mean you could solve the problem. So you could sign something because a problem would come to you and you would solve it and send them the answer, and they could recognize the answer as having solved their problem even though they couldn't solve the problem themselves.

I thought about various things we might do to do that. About a week later… And I've sadly, at least so far, lost the exact date, because I was keeping my log of my work on the AI Lab computer and for the first time I realized I'd done something that was really important and I knew the log on the computer wasn't secure, so I didn't enter it in that log. It may be on a piece of paper somewhere. I realized you could turn that around and use that system to establish communication when people had no prior contact with each other. That's not the same viewpoint as is involved in Diffie–Hellman–Merkle–Williams key negotiation, but it does achieve the same thing, which is that you have a public key that you can list in a telephone book, and I look it up and I want to send you a message and I use that key, and I can encrypt using that key but I can't decrypt the results, only you can decrypt the results.

Williams: How'd you react to the announcement in 1977 that Rivest, Shamir, and Adleman had produced a practical asymmetric cryptosystem?

Diffie: Well, some combination of I was thrilled and at some point I kicked myself. Let me deal with the latter point first. I had talked to John McCarthy about difficulties in proving things and he mentioned something called Wilson's theorem. Wilson's theorem is in one sense totally trivial. It says the product of all non-zero elements of a finite field is 1, and since you throw out the zero, all the rest of them have inverses. Perfectly clear you just organize the field that way and you get that product.

But for some reason I had a copy of… Mary and I were babysitting his younger daughter Sarah in the spring of '75 because he was off in Japan. This is a perfect example of the kinds of things… Mary made all of this possible, because it seems very unlikely, much as he trusted me, that he would have thought it looked alright for a single man to be babysitting his 13-year-old daughter. The fact that we were a couple made it perfectly reasonable.

In any event, we had his house and his home office and his remote… he had a remote connection to the AI Lab computer at that time, which now is ubiquitous but then was a rarity. I was moping around the house the way I do and I had thought about Wilson's theorem and I picked a copy of Knuth off of the shelf and looked it up. It was in there. I thought to myself, "There's some other theorem in number theory. That 1 on the right side of the equation is just what we need and there's some other theorem in number theory." But all I could think of was Fermat's little theorem. I didn't remember Euler's theorem. Then instead of going

and getting one of his number theory books and looking this up, I have a wandering mind. It wandered off to something else.

That's as close as I got to discovering RSA. But when I learned about it, which was from a letter that R, S, and A had sent to the guy at *Scientific American*, Martin Gardner, I realized this solved the problem. I didn't know in what detail. I went and told Marty that, and he had thought it was just another example. [1:40:00] He didn't realize. I said, "No, this really is not obvious how you would break this thing," and he realized that that had solved the problem.

That seemed to me it was just wonderful. I didn't know at that time, didn't understand that interestingly RSA is a dead end. You can make it work in other kinds of arithmetic, but nobody has ever found any arithmetic in which it works better. It will work in matrix arithmetic and things like that, but if you do that, it just costs you more to get the same amount of security.

That's one of the things none of us imagined. I imagined frankly, now it seems to be so utterly clearly mistakenly, that we were going to develop systems that were much, much more efficient and you wouldn't bother having asymmetric systems because you'd have these wonderful… wouldn't bother having symmetric systems. You'd have these asymmetric systems.

So I was very much a fan of RSA. I liked RSA better than our own scheme for years. It seemed to me much, much cleaner. In retrospect, it seems clear that particularly after ElGamal produced his signature scheme, I would have been much better off if Marty and I had understood we should push… we could have created the same company … that filled the same niche that RSA did, and been a lot better off for it. It's just fascinating that RSA has held on for so long. It's recently been reasserted by NSA as some … pick larger moduli for interim quantum resistance rather than elliptic-curve systems that do the Diffie–Hellman sort of thing. So I liked it.

Williams: Your 1976 seminal paper, co-authored by Marty, entitled "New Directions in Cryptography" established the subject of public-key cryptography. Could you tell us about this work, what inspired it, and what the individual contributions were?

Diffie: Well, my memory of this differs from Marty's. My recollection is what happened when I came to this insight that it could be done was… I was making a living as a househusband at this point, so I was taking care of that house and I had Mary and I had Sarah. I cooked dinner and Mary came home. Mary was working for British Petroleum in San Francisco. She came home and her memory of this… I do not remember this. I'll tell you what she says, and she's on tape saying it, is that I invited her in and said, "Sit down. I think I've made a really important discovery."

At any event, I certainly did tell her about it when she came home and we had dinner. Then after dinner, I walked downhill… John McCarthy's house is pretty much right above Marty Hellman's house. I walked downhill to see Marty and it took me an hour to persuade him that this could be done. It's interesting to me now that we know for sure it can be done, I can persuade people in a minute or something, but it took an hour to persuade him it could be done. My recollection is that when he realized it could be done, he said, "I have an invitation from Jim Massey to write a paper on cryptography for a special issue of the *Transactions on Information Theory*. Would you like to join me in it?" He doesn't remember it that way. He thinks he can't imagine he wouldn't have invited me to join that paper already. But in any event, that's how it seemed to me.

We then worked… that would be roughly speaking more than a year – and the sloth is likely mine – to produce "New Directions." What does it contain? I had been… The whole point… Oddly I mean, I probably didn't yet see public-key as the most important thing I would ever do in cryptography. I had spent the first two years while I was travelling around largely thinking about what the requirements for a cryptographic system were. It took me quite a while to come up with this notion of a chosen-plaintext attack and I was just generally thinking about framework.

I guess I don't any longer remember clearly what's in "New Directions" and what's in "Privacy and Authentication," but a whole lot of this issue sort of about modes of operation and the general ways you can build a cryptosystem I think I brought to the table. Anything that's information-theoretic Marty contributed. The general framework of public-key cryptography was mine, but the solution to the key negotiation was 90% his. We follow… And I don't know if cryptography has anything to do with this. I think we both agree with the practice followed at NSA, which is when a discovery is made, everyone in the room is listed as contributing to it. This was something we had been talking about, one-way functions and things you could do and so forth, for a year, and it's actually Marty who had the insight, but we both take credit for the discovery as being a product of joint work.

Williams: In fact, this investigation led among other things to the eventual establishment of the Diffie–Hellman key exchange protocol. It's one of the most widely used encryption techniques. It has applications throughout the Internet to secure online transactions. I think you've more or less discussed this, but for the purpose of people who may not be expert, what is a public-key distribution system?

Diffie: One of the issues here is that because of the fact… there are two ways of seeing the problem. RSA solves the problem in the way I saw it and Diffie–Hellman solves the problem in the way Merkle saw it. If you just see it as a matter of negotiating… I mean if you have RSA, other than for efficiency, you don't need any other cryptosystem. Anything I want to send to you, I encrypt in your public key. Anything you want to send to me, you encrypt in my public key.

As I said, that seemed much cleaner to me. I didn't see why you didn't… Now I look forward to getting rid of symmetric systems. We would just have… We were going to discover high-efficiency public-key systems. Of course in retrospect, why in the world could you think that adding a really difficult additional condition was going to allow you to have systems just as efficient as you could have without it? But I thought that way for years.

Merkle's approach to this is one of "We're going to negotiate a symmetric key by whatever means we do it, and then after that we will use the symmetric key for our communications." That has come to be the case, but you can use a different case. What you call in Diffie–Hellman or Diffie–Hellman–Merkle or Diffie–Hellman–Merkle–Williams, Williams being somebody at GCHQ who also worked on this problem — there there's a public element and a secret element, but the public element is not a key. You don't encrypt anything in that pub-… It's not like an RSA key. So there is a different… What happens is each person has one public thing and one secret thing, and they combine them to make a common value. That's done with a property of exponentiation or with a property of elliptic-curve groups in the more recent systems.

This is in some sense something, if the rules didn't prohibit it, you could do it in bridge. That is to say, what's going on in bridge is you and I are north and south, and each of us knows one hand. In talking to each other, we communicate more efficiently than we are communicating to east and west, who know neither of our hands. So in some sense, what we're talking about is perfectly efficient bridge bidding. There's a paper by one of the NSA guys called "A Cryptographic Approach to Bridge Bidding" or something like that. The rules don't actually allow it, but this is getting the effect. As I said, you negotiate in public and even though all the negotiation is in public, none of the observers learn anything about the secret that was developed by the principals.

At the time, because I'd had the notion… I didn't see that as being as good. We didn't call that "public-key" at the time. We called it "public-key distribution." That was a disaster in the sense that had we called that "public-key" also, we might well have gotten more action out of our patent, which basically the Diffie–Hellman – I imagine Merkle's name is on it – [1:50:00] patent claims in effect all public-key systems. But because we didn't set the names up right, as one of the lawyers for Stanford explained to us, we didn't get to milk everybody for as much as we might have gotten out of them.

Williams: What was NSA's response to all of this?

Diffie: Well, in the summer of '75, it'd be the third summer that we went back to the east coast, but this time we did it, I drove and Mary flew. One of the things I did along… I had liked Arthur Levenson. I didn't have the big fight with him. Marty did. And I guess in between I had learned… I talked to a man who died last year named Stockton Gaines who was at RAND who had been at the

laboratory in Princeton. In effect, it was one of the few times I got a real double take out of somebody. I mentioned that Arthur Levenson had come to this meeting and he was silent for it must have been 20 seconds. So I understood that Arthur Levenson was important out of a scale ...

So when I went back to Washington, I went to see Arthur Levenson, who was very gracious. I told him about how this worked. He said… you know … "Hmm, that's a nice idea. You might do something with it," or something of that kind. I probably also told Jim Simons about it. I had it in mind basically… I had a more … or a less hostile-to-NSA view. I don't know exactly friendly to them, but I hope I never said during the argument about DES that I thought they put a trap door in it. I didn't. Other people… I thought that was an indiscreet thing to say. I didn't know, but I wasn't inclined to accuse them of that. And same sort of thing. I told Levenson before this was published to give him a chance to have him come back to me and say, "We really do wish you wouldn't publish that."

They didn't say so and it was published. It had been in fact aired at a number of places. Marty went and talked in Ronneby, Sweden and the following… Well, that's actually months later than this. I'm obviously confusing '75… No, it's the summer of '76, because we already had the key negotiation scheme. Marty had talked about the ideas but probably not about key negotiation yet.

In any event, I don't know what they did. Things emerged from them years later. I see the Industrial COMSEC Endorsement Program, which as I say is a market expansion thing. It's not so much aimed at us as aimed at trying to… Well, the part of it that was Type II equipment was aimed at trying to sell into and maybe capture the market that we were imagining we were developing stuff for.

Williams: You mentioned a patent. What aspect of these ideas did you patent?

Diffie: Well, I find patents the most unreadable technical writing and I'm not sure I've even read my own. But it's a description of the key negotiation scheme that attempts to make it sound like a piece of hardware, because at that time, algorithms were not considered patentable, and it has all this jargon about "said-first signal" and "said-second signal," and it's a matter of our exchanging signals until we come up with a common secret thing.

Williams: These results became the cornerstone of Internet commerce. Did you gain anything from it beyond the gratification of your own intellectual curiosity?

Diffie: Oh, come on. Yeah. Making a living off of it for 40 years. My revenue on the patent is negligible. It made around $10,000, and there was some year that it sold reasonably well and I got a few thousand dollars off it. Aside from the Turing Award, the real income was making millions of dollars off of working for Sun,

which is a job that I got because of public-key cryptography and of working with, not exactly for, RSA. So yes, I have been awarded. Of course one can imagine ways that I could have made more off of it. R, S, and A did better off the deal. But no, I think I did very well on this.

Williams: Of course the Diffie–Hellman protocol is broken once a quantum computer with enough qubits is constructed. What do you think of that possibility?

Diffie: Well, [chuckles] there's a wonderful scene in the book *Wiseguy* in which something is described as "an affair among the Italians." Henry Hill, the subject of *Wiseguy*, his mother is Italian but his father has the name "Hill" because his father's Irish. Anyways, it was understood in Brooklyn mob circles that Italians were a form of life above everything else.

So the… Well, ask that question again. Let me try to get at it again.

Williams: I was just asking what you thought of the possibility of a quantum computer…

Diffie: Oh, okay. Fine. This is an affair among the physicists. I have no idea whether they're going to produce a quantum computer or not. As a matter of fact, what I thought at the time, that the NSA family of algorithms that are called Suite B – a set of public algorithms announced as trusted for all levels of classified information, Suite A being presumably a set of secret algorithms that they had before that. I had inferred, I knew that they'd been spending millions a year on quantum computing, on research into quantum computing, and I figured this meant that since they had moved to a system that had smaller keys and smaller numbers overall, and therefore ought to be more vulnerable I thought to quantum computing… Because what quantum computing does so well is find hidden cycle lengths. That breaks both RSA and Diffie–Hellman.

But they moved to elliptic-curve Diffie–Hellman. I figured, well, alright. They have a timescale of 60 years or so. I figured it takes five years to design a cryptosystem and another 5 to 10 years to get it fully into service, and it will be in service for 20, 30, 40 years, and on the last day it's in service, somebody will encrypt a Top Secret message which has a statute lifetime of 30 years. Altogether it adds up to 50 years to a century, depending. I thought, "Okay. This is evidence" – this was 1904 or something, 2004 – "they must have thought, 'Oh. Well, quantum computing isn't going to be any great threat in the near future.'" They seem to have changed their minds and everybody else seems to have changed their minds. I don't know. They have this Canadian computer whose name I forget. What's it called?

Williams: D-Wave?

Diffie: D-Wave, which claims to be a quantum computer but it doesn't seem to break RSA. I don't understand exactly what it does do except it costs a lot of money. I think NSA has bought several of them. Good for the Canadian balance of payments. I frankly don't… [I've talked to people and I get mixed reports on whether in fact it is easier to break an approximately 500-bit elliptic-curve system or a 2,000-bit RSA system. It has something to do with how many qubits you would need to handle the set-up of something or other.

What's Tanja's last name? Daniel Bernstein and Tanja…?

Williams: Lange, Lange.

Diffie: Lange. Okay. I asked Tanja Lange and Daniel Bernstein about it. They believe that the number of qubits needed to do the opposite to what I thought, the number of qubits to do this is much larger… it's 10 times as much or something as the same-size RSA number or something, which proves still to be several times the number needed for a somewhat larger RSA number. But the woman who was in charge of it at Microsoft Research thought that no, that was mistaken.

What NSA did, they published this very impolitic announcement two years ago that said after… They had people who went around the world, or around NATO in particular, twisting arms to get people to adopt Suite B, and [2:00:00] countries had to change rules that said secret traffic can never be encrypted with a public system to accommodate this, and then they publish announcement that says, "Well, if you implemented Suite B, thank you very much. But if you didn't, maybe you shouldn't bother because we're going to come out with quantum-resistant algorithms."

Now nobody I know believes they know operationally sound quantum-resistant algorithms yet. But they don't tell us everything they know. Maybe they do. Two years of all the people's arms were twisted, being pissed off have gone by, and all they did was to say, "Well, why don't you jack up your RSA? Go to RSA 3,000 bits instead of 2,000 bits." And if – this is a critical operational question – if you can solve the quantum computing problem by using RSA… For example, it can be run feasibly at 16,000 bits. Cylink had a set of chips that did that. People commonly went into 4,000 bits. The announcement allows 4,000, but 3,000's the minimum. If that will solve the problem, it's presumably much easier than all of this work on "Shall we resurrect knapsacks? Shall we use McEliece-type error-correcting codes? Shall we use lattice-basis reduction systems?" I do not know the answer.

Williams: Are you happy about what happened to public-key cryptography subsequent to your involvement?

Diffie: Of course I am. My response to the Turing Award was, and I said this in my acceptance speech, I feel utterly humbled to think the same award was won

by people like Les Lamport and Don Knuth whose work extends over such a broad swathe of things. I'm in another sense particularly embarrassed relative to Barbara Liskov, who won it for what kind of work I would liked to have done. I didn't und-… work on developing programming technique, that seems to me really the heart of computer science. This seemed to me kind of peripheral. I think there are respects in which I did very valuable work, but it's not in the usual sense of the sitting down at one's desk solving some mathematical problem or something.

But on the other hand, I look around the world, the impact, very few things have as much visible impact on the world as this does. It is possible there are things… of course having invented compilers or schedulers or something like that, parsers has even more impact, but less visibly because the individual algorithms didn't take hold in the world and become so well known.

Williams: What was your reaction to the news that the Diffie–Hellman protocol and public-key cryptography had been discovered a few years earlier than your work by researchers at GCHQ in the UK?

Diffie: Well, so I was first told that by Howard Rosenblum. Bobby Inman didn't quite tell the truth to a congressional committee. He said to them, "We discovered all of this at NSA 10 years ago." That date was like '77 he said that or something like that. They didn't discover it. It wasn't 10 years ago. It may be as much as eight. They didn't discover it. The guys at GCHQ did. I wanted to know who did it, and Howard Rosenblum told me, gave me the names of James Ellis, Clifford Cocks, Malcolm Williamson.

I set out to try to contact them. My style, one of the things I think is rare, close to unique about me as a researcher is that I employed the techniques of historian, investigative reporter, and researcher. So a pure reporter wanting to know how the NSA key management system worked would go around trying to find people who'd tell him. I did that, whereas some researchers were very leery of… they didn't want to talk to, they didn't want to be polluted by somebody saying they'd learned it from… I didn't care about that. But I also sat down and worked on the problem.

At any event, once I learned about the British work, I took a trip to Britain in the fall… Well, I was in Paris in the fall of '72 and took a side trip to go visit Racal COMSEC on business in Britain. I went to look up James Ellis, who is, if you look at people … James Ellis came up with the notion of public-key cryptography. He didn't come up with signatures but he came up with key exchange in the RSA style the way I did. Clifford Cocks came up with a system that is essentially the same as RSA. And Malcolm Williamson came up with a system that is essentially the same as Diffie–Hellman. Just incidentally, his internal secret paper on that subject is in August of 1976, two months after I presented at the National Computer Conference.

I observed this as… I see a principle of linearity in research. There are six Yanks and we did everything we did in about three years, and there were three Brits and they did it all about six years. So it just depends on how many bodies you can afford to work on this stuff.

James Ellis was the only one they were willing to let talk me. James Ellis was very senior. He was late in his career. So I went to see Ellis, and this is recounted in Steven Levy's book *Crypto*. We went and drank cider and ate skin potatoes at a pub down in Cheltenham. Unfortunately, maybe my travel reports know what it was, but I don't remember for sure. Then we went out for a long walk. There's a wonderful line in a book called *Northwest of Earth*, which is a piece of science fiction written in the '30s, that liquor went to Northwest's heels, not to his head, or he would never have lived as long in his profession, which was loosely speaking a smuggler. I realize I'm exactly the opposite. I mean after this walk, my feet were as solid as a rock and my head was just absolutely in the clouds. I had drunk two pints of Taunton Blackthorn, two British pints of Taunton Blackthorn cider and it took a two-hour walk or something before I was in any condition to drive anywhere.

But then after that, I tried… It was a long… I now get along very well with Clifford Cocks, but for years he was very… The first time I saw him, I read his badge at what's called the Cirencester Sciences, and I said, "Oh, hi. I'm Whit Diffie. I'm honored to meet you," or something like that. And next I looked around, he'd manage to vanish. He resisted talking to me. I tried to talk to him — he was at the 1985 Workshop on Information Theory in Brighton and he refused, didn't want to sit with… I think that's right. I think he didn't want to sit with me at dinner.

And I ran into Williamson. Unfortunately I never realized… Because of my own wife's illness, I didn't get to go dig Williamson up. He was down in La Jolla. There were things I wanted to know about Williamson's work that I never got to find out. He died say a year ago. But I saw him twice and I did have dinner with him once in Las Vegas at the Finite Fields conference.

But for years they were very standoffish about this. They waited until after Ellis's death before they published those papers. I've never… I have no solid understanding of why they got to these problems. Dick Walton, who was Ellis's boss, claims that Ellis was working on key management issues, that there was a problem with manufacturing and distributing enough key, and that he put him to work on this not imagining he was going to get any such imaginative a proposal, and that ultimately that was solved by key updating and more careful examination of the re-keying periods and things like that. But there's nothing in Ellis's paper at that time that suggests that he was working on key management. He published a later hist*[orical]* paper, but that paper's '87, saying, [2:10:00] "We were thinking about key management in very large systems at the time." But

aside from Walton's confirmation, no papers have been released that show any sign of that.

My diagnos-… And I spoke to Shaun Wylie, who had been the chief mathematician, and he's widely quoted as saying, "I wish I could see some reason this wouldn't work, but I don't," and it was disquieting to them.

So my sense is that until we worked on it, aside from having contributed signature, which they acknowledged they didn't understand, they didn't see it as something useful until after we did. Now there's somebody, and I can't think of his name, who gave a talk at the Computer History Museum claiming that they did know it was useful and he implemented a prototype in the summer of '74 at NSA. But no substantiation of that claim has ever come along and nobody else I've asked knows anything about that.

Williams: Alright. You mentioned during the course of this interview Steven Levy's book, which was written for popular consumption, called *Crypto*. What did you think of this book?

Diffie: Well, I no longer remember exactly what I thought. I was very pleased with Levy for sending us a draft to correct and making changes that detracted from the lilt of the language in order to be accurate.

Williams: You were Marty Hellman's PhD student at Stanford, but you did not complete your degree. You had a great dissertation topic, so what happened?

Diffie: Well, it requires a lot of courses in electrical engineering and I don't do very well with courses.

Williams: So you just decided to abandon it?

Diffie: I don't know. I don't remember exac-… My wife's Egyptian professor's husband got a job at Bell-Northern Research and he hired me.

Williams: Alright. Can you tell us something about your work at Northern Telecom?

Diffie: Sure. Let me think. I dragged on for about a dozen years. The most important thing I did was the high-level design of the something-or-other security overlay. What's the product's name? Packet network security overlay, which actually became a product and it started out, I did the key management design for the system.

Williams: Why did you leave?

Diffie: Sun made me an offer I couldn't refuse.

Williams: You worked a number of years at Sun. What were your duties there?

Diffie: Well, over a course of nearly 20 years, they varied. Mostly… I mean the best thing I did for them was probably talking to customers, but in the mid part of that period, most of what I did was fighting the political fight. When I was hired as a Distinguished Engineer, I was told it was the job of a Distinguished Engineer to do whatever within the scope of the company's business considered most important. I took that more seriously probably than most people took it. In about '93 with the advent of key escrow, it became clear to me that the political threats to network security were more significant or at least as significant as any technical threats, so I spent the next several years working on that.

Williams: Now I recall seeing you on television in the early '90s. You were sitting at a table with a number of individuals, one of whom was President Clinton. Can you tell us what that was about?

Diffie: Yeah, it was not the early '90s. 2000.

Williams: Was it 2000?

Diffie: Yeah. That meeting was about… There was some big denial-of-service attack and I don't remember… It goes something like that Clinton or Clinton's office invited… It was probably the one who was CEO of Google. What's his name?

Videographer: Schmidt?

Diffie: What?

Videographer: Schmidt?

Diffie: Schmidt. Yeah. Eric Schmidt. And Eric Schmidt, I'm guessing he … asked Greg Papadopoulos, and Greg Papadopoulos asked Jeff Bayer, and none of them wanted to go, and Jeff Bayer asked me and I said, "Sure, I'll go."

Williams: [laughs] Okay. You wrote a book with Susan Landau entitled *Privacy on the Line: The Politics of Wiretapping and Encryption*. What was this book about and what prompted you to write it?

Diffie: Well, I mean the title, the subtitle explains what it was about. I don't know what prompted us to write it. I mean I guess it seemed a valuable thing to contribute to our… Oh, I know. I'm sorry. Actually there is something on… We were on, it was an ACM committee with Barbara Simons… Barbara Simons put it

together, she wasn't on the committee, to produce a whitepaper on cryptography policy. Susan and I worked on it and each of us was convinced that we were the only two who cared what the truth was. The others were all there to support positions and … It appeared. Then we decided we would write a better book about the subject.

Williams: Was it a success?

Diffie: I don't remember exactly how well it sold. Yeah, I think. Everybody spoke well of it.

Williams: What do you believe were your most important contributions to the computer privacy debate?

Diffie: I should have knocked off at lunchtime.

Williams: [laughs]

Diffie: Well, I mean that question presumably doesn't mean to allow the answer, "Well, public-key cryptography," which largely created the debate because before public-key, all systems were escrowed. So I actually don't… I naturally believe everything I said was important.

Williams: [laughs] The Turing prize is the equivalent in computing to the Nobel Prize in other areas of human achievement. How did you react to the announcement that you'd won this very prestigious prize?

Diffie: I was utterly thrilled.

Williams: Have you made use of the award?

Diffie: You mean have I spent the money? Sure. No, the first thing I did when I got the message was that I studied the email header very carefully, [laughs] and I eventually persuaded myself it really had come from Berkeley. I went back downstairs. This was like two in the morning on a Monday morning. Mary, as usual, said, "What's new?" and I said, "Oh, there's good news. We won the Turing Award." It was really very satisfying, because it made the last year of our lives together much, much better, and made everything feel complete.

Williams: You've been a longtime supporter and attendee at the annual Crypto conferences since their beginning in 1981. What can you tell us about this meeting and why have you been such a faithful attendee?

Diffie: Well, it's not far away. I'm a conference lizard. I actually finally did miss one, although not on paper because I registered for it, but I couldn't get there last

year. I registered. I was on paper a member of the conference, but my body didn't get down there. That was the first time.

The wonderful thing about that conference, it changed a little bit in style and probably my involvement has changed, but somebody, Gersho or Kemmerer or somebody at the first one, had this notion you hold a cocktail party every night with food. That keeps 90% of the people within a hundred yards of the lounge, which means you get much more out of the conference. Because if we look at the Oakland conference, that in effect broke up every afternoon at five and people went off in groups of half a dozen to the good restaurants of Oakland and San Francisco. Whereas if you do something that keeps everybody there… I mean to start with, you hold it in an isolated place. I mean in some sense the worst, the least productive of certain kinds of our conferences was Paris '84 EUROCRYPT, because people quartered in hotels [2:20:01] around the Left Bank and the meeting is at the Sorbonne, so you have to walk past the cafés and the bookstores to get to your work. It's really very distracting. Whereas having a meeting at an isolated, reasonably isolated place like this and you're all staying in the dorms, you can stay up as late as you like… Dick Walton used to be up at three in the morning counting who was above the table and who was below the table and what nationalities they were.

So I just got in the habit of going there. I'm not as pleased with it now because while I was away, it turned into a multisession kind last year. I've never understood the attraction of that. I understand the attraction for people who want to get their way paid by having papers accepted. I understand. But for merely attending and learning what's on at the conference… Particularly we are discussing, probably will adopt some scheme like what DEFCON does where you get everything videotaped and you can watch conference meetings you didn't go to and things like that. But without that, you're in a constant position of having to choose between talks.

Williams: What are you doing now?

Diffie: Trying to get the house cleaned out so that I can sell it.

Williams: This is now the retrospective part of the interview.

Diffie: I see. What was un-retrospective about my childhood?

Williams: Well, that was your early childhood part. [laughs] Who during your career were your most influential role models?

Diffie: Hmm. I don't know. I never… I think the terms haven't accorded me well. You asked once about who my mentors were or something. I can tell you people I think very well of, Hilary Putnam for example, philosopher. It's funny. I can tell you lots of people I liked and think very well of – Minsky and McCarthy are both

examples. Are they role models? Do I remember having tried to imitate them in some way or other? I don't know the answer to that. I mean Dick Leibler, he's the reason I'm so well dressed. Or as well dressed as I am. I shouldn't claim to be a well-dressed man.

Williams: [laughs] Looking back, what were the turning points or major decisions that led you to where you are today?

Diffie: Well, obviously getting interested in cryptography. Going off to work in cryptography. Running off with Mary. That's utterly pivotal. I can't emphasize too much how big an influence she had in the way things went, not just the emotional support of her faith in me but the fact that everybody loved her, which opened all sorts of doors that might not have opened if I had been by myself. So that's… Getting involved with Marty.

Williams: What was a biggest regret in terms of the decisions you've made?

Diffie: I don't know. They all seem inevitable or something. I should have left BNR and joined Sun sooner, I mean.

Williams: What were your most important life lessons?

Diffie: Who wrote this material anyway? Important life lessons. Well, my friend Bill Gosper showed me that there could be a confidant configuration that keeps emitting gliders indefinitely, and so the diameter of the Game of Life grows unboundedly.

Williams: [laughs] What was your proudest moment?

Diffie: Oh, that's a good question. I don't know. I mean certainly… Do I think of things in terms of pride or just…? I mean I was really thrilled about the Turing Award. Now there probably are things I am proud of that won't have risen vastly to notice, but I don't know that I remember them.

Williams: Are there other interesting things that you worked on that you might want to talk about?

Diffie: Well, I'm really proud of my marriage. I mean I managed to play my role there adequately. [laughs] Thinking of that pleases me a great deal.

Are there things that I worked on? What have I worked on? I don't know. I don't remember offhand, I mean ones you don't know. I mean to work on the structure of programs I liked, but I never published anything significant about it and never got it into final form. I've worked on a whole bunch of yet-to-be-published, it may never be, sort of cryptographic stuff about design of systems. I've worked on… I

have yet to complete and there are still things that I think I can break that I haven't and hope to and may eventually publish something about that.

Williams: Well, we've covered a great deal of ground, but I wonder if there is something further that you'd like to mention.

Diffie: Oh, I see. It's a catch-all question. That one you probably should have said by email a week ago and maybe you would have gotten something like an answer out of it.

Williams: [laughs]

Diffie: I don't know. I'm sorry. This process has tired me out to the point where I don't… I may have some great thing I wanted to say, I'll remember it when I get home.

Williams: Well, thank you very much.

Diffie: Well, you're very welcome.

[2:26:39]
[end of recording]