An Interview with

RICHARD A. KEMMERER

OH 450

Conducted by Jeffrey R. Yost

on

30 April 2014

Computer Security History Project

University of California Santa Barbara, California

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

Richard A. Kemmerer Interview

30 April 2014

Oral History 450

Abstract

Computer security pioneer Richard Kemmerer discusses his graduate training (at UCLA), his early and long-term consulting work for System Development Corporation in computer security research and development, and his research and education of graduate students at University of California at Santa Barbara. Among the topics covered are his work on Secure Unix, electronic voting, intrusion detection, and other areas. He also relates perspectives on early conferences (VERkshop, IEEE Symposium on Security and Privacy, and others), the NCSC and TCSEC, and other topics.

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

Yost: My name is Jeffrey Yost from the Charles Babbage Institute at the University of Minnesota, and I'm here today, on the University of California Santa Barbara campus with Richard Kemmerer. This interview is part of CBI's NSF-sponsored project, "Building an Infrastructure for Computer Security History." It is April 30, 2014. I'd like to begin by just a few basic biographical questions; can you tell me where you were born and where you grew up?

Kemmerer: I was born in Allentown, Pennsylvania and I grew up there. I did my undergraduate at Pennsylvania State University and moved to California after that — a few things in between but basically after my undergraduate. Prior to that, I used to visit California because my older brother had moved there, and so for the three or four summers before I finished my undergraduate, I had spent my summers in California, so I was kind of "California experienced." [Laughs.]

Yost: Can you describe yourself as a student, in your pre-college days?

Kemmerer: Sure. I guess most of the people that you talk to that are professors or researchers probably took a different path than I did. In junior high they asked us as students to decide what we wanted to do — go to college or be a shop kid or a business kid — and in eighth grade you made that decision. I decided I wanted to be a shop kid. I wanted to fix radios and TVs. I thought that was the best thing in the world. And if you could do that, you know, you were cool. So I was a shop kid, over my mother's arguments. She wanted me to go to college but she let me do what I wanted, and so I was

a shop kid in ninth grade. When I went to high school, which started in tenth grade, fortunately for me they took some of us who had done well on some standard test that they gave us — the normal way that you were in shop, like I was in electronics shop, was you would do three weeks of shop and then three weeks of school. I think there were six of those things you did through the year. So for some of us who had tested well, they started a new program when I went to tenth grade where during our three weeks when we were in school instead of taking the normal shop classes, which was like lower level English, low sciences, if any, and all that. Instead of that, they gave us the equivalent of college-level math. But of course, we were like a year behind everybody else, so that the normal college prep kids did algebra in ninth grade; we didn't get algebra until tenth grade, and then geometry, and then whatever you did — calculus —so anyway, I was a shop kid. There was an advisor there, that in tenth grade already when I had to go see him, he said, what are you doing in shop? You should be in college prep. I told him I want to be in electronics shop, that's what I'm interested in. By the end of 11th grade I had finished all of the projects that you do in electronics shop, and so the next thing would be that my senior year, the three weeks that I was supposed to be in shop, I would actually be doing an internship or what's the other word for it? Something like an internship at an actual TV repair shop, and so I would've done that for three weeks.

Yost: Apprenticeship?

Kemmerer: Apprenticeship, that's the word I was looking for. So that was happening, that was going to happen; meanwhile, this guy is telling me I should change to college

prep so I changed to college prep my senior year. So my senior year I was actually thrown in with the college prep kids but I was still, you know, a year behind on the math; and I did well. At that time, we didn't have community colleges or junior colleges and to get into college you needed two years of a language. I took my first year of a language my senior year and I studied Spanish; so I wasn't qualified to go to college yet. The high school was really great. They let me come back the year after I graduated. So I graduated because I had everything I needed to graduate, and that summer then I took two different math classes that I needed — solid geometry and something else — to help me catch up. Then the year after that I came back, and I took another year of Spanish, I took another year of math, I took I think psychology, I took some kind of science — I think it was physics or something like that — and at the end of the year I was qualified to go to school, and then I started college after that. [Laughing] I'm the shop kid so it's fun when I go back to my high school reunions, especially the first one I went back to because I was just graduating college now five years later and they go oh, college? You're a shop kid. And then now that I'm a professor it's even more fun to go back because it's like hey, you're the shop kid. So, yes, that was my high school experience and I always liked to work on projects, work on cars, mechanical work with my hands, and do things like that. But that was high school.

Yost: What year did you start at Penn State?

Kemmerer: I actually started at Kutztown State, which had just changed their name from Kutztown Teacher's College to Kutztown State College; and I was a math major and

physics minor. Again, just like in eighth grade, they ask you what you want to do. I said, I don't know; I'm good in math and I like physics. To me, calculus and physics gave you so much power; those tools are so great for being able to figure out things. So I went to Kutztown State College, which was 18 miles away, it was between Allentown and Reading, and I would drive. I was a commuter, so I lived at home and I drove out there every day. Like I said, a math major with physics minor; but I always wanted; I mean, Penn State was a good school and I thought I couldn't afford to go there. Basically, I got some student loans and I got a very small scholarship from a state senator, so I was able to afford to go to Penn State and so I actually changed to go up there my junior year. I spent two years, from 1964 through 1966 at Penn State, which was wonderful. And it was probably good because it was good doing the two years at Kutztown State. It was kind of like going to a junior college so you get a little more maturity, you know what it's about and now you can leave home. And like I say, in between there, I had spent summers living with my brother out here, living in California, which also gave me some maturity. But I went to Penn State and I was a math major, physics minor, did well, had fun, loved it, and then all of a sudden it was graduation time.

Yost: While you were at Penn State did you have any exposure to computing?

Kemmerer: I did not. I don't think they had a computer science department then. There was a guy in my dorm that had Snoopy calendars that were printed out on a; so basically he had — I didn't know this until years later — he had a deck of punch cards which he ran through and printed Snoopy the dog with a calendar on it. I thought, oh man, this

guy's a — I wouldn't call him a computer scientist, because I didn't even know that term then, okay? The one thing I did do is we had a math class where we worked with the Friden calculators and did that kind of work. That's as close as I got to a computer. I had a friend who, at the county fair in Allentown, they had this supposed computer – a box with blinking lights and stuff, and it told you your fortune. He was inside that box looking at who was out there to hand them the appropriate fortune. I can't really call that using a computer.

Yost: So as you finished college, what were you thinking career-wise?

Kemmerer: Oh yes. Next thing I know, I was ready to graduate — and this is 1966 — and of course, I got my draft notice to go for my physical in Harrisburg. If you remember those times, we were in the Vietnam War then. We got a little bit deeper years later, but we had admitted to being in Vietnam at that time, and I wasn't keen on that and so I thought it would be good to go to grad school. I asked one of my professors, Donald Rung, because I had no idea what grad school was. I knew people go to grad school. So I had these lofty ideas. In fact, at Penn State, a guy that started a year after me came in and knew he wanted to be a doctor. So he finished, went to pre-med, he's a doctor now. I didn't have any idea what I wanted to do. My life has been that way, I never knew. Not that I didn't like what I was doing, but I didn't know what I wanted to do. So anyway, I asked Professor Rung what are good math schools to apply to? He told me, and I don't remember which ones I applied to, but one was Michigan State. I found out later that that's where he graduated from, so it was not a surprise that he recommended it. So I

went to Michigan State the fall of 1966 and I went there because I had a TAship so I could afford to go because it paid for everything. Michigan — moving to the Midwest after growing up on the East coast and having spent time in California, and particularly moving to — God, now I forget where.

Yost: East Lansing.

Kemmerer: East Lansing, yes. So moving to East Lansing. Had I gone to the University of Michigan it might have been different, but East Lansing was sort of "turn your watches back 50 years." No, really. Everything was; I really felt this is strange. So anyway, I was a math major, took math classes, I was doing fine, great football team. I was introduced to Big Ten football. I had 50-yard line seats to see Michigan State play Notre Dame to a 10-10 tie. Exciting. Penn State, you know, was a big football school, but that was pre-Paterno days so we weren't that good; if we had a winning season in football we were happy, okay? And Paterno was an assistant coach then. But to go to Michigan State and the Saturday lunch — I was living in the graduate dorm — at lunch, when they're playing the fight song as you're eating lunch, I mean, that's a whole different game.

Yost: I grew up in Lincoln, Nebraska, a small rather dull place, but the University of Nebraska had and still has a rich college football tradition. A rather pervasive football culture.

Kemmerer: That's kind of fun but I wasn't happy there. So one of the things that happened was they had a computer science department at Michigan State, and one day, I thought I ought to go find out what computer science is about because somehow it seemed like I would be interested in that. So I walked over to the computer science department, told them I was a math major, and I'd probably brought my grades with me or something. And so basically, they said if you want to come into computer science; I think they actually could've given me a TA, I'm not sure. But we laid out the coursework that I would take and basically I would spend a year — maybe more than a year but at least a year — taking the required undergraduate classes and then I could start taking classes that would count towards my master's degree, because I was only in a master's program, not a Ph.D. And I just said no, that's too long, I can't do that. But then in the middle of the quarter, or maybe it was the end of the quarter, there was a career fair where companies came in to interview people for jobs. I thought I'd just go over there. In fact, in the back of my mind was I'll talk to places from California because the summer prior to going to Michigan State was the first summer in about five years that I hadn't gone to California so I sort of missed my California fix. And so I thought okay, let me talk to some of these companies and maybe they'll fly me out for an interview and I'll visit my brothers, because I had two brothers living there by that time. And so I interviewed with some of them, and I also interviewed with some around the Philadelphia area, too, just more to find out what was going on. Anyway, I remember one of them was North American Aviation, and North American Aviation programmed Minuteman Missiles. They said they were very interested in me, and I said, do you know that I've never touched a computer? Touching wasn't such a big deal because you didn't touch

them back then, they were behind a glass wall. But I never wrote a program. I said I've never taken a computer science class. They said that's okay, we find that there aren't computer science people. You know, we weren't producing computer science people, or very few, and they found that people with math and physics are a good match to learn this.

Yost: Did they have a test, sort of like IBM? An aptitude test?

Kemmerer: It was interesting because one summer, I went to IBM and took one of their tests, but I didn't tell them it was only for summer employment. I probably sound like a devious guy. I did real well on it and they wanted to hire me; then I thought well, no, if I went to work for them and then three months later quit, you know, maybe later I'd want to go to work for them and that wouldn't be good, which was probably a smart move to do that. But I know the test you're talking about. No, I don't remember any test at all for North American Aviation. Basically, they were hurting so badly to get programmers and they took people that did well in math; and they went for physics, too, because that is what they were looking for; or if you were engineering — math and engineering — they probably would've gone after you. But anyway, they said they were really interested and they asked if I was interested, and I said yes, I guess, and I'd like to know more. They told me more about it and then there was a second call where they made me an offer. I had talked to other math students and I knew what — also from that job fair — I knew that getting a master's degree in math, you became an actuary or you went to work for NSA, basically. And I didn't have any idea what NSA did but the thing was, I knew what

they would pay you with a master's degree, okay? Anyway, this guy called me and I was so proud of myself because he said we're prepared to make you an offer. And he said the offer is we'll fly you out and move all your belongings, we'll give you; you know, all the nice amenities; treating me nice, which is always good when you're a college student and see someone's going to treat you like a real person. And then basically, he gave me the number, which I believe was around $770 a month, if I remember correctly, which was way over what people were making that were graduating with master's degrees. And he said, how does that sound? Here's where I was so proud of myself; I said, well, it keeps you in the ballpark. [Laughs.] Anyway, long story short, I accepted that position because the other thing they said, the other piece was we'll pay for you to go to school. You won't go full time, but you can take classes and we'll reimburse you for all the classes you take. I said, oh good, because I'd like to finish my degree. Anyway, long story short, I accepted; went home for Christmas; then flew out, put my car in the back of a moving van. I had my car moved out there because I didn't have much else to transport. And I started at North American Aviation in January 1967. And oh, the key thing there was you had a draft deferment. I programmed Minuteman Missiles for three years. I turned 26 in November of 1969 and they weren't drafting people after 26. My brother, at that time had started working at UCLA and he told me you should come up here because we hire consultants. You could probably get a consulting job. So I stopped by one day and talked to them and talked to this guy who was managing the programmers and data collection staff, and basically, they didn't need anybody. But he took my name and then a few months later he called me and said, would you like a job? And I said what's the job? And he said well, my job, because he found that he didn't like being a manager. And so

basically, that happened, and I had my three years in at North American and so I went to work at UCLA. At UCLA I was the manager of computer services at the Institute of Transportation and Traffic Engineering, and we did things like crash cars at 30 miles an hour into a barrier to see what the results were. We did diamond lane studies, because this was pre-diamond lane. We also had a helicopter in the sky and we'd have cameras from the helicopter take pictures of the freeway where we were, which was close to where there was an overpass, because we would also take pictures of the license plates as the cars went by and then we would see how people were weaving or what they were doing. Like I say, we also did some diamond lane studies, but that wasn't outside. But all of the programming to analyze that data, put it into a form where people could make conclusions, we had about I think when I started I had 12 programmers that worked for me, who did the scanning; other people set up the things. We would photograph those license plates. We would take the film that recorded those license plates down to Hollywood to one of these people that did the dailies from the movie industry, who'd process them. Again, you have to remember this was the 1970s, it's not like today where you go to Costco. So we'd process it right away. I'd put one of my guys on a plane. He would fly up to Sacramento to the DMV. We'd take those plate numbers so we could get addresses and names for them, and then we'd send out a mailer because we wanted to see what kind of trip? You know, were you working, shopping, what kind of trip were you on? How often do you travel on this road? And all that sort of information. And we would process that. At the same time we were taking the film from the helicopters and we were plotting each car and then had a program so that it would tell you, based on where it was, here's how that car went and things like that. We also had a driving

simulation lab, where we had a 1966 Chevrolet in a room, running on a dynamometer and so you felt like you were really driving. We had screens; at first, I think, the screens were 180 degrees though eventually we did a 360-degree vision, again in that time period. And it was all by engineering it ourselves, and, in fact, for the 360 degree movies, General Motors built a vehicle that we used and mounted cameras on so we could go drive like out in the desert and film these things. I think we had six cameras to give us a 360-degree view, and we would get people — we did alcohol studies — we'd get people drunk, see how they drove, and how they drove when they weren't drunk. We'd test people on amphetamines. We did things from drug companies. We did studies for the U.S. government on marijuana; you know, where you had the placebo and the real one. So we collected all this data. We'd collect data on how they were driving, obviously, but they were also doing other tasks. Like there were two lights up above here, when one went on you were supposed to pull a handle; how quick were their responses? We also sampled the galvanic skin response and all sorts of data. And then, we processed all that. We had a cart that collected it on an analog tape, and then we used to take it to the campus computer center and run it on their IBM 7094 to process it. But they sold their 7094 and bought an IBM System/360. We bought their 7094, and so we had our own 7094 that we ran on. And then they decided that we were an institute, a research institute, and so we were competing with them by having a 7094, so they wouldn't let us do that anymore. So now I had a courier that would drive up to Point Mugu, which had a 7094 to process their satellite data. And on their [Pt. Mugu's] down time they would let us run our programs— all we were doing was A to D conversion. You know, reading the analog values and getting them into digital and then we could do it ourselves. And so this, besides being

time consuming, was expensive and not very convenient. And so what we did was we had $30,000 and we bought a DEC PDP-8. I looked at a bunch of computers. Part of what I did as the manager of computer services was I — all the programmers worked for me, the scanning people, all that, plus all the contracts that went out, I had to put the bid on it of what the computer cost would be, or what hardware we'd need. Somehow I got $30,000 and I looked at a bunch of minicomputers, because minicomputers had just come out then. I decided to go with DEC, and I got a DEC PDP-8E, with the idea that with the PDP-8E, I got a tape drive, the operating system, and a model 33 teletype, and a paper tape reader. My idea was to use this and put it in the simulator room next to where the car was on the dynamometer, and then collect the data digitally. Right? You don't have to be a rocket scientist to see that makes sense. The problem was that we only had $30,000 so we couldn't buy a disk, so we had a tape drive, a nine-track tape drive. So we took the operating system we had and we formatted the tape drive into blocks because when we were recording, we're doing everything sequentially. So we took that operating system; we took the one for the disk, made it think it was writing out to the disk, but it was actually writing out to the tape, you know, which we had, and then we would process the data from the tape. So that was a lot of fun. And I forgot to say, when I was programming Minuteman Missiles, we had our own specialized computer. It was called the D-33; particularly for hardness because you know things are blowing up all around you when you do that. And it had a rotating disk for memory; in fact, it had four of them, and 128 words on each disk. So one rotation was 128 words, and four of these — that was on the Minuteman 2, Minuteman 1 only had one disk — and you had to program on that. So you had to keep track of where you were on the disk, and you're using assembly language

programming, again, their own assembly language. Anyway, for each operation, you needed to know how long it took. For instance, for clear and add, or an add instruction, it only took one word time, okay? Clear and add took one word time, transfer took one word time, transfer to another sector or another location on the disk which is what we called a sector. But then the divide instruction took, I think it was 12 or 14 word times, so you had to know that because you had a major cycle. So this is like a real time system, right? And I had no computer science degree, by the way. [Laughs.] So the major cycle was basically 130 milliseconds; the disk rotated once a millisecond, okay? Is that right? Once every ten milliseconds, I forget the numbers, but I think one disk rotation was 10 milliseconds. So you had your major cycle, and the major cycle would collect the inputs from the gyros and various other data, and two-and-an-eighth disk revolutions of the 13 was when you could run whatever programs you had to run. Like taking the values from the accelerometers, converting that and figuring out what roll, yaw, and pitch changes you had to make to keep the missile on course. So you did that in two-and-an-eighth disk revolutions. And if you did something like, if you were at 120 and you said add, then your next instruction had to be at 122, if you were really efficient. Okay? You had to be efficient because you had to get a lot done. But if you made this mistake where you were at 120 and you did a divide — and a divide took I think it was 12 or 13 word times — so if at 120 I did divide, and then I put 123 for my next instruction, it would finish at 132. Well there is no 132 because it goes back to one. So now you go back to one, and I'd have to go through another disk revolution until it picked up the instruction. So it was really a combination of things. And these disks were hardened, because like I say that's what you needed. So I learned all that, but again, without ever taking a course. They had

a master programming reference, which was about four inches thick, and they just sat you down with that and said read this. A guy, Mickey Bramble, who's a good friend of mine ever since then, he had been hired in six months before me and he was still reading the master programming reference book, because that's what they told him to do. They hired people because they were busy but they didn't have time to give you a class or work with you. After about a week of this, I said I'm not going to just sit here with the master programming reference. So at lunch time, I befriended one of the more senior engineers and he would explain things to me. And then he started giving me little sample programs to write. And then after I did well with that, he went to our supervisor and said, you know Kemmerer is ready to do programming. I remember he gave me my first programming assignment, which was basically doing this matrix manipulation, but in assembly language, okay? And I remember thinking, boy, did he oversell me? But I did it; I got through it, and it worked and that was fine. I was there, like I say, three years, and I knew assembly; I knew D-33 assembly language and [pause]

Yost: Moving back to the North American Aviation setting for a moment, did you have any exposure to the idea that computer security wasn't just physical security of machines and storage devices?

Kemmerer: Computer security; well, here's what we had. We wrote our code on coding sheets, handed them to a data entry person who punched them in and got us a deck of cards. We would do that between 8:00 and 5:00; 8:00 and 4:42 actually. You hand it in. At night they had their central computer facility that also did the accounting for

everybody, okay? But they would run — because these were classified jobs — and I don't know what time, but like I say between 2:00 and 4:00. And then at four o'clock they would sanitize the machine for an hour or two, and then they would let administrative people use it. At eight o'clock you would get your deck back, and then you would go — or a Mylar tape, depending on what you were doing — then we had labs where we had a version of the gyro-stabilized platform, and it had an entry device and you'd mount your Mylar tape there, and you'd test out your program. But more times than not, it didn't work. But it's assembly language so you could key in changes. You'd have to set *certain* switches to indicate that this instruction at this location should change to something else. I would get it to work, debug my routine, and resubmit it again that night. This clearly was not a very efficient process. It is called periods processing, if you're familiar with security terms. The idea is that you run unclassified, and then classified, and you sanitize in between. It's a terrible, terrible approach but that's all we had. I was in charge of the burnin program, which is where you ran 100 hours of testing on every gyro-stabilized platform before you put it in a rocket out there in the field; you run tests to see if it fails. I said well, what's classified in this document? Does anybody know what's classified? It turns out that there were two values. When you were shooting a missile off, you're doing the calculations and based on that, you determine how much it was going to miss by? I can't tell you to this day —I think I'm still not allowed to —what that was measured in, but there were these miss values; there was a maximum absolute value that the missile could miss by. That was the only classified thing in the whole program. So I said, well, I think we should try this out. What if we take those values out of the program, okay? We can insert them in by hand later. In the programs that we were

working with, in the paper listings that we were working with and so forth that were all classified we had to lock them up every night. Or if you went to lunch or if you went to the bathroom, you know, you had to lock it up. So I knew computer security that way. Anyway, I found out it was these two values so we just took those out of the software and out of the paper listings, and now we had an unclassified document and it just made life a lot easier. You could even run at other times and things like that. So I knew that about classified data, and I knew about physical security, but computer security? No. I mean, we weren't on the Internet; there was no ARPANET. All that came at the end of my time at North American Aviation. The end of my time there was 1970; yes, so at the end of my time, ARPANET came but of course we weren't hooked up to it or anything like that. Everything was sneakernet, you carried a deck, and everywhere you ran it you put it in a machine; so I had no idea what the Arpanet was. And then I worked at UCLA for two years and some number of months, almost three years [pause]

Yost: At the UCLA Institute of Transportation and Traffic Engineering, did you have interaction with faculty while you were at that Institute?

Kemmerer: Yes. But the faculty were from the psychology department. These were mostly psychology tests. We did do, like I told you, these barrier tests. We had a facility at San Pedro, down by the port, but there was San Pedro Naval Base or something there where we ran crash tests. You had this wall and then you had a pulley, and you had a cable that ran around this pulley, and you'd take a car, and you'd have the front of one car hooked to this cable, the back of this other car hooked to this cable, and you'd get the

car up to 30 miles an hour, hopefully before this one hit the barrier and you'd see how much damage the crash did. So barrier crashing, we collected data. We crashed a brand new De Tomaso Pantera one time, because they needed crash information. The study they were doing in that case was how far did the steering rod go back because I think it was only allowed to come back something like six inches. It came back to the seat and then went up; so it would have impaled the driver and then ripped up this way afterwards, in case he wasn't dead yet. So anyway, we had faculty and they were all psychology faculty. We had one faculty who came later, he was actually a faculty at Cal State but he worked with us in research. He was an engineer as an undergraduate so he worked on more of the engineering sorts of things.

Yost: You were there until late 1972?

Kemmerer: Yes, until September 1972.

Yost: That was a little bit before you started pursuing your doctorate?

Kemmerer: Yes. We're never going to get to computer security you know [laughs], which is okay with me. No, I mean it's not okay with me. But so what happened is I had a bunch of college kids, mostly undergraduate students that were working for me, and they were all going to Europe. And I'd never been to Europe, and anyway, what I did was, with one of my students, I took a leave of absence in September of 1971 and went to Europe and traveled around Europe. The student had gone over in June, bought a

Volkswagen camper, drove it for three months — bought it with his money and my money — drove it for three months, I picked it up in Frankfurt, I drove it for three months, and then we shipped it home. On that flight back, I decided that I wanted to retire and just travel around the world. I had no money, but I decided I wanted to retire because I had met some people in Viken, Sweden, that had a 60-feet double-ended wooden sloop that they were going to sail down to the Canary Islands, through the Panama Canal, and then up to California. And they told me I could go with them and just work for my subsistence on the boat, and I thought that would be cool. In the meantime, I met my wife, in June of 1972 and fell in love. But I went on with my plan, which was I was going to retire in September; oh, and when I came back in December I had zero money. But I saved money, and I also did some consulting so I had some extra money. I remember standing with her on a beach in Laguna Beach, a private beach, there, where her friend's parents had a house. I stood there with her and I said to her — we had already decided we were probably going to get married although she still hadn't said yes, I don't think — I stood on that beach and I said, Lorna, if I continue on the trajectory I'm on at UCLA, we could own a house like this someday. Or, we could go travel for a year and then come back. I said I'll do okay, but understand I'm interrupting my trajectory so I don't know. She said she would like to travel. So our plan was that I was still going to retire. Then it turned out that the boat got delayed, and it was going to be delayed by a year. So my new retirement plans were I was going to spend the fall — and actually this was before Lorna — I was going to spend the fall in the Yucatan Peninsula, and then I was going to spend the winter skiing because I was really into skiing, and then I didn't know what I was going to do after that. So then I met Lorna, she had been at San Diego State and was in

the process of transferring to UCSB. And so that fall I actually came up here and stayed with her in the fall. I didn't go to the Yucatan Peninsula. I did go to do the winter of skiing, and when I came back in the spring — oh, and then the plan was a winter of skiing and that following July, we got married — and we were going to travel around Europe. But I got this consulting gig, and the consulting gig was enough that we decided we could travel around the world. And so we got married in July of 1973. We also decided that since she had never been east of Nevada, we should travel around the U.S. first, because you should see the U.S. if we're going to see the world. And so in September, which is when my consulting ended, September of 1973, we got a Volkswagen squareback and converted it into the place we slept and ate out of, and we traveled around the U.S. for three months. My parents were still in Pennsylvania so we stayed with them for a while. We came back in December and booked around the world flight tickets to — we were originally going to fly to Katmandu. We were going to fly to Katmandu, and at that time there was like the hippie express. These guys would take camper vans, drive to Katmandu, get hooked on the dope and other things there, and then you could pick up a van fairly cheap, and we wanted to drive at least through the Mideast. You know, young and stupid. But what happened was, in December of 1973was the first gas rationing. Remember we had gas lines in California? You filled up odd on one day, and even on another day, and you had the long lines. This was about the time we were doing our trip planning, and I said, you don't know; this is the first time we had rationing since World War II; for us, that we had gas rationing at all. So I said let's buy airline tickets all the way around the world; we'll fly to Katmandu, stop in Iran, and Afghanistan, and some other places. And I had a friend, who is a nuclear physicist, and

was in the ski club with me at North American Aviation. He decided he didn't like what he was doing and became a travel agent. So I went to him, he explained to me how you set up tickets, and Lorna and I set our route up for this whole thing. We planned it so that every leg we were traveling on a different airline, except to go to Berlin because you could only fly on Pan Am into Berlin and back out again. Anyway, we set that all up, and then we traveled around the U.S. for three months, came back in December, and in late January or early February we flew to Hawaii. Hawaii was where Lorna was born, so we spent a month there living with her aunt and uncle, and then we took off and traveled around the world for seven, eight months. We came back in August in time to see Nixon resign. In the meantime, Lorna had completed a year at Santa Barbara, and she liked Santa Barbara, but she wanted to go to UCLA. So while we were on this trip, she applied to UCLA. She needed five quarters to finish her undergraduate. So when she was applying, I said, if you're going to be there for five quarters, I ought to go and apply to the computer science department and see if I can get in there and learn what all this is about. And so I got accepted, and I didn't apply for a TA or RA. We both got accepted. I went and applied for a job, because I knew people there. The guy who I told you about earlier who called me and said I don't want my job, I don't want to be a manager; well he was now working for Len Kleinrock, running all the DARPA machines. One of the guys that used to work for him and later worked for me when I was at the Institute of Transportation and Traffic Engineering was also working for Kleinrock. So when I got to UCLA I talked to those guys, and the next thing I know, I was working for Kleinrock. And so I was working as a programmer, not as an RA, but as a programmer for Kleinrock 20 hours a week to pay for us to go to school. So in the fall of 1974 was when I started at

UCLA in computer science. And it was wonderful because I'd take these courses — I took this one dumbbell course, one of these introductory told-you- about-all things, and the professor showed us what an adder looked like, and what a divider looked like, and basic things like that. When the professor was showing us how the divide operation worked, all of a sudden I understood why that operation took 12 word times instead of 1. If it was a cartoon, a light bulb would've gone on in my head, because I went wow, this is really neat! And meanwhile, I looked around the room at the other students and they were just sitting there like, you know, it didn't mean anything to them. That was a great time. I enjoyed being a student. But then I think my second year there, I was in a class with some guy, and he worked for a faculty member, Jerry Popek, who was doing computer security, building a secure UNIX; data security UNIX kernel. What I was getting interested in was formal methods and in fact, in the class that we were taking, we were combining formal methods with something else. And he said you ought to go talk to Jerry Popek, because we would like to formally verify the kernel. And so I wound up working for Jerry Popek. And oh, by the way, I only applied for the master's degree, right? So in five quarters, I got my master's but the last year, last two quarters of that, I worked for Popek writing formal specifications for the data secure UNIX operating system. A funny thing happened before my last quarter, which would have been January of 1975. Every quarter you would go up to the dean's office if you were a grad student and fill out a form. And what this form asked was what's your degree objective and when do you expect to get it? So I put down master's degree, computer science, and I said March of 1976. Well, there was this Iranian guy that started at UCLA the same time that I did, but he was a Ph.D. student, and we had taken a lot of classes together. His name was Farid,

but I can't remember his last name. But anyway, he was up there filling out his form the same time I was, and he looked over as I was filling mine out — we were friends, he knew me — and he said, what are you going to do? Wait another eight years and then come back for your Ph.D.? Because I had been out eight years before I went for my master's.

Yost: Right.

Kemmerer: He just said that in passing, and he doesn't know because I haven't seen him since then — I must have seen him but I don't think I ever told him — but that made me think. And my wife was also graduating in March of 1976, but she was going to get a job. I talked to her about it. I got along really well with all the faculty and part of that was that some of the faculty — you know, my advisor was five years younger than me — and I think because I was older; and UCLA was like that; it was a friendly place. I got along really well with him. Because Farid said that; it started me thinking, he's absolutely right. I mean I said, I'm as smart as these guys in the Ph.D. program and I get along with the faculty, where could I get a better situation? And so I asked, I checked into it, and they said great, we'd love to have you go on for a Ph.D. But just because of that one moment in my life, you know, where people in my life tell me little things that direct me.

Yost: Was Jerry Popek your primary advisor?

Kemmerer: No, my primary advisor was a guy named Dan Berry. And Dan Berry was programming languages, but he also did formal methods and I did my master's thesis on a debugging system, and I did it with Dan Berry. But Popek, just like I used to work for Kleinrock but my advisor was Dan Berry; and then I worked for Popek but my advisor was Dan Berry. I was looking for a Ph.D. topic. And in the meantime, as part of my master's, I had taken a computer security class from Jerry Popek. So I knew about computer security and I thought it was cool, and we had our computer security group and we were building data secure UNIX but we were talking about a lot of other topics too. I learned a lot of computer security, especially for that time, because there were probably only two or three, for sure not five, universities that had a computer security course at that time, and Jerry had worked on the M.I.T. Multics system, even though he was from Harvard. And so we got a good history of computer security. I worked for him, and I was looking for a PhD topic. I actually had a topic for a year, but I was going nowhere with it. It was a programming languages topic. People used to say, you ought to make what you're doing for Jerry your topic. And somehow I thought it was not good enough for a topic. In my mind I thought that because what did I know? So I went to the professor that was known to be the toughest guy in the department, Dave Martin, and Dave also did formal verification. I sat down with Dave and I said listen, people keep telling me that this would be a good Ph.D. dissertation topic. He said yes, absolutely, no doubt about it. I said would you be on my committee if I changed to that topic? He said yes. Then I went to Popek and I had Popek and Dan Berry as co-advisors, because Popek knew the security and Dan Berry knew the formal methods. So I basically wrote a formal policy for data secure UNIX and I wrote high level specifications and low level specifications. I

then proved the PASCAL code that we implemented it in, and that was basically what my dissertation was. But like I said I spent this other year, wasted time doing something else. But I guess it wasn't wasted because I learned a lot. So then I finished in June of 1979.

Yost: It would've been one of the earliest dissertations on computer security. Dorothy Denning was certainly was several years earlier (1975), but there were not many at that point, 1979?

Kemmerer: Yes, Dorothy Denning had hers. Another was Anita Jones from Carnegie-Mellon. She had done, I believe, a secure operating system design. There were a couple out of M.I.T., but not much. I don't know; Paul Karger, when he worked on Multics, he didn't have his degree, because he got it from Cambridge much later. He got his Ph.D. after me. He worked on the original Multics system, but he was in the Air Force during that time. So yes, I had one of the earliest computer security dissertations. It was also the first typeset dissertation out of UCLA. [Laughs.]

Yost: Were there other graduate students that worked with Popek on the secure UNIX project?

Kemmerer: Yes. There was a guy named Bruce Walker. He did some work with formal methods but what he was doing was looking at the code. Charley Kline, who was the guy who sent out the "hello" on the ARPANET for the first message from UCLA to Stanford. So Charley Kline was there and there were other grad students that were working on

computer security. We had a group of maybe 15 or so students working on different projects. But then Popek moved into distributed systems. He developed a new approach to distributed systems, started a company called Locus, Inc. and actually left campus. He was associated with UCLA up until his death a few years ago; I think he was an adjunct professor. But once he started his company he pretty much wasn't on campus anymore. Just before he left, most of his group was concentrating more on distributed systems. So he had something called Locus, and they were bought by IBM, I think. Anyway, they sold some of their developments to IBM, but then they were bought by somebody else; I don't know who.

Yost: So as you were working on this research for your dissertation, did you get a sense of the context? Were you aware of the Anderson committee and the work that Roger Schell was doing leading the ESD research program for computer security? Of the subsequent MITRE work of Bell-LaPadula?

Kemmerer: The answer to that is probably no. We might have looked at Roger Schell, although I don't think we did. Some of the things we knew, because we had papers that we read in Jerry Popek's course. I was aware of MITRE building a secure kernel. What I was most aware of; I did the research, basically, on security kernels, and I found that MITRE was building something called KSOS, Kernelized Secure Operating System. I don't think they were formally verifying it, but it was a secure operating system. At SRI, they were building PSOS, and PSOS stood for Provably Secure Operating System.

Yost: Peter Neumann.

Kemmerer: Peter Neumann, right, and they actually had a report on that. And the report said that they formally verified their provably secure operating system. I got a copy of that report, and, of course, my heart sank because I said someone took my dissertation topic, you know? So of course, as a good graduate student does, I read the whole report very carefully. And they did have formal specs, possibly — I don't remember — but possibly for all of the operating system. They proved two specifications, two small specifications for a small piece of code, and I found an error in one of the proofs. So I contacted Peter Neumann, and that's the first time I met Peter Neumann, and we became great friends, which we continued from there on. And in fact, when I came here, my final decision was do I want to go to a research lab, which would have been SRI, or do I want to go to the university? I chose UCSB and decided to come here. I told you, I never know what I want to do. [Laughs.] At that point in time I was 35 and I had a child, and a wife, and still didn't know what I wanted to do for sure. So I was aware of their work.

Yost: As you were doing your dissertation research, was there a connection yet with SDC?

Kemmerer: No. What happened was I accepted my position at UCSB. I had the summer before I came up here so I was looking for a job. I think I had actually accepted a job with somebody who had a small company in LA and was supposed to start right after the 4th of July with them. About a week or two weeks before that, one of my fellow grad

students who worked at SDC said do you know we're doing stuff with formal methods at SDC, because he knew I did formal methods. He said we're looking to hire people; would you be interested? So I basically went in for an interview Thursday or Friday before the 4th of July at SDC, I saw that they were working on things that were right up my alley — I mean, the match was like made in heaven because what I did for my dissertation was what they were doing. They had their own language. I did all my proofs manually. While I was at UCLA, the last year or two years, I had an office at ISI because people down there were doing formal methods work. And it also gave me a chance to get away and get things done, because I had my own private office. I worked with Ralph London and Susan Gerhart, who were software engineering formal methods people. And so I had lots of information on formal methods, but it wasn't being done for secure operating systems, other than the PSOS work. And so anyway, I had this interview with SDC, and he said how much do you want to make as a consultant for us? I told them an hourly rate; I think I gave them a range and he said we'll pay you at the top range. So that was great, and I went to work for them. I spent that whole summer working full time for them. I became very familiar with Roger Schell and the Multics work; I don't know if I learned Multics that summer, but it was a great place because all the people that were interested in security were coming through there. When I was still at UCLA, IBM put together a task force to go look at computer security, and I think they were particularly interested in secure operating systems, although I'm not sure. They visited UCLA while I was there, and I met with them. And then they came later to SDC when I was there. So things were just starting to happen and it was a great time.

Yost: Do you recall anyone in that IBM group?

Kemmerer: Yes. One of them was David Lomet, who is now at Microsoft. He left IBM, went to Wang Institute, and I actually was at Wang Institute as part of my sabbatical while he was there, so we sort of reconnected. I only knew him vaguely; we never knew each other well. Then he went to Microsoft and he does database work, mostly. The guy who ran the study [William C. Carter], I can't remember his name. But Dave Lomet is the one I remember. He's one of them. If you were to connect, if you're interested, he would know who those people were.

Yost: And so right from the start, you got to know Clark Weissman?

Kemmerer: Oh absolutely. I mean, Clark Weissman; I worked right with Clark, so I was kind of at that level. I had a supervisor. It was a guy named Tom Hinke; and the guy that hired me was Harvey Gold. So Harvey Gold was sort of a branch manager, and Tom Hinke worked for him. Clark would call me into his office and get my opinion on different things. It was great; he's a great guy to talk to.

Yost: Obviously, Jerry Popek's one of your intellectual mentors early in your computer security career.

Kemmerer: Yes. If I hadn't taken his computer security class or if he hadn't had his group, that wouldn't have happened. Yes.

Yost: Are there some other people that [pause]

Kemmerer: Dan Berry is the guy that got me into formal methods. I had taken his class; and probably Dave Martin, also. And so they kind of got me into the formal methods, and Jerry Popek clearly got me into computer security. Going to SDC broadened my understanding — you know, I knew about all sorts of things. There was a guy there who was blind, and he did Data Vault. He was interesting because there were companies using computers to aid the blind, and they would always come to him before they added a new feature. So I'd see all kinds of neat things, like optical readers that would then go to voice, you know, so he'd have a new toy there all the time. Glaser was his last name; it was Ted Glaser. Yes.

Yost: He was at M.I.T., and then he went to Case Western.

Kemmerer: Right, then he went to SDC after that. Has he been interviewed by you guys?

Yost: Unfortunately not. He passed away a number of years ago, long before we started this research project.

Kemmerer: He has passed away, yes, but did you get [pause]

Yost: Nobody from our institute got to interview him and to my knowledge there is not an oral history interview with him—at least not one published or in a public repository.

Kemmerer: Because those are the guys you wish you had.

Yost: Definitely. Two that really stand out are Glaser and James Anderson.

Kemmerer: Yes, Jim. I love Jim. So I started working for SDC and they had their own formal specification language and their own theorem prover. A guy named John Scheid designed the specification language; it was called Ina Jo, which was the name of his first wife, who he was no longer married to. [Laughs.] He wasn't married to her when he named the language after her so you could interpret that however you want, and Val Schore was the guy who built the theorem prover. Neither of these guys were what I call front men, and they had done good work. There were other systems out there, but the SDC system didn't get the credit it should have. I came in there and I also knew about some of these other systems, because I had studied them, such as the algebraic spec work that they were doing at ISI. Dave Musser was at ISI for a while. I worked with him on stuff like that. So I came in, what SDC had was really great, and it was the sort of thing that I thought gee, if I had had that when I did my dissertation I would've used it. I did use the theorem prover at ISI to do some of my low level proofs, but everything else I did, you know, I defined my language and did the proofs manually. But anyway, at SDC, these guys were great, but they weren't great presenters. We were working on something called **COS-NFE,** which was a secure network front end for the Air Force, and we were a

sub doing the formal methods part of it. A group called DTI out of Champagne-Urbana, if you're familiar with Champagne-Urbana, their offices were in the Holiday Inn, which was one of those round tower Holiday Inns. DTI rented one whole floor of it. Anyway, we did formal methods for that. I worked with them; I used a system to do that. And in the fall of — I think that wasn't the first year; it was the second year —what I did was I became the front man for Ina Jo and the theorem prover. And in fact, Don Good, who is a formal methods guy from U Texas, he referred to me as SDC's hired gun. [Laughs.] But they needed a hired gun, and I knew and understood all of the systems. I don't know how I got on this tangent, but that's how I met a lot of these people. I met Roger Schell. I worked with Val Schore, and John Schied; and Marv Schaefer worked in the group, too. He kind of advised the formal methods people but he wasn't, you know, he was not a hard core not-then-and-not-now a hard core formal methods person when he was doing it. But he's a good, broad security guy.

Yost: You and he had a background in mathematics. Can you talk about how that helped you in this sphere?

Kemmerer: It's interesting because my background in mathematics was theoretical math. If I had been in applied math I might have stayed in math. But the most applied thing I did was fiber bundles. Do you know? Are you a mathematician at all?

Yost: No, my knowledge in mathematics is limited to completing a few undergraduate courses.

Kemmerer: I studied real analysis, complex analysis, and topology. I think that's why computer science was a draw for me, and I think the only thing that I got from math is I think mathematically. I think of proofs. You can't just tell me something, you can't say "clearly we see that…"; I have to work it out. So my mind works that way. I don't know of any math courses that I took that helped me when I was doing formal methods. I only ever had one logic class and that was at Penn State. We studied Hume, Tillich, and those people. I didn't have the kind of logic training that I needed, right? So what I did before writing the final version of my dissertation, I asked people at ISI what was a good book on formal logic, and I sat down and read that book. That was the first time I knew what *modus ponens* was, and *modus tolendo ponens*, and all those logic terms. But once I knew those, then I could take what proofs I had done, build a consistent form for my arguments, put these right words in there because I knew if A implied B, and A was true, well then B was true, but I didn't know that was *modus ponens*. So now I could put all the right terminology where I had justification for each step. But I can't say that was from a math class I had or a logic class, it was just that I sat down with the book and read it. And the formal methods, I think, just logical thinking. I always thought logically. Did you ever read *Zen and the Art of Motorcycle Maintenance*?

Yost: A wonderful book.

Kemmerer: A wonderful book: I just reread it. I read that while I was a grad student and I just reread it again about a year ago, because I had recommended it to a colleague in

San Diego who then started asking questions about it. There's not as much of it as I thought, but in that book I have always thought that his approach to solving a problem is exactly what I do. I mean, I think I can fix almost anything, you know, if it has a broken part I can't manufacture it, but I can take things apart, put them back together, and I love to do that. Of course I have lots of things that are apart that I haven't put back together yet, but you know. So I think that logical thinking is what made formal methods fun for me. Again, I thought it was so cool when I first saw it, and I stayed with it.

Yost: So SDC is a real center of activity in computer security

Kemmerer: All kinds of computer security, yes.

Yost: . . . in the late 1970s and into the 1980s.

Kemmerer: I was only there from about 1979 on.

Yost: Had you started to go to any of the major meetings in computer security at the start of the 1980s?

Kemmerer: Yes. In May of 1980, Peter Neumann started something he called the VERkshop. He likes puns.

Yost: Lot of the computer science people [pause]

Kemmerer: Yes, but Peter Neumann is known to be one of the worst; Steve Lipner a close second; Ted Linden a close third. But Peter Neumann is definitely known to be one of the best, or one of the worst, depending on how you view it. So he started something called the VERkshop in May of 1980, and that was also when the first [IEEE] Security and Privacy Symposium was. I went to the VERkshop, and I went there for SDC. And part of what I did was I presented the Ina Jo language and the Ina Jo system, and said here's what we have specified. Schied's work was right on. He didn't think about what theoretical approach he was programming; He did it because in his mind he knew that was the way to do it, but he didn't know why; for example, here's your justification of why this is the right way to do it. So I went to the VERkshop and I presented for SDC, and that's when I became their front man. And then in fall of 1980, there was I think the very first National Computer Security Conference, I think we called it, and it was at Linthicum, Maryland. I went to that and they had a session on the different formal verification tools that were around then. Don Good was there with Gypsy; Dave Musser was there — I don't know if he was there with Affirm at that time — probably Susan Gerhart was there with Affirm, and Dave Musser had moved to Albany, and he was doing another algebraic spec language. SRI was presenting Special; probably Larry Robinson was presenting it. And I was presenting Ina Jo for SDC. At that meeting is where I first met Jim Anderson. Excuse me; names I used to have, they used to all be right there. So at that meeting in Linthicum, Maryland, I believe it was November 1980, at night a bunch of us went out to dinner. Someone said to me — I'm not sure I knew the Anderson Report then, no, I must have — they said oh man, Jim Anderson is going to

tear you apart, because he hates formal methods people, right? I sat down next to Jim, you know, I don't remember if it was a coincidence, but I'm the kind of guy who would have done it purposely; anyway I sat down next to him. At the end of that dinner, we were best friends and he said, you're the first one that could explain formal methods to me in a way that I would believe it. Before that, it was all hog wash. It wasn't just because of me, it's that a lot of it was oversold before. I mean, people said we are going to prove that this system is formally correct. We never do that today. What we do is prove that the system is formally consistent with its specifications and if its specifications are correct with what you want it to do, why then okay. But you're starting when you write the formal spec, and you can test the specifications and things like that, but to try to tell yourself its good; testing specs is just like anything else, if you don't try the right test on it, it's not going to show you if you have a problem. But so anyway, Jim and I became great friends from then to the end, and that was nice. I don't know when I became aware of the Anderson Report, but Roger Schell was still in the Air Force when they started the National Computer Security Center, I think. Is that correct?

Yost: Yes.

Kemmerer: So he'd show up; he was probably at Security and Privacy. I might have missed the first two Security and Privacy meetings, the IEEE meetings at Oakland that are usually referred to as the Oakland Meetings. And that was because the VERkshops both times were scheduled the week before Security and Privacy, and as an assistant professor, I didn't feel that I could miss two weeks of class, which was too bad, because I

missed those beginning meetings. But I think Roger might've come out to visit us. I met Roger later on, but when Roger was at; well, when they started the National Computer Security Center, which was in 1982, so probably around spring of 1982 they started National Computer Security Center. Marv was hired in as Chief Scientist. Oh, something happened before that. So Steve Walker, who was the impetus for starting the National Computer Security Center when he was at the Pentagon [pause]

Yost: He had been putting together NBS meetings, did you attend any of those?

Kemmerer: Okay, so the NBS meeting, the one that was at Linthicum in 1980 was, I think you're right, I think we first referred to those as the NBS meetings.

Yost: Okay.

Kemmerer: I think that's right. And I think it wasn't until they moved to Maryland that they called them the National Computer Security Conference meetings. I probably have the proceedings here; I could find out. But what happened was that the project we were doing for SDC that I was working on, and I wound up running that project, even though I knew I had to come up here as a full time professor in the fall. So I was in charge of that project and working directly with the guys at DTI [Digital Technology Incorporated]. Steve Walker knew he wanted to start this thing called the National Computer Security Center, and he knew one of their goals was formal verification of systems. They were working on the Orange Book at that time, but it wasn't out yet. He knew that he wanted

to have people writing formal specs, and he wanted groups who would go around and evaluate the systems. What's an evaluation going to look like? And so he chose DTI's COS-NFE, as a project that was sort of a zero test of this. We would go back to meetings at the Air Force, and all these guys, you know, they had a group of people. Morrie Gasser from MITRE was there, that was before he went to DEC. There was also Annie Discepolo and another guy from MITRE [George Huff], and there was Anne Marmor-Squires, who was from NSA at the time. So there was a group of about five on the evaluation team, plus Steve, and they'd sit and ask us a question. We had a meeting with the Air Force one day and a meeting with the evaluation team the next day. I don't remember the order of it, but the Air Force was so much easier, because the evaluation team was riding us. And again, I was from the SDC side, I was the guy who was leading that effort and defending it. And so that happened, like I say, started; I don't know when the first of those were but we had those meetings for two or three years until the COS-NFE project was over.

Yost: So I haven't asked you yet about your start at UCSB; you were hired here and working a day a week at SDC at that time. I assume you were the only computer security researcher in the UCSB Computer Science Department at that time?

Kemmerer: Right. And I was the only computer security researcher in this department up until; I'd have people who would come visit me; well, we didn't have another faculty member in computer security until we hired Giovanni Vigna. Giovanni came to be a postdoc with me in October of 1997. He was supposed to be here for five months. We got

along so well, and I liked his work ethic, that I asked if he wanted to stay for two years, so I got another DARPA grant so I could support him. And after two-and-a-half years, so it would've been 2000, I guess, we hired him as an assistant professor. So that was the first time I had a colleague in computer security. And then we hired another guy who also was a postdoc with me, Chris Kreugel. And we are the core computer security group. We have worked some with other people in the department, like there's Elizabeth Belding-Royer who does *ad hoc* networks. What we did is we ran a seminar with her on security of *ad hoc* networks, and we were designing intrusion detection systems at the time, so we actually built an intrusion detection system for *ad hoc* networks. But that's the only work I did with her. We have these three core security faculty. Starting this fall, we hired two crypto people, assistant professors who were both postdocs at M.I.T., and so they're part of the group. We have people working on the architecture and hardware side of computer science, computer engineering, and we are doing security work with them, and we were doing some service-oriented architecture research where we work with one of the other guys who did some data mining. So I work a little bit with some, but in terms of security people, before Giovanni, there was nobody.

Yost: In 1979 when you came here, were the senior most computer scientists in the department open to this new research area? Obviously, you were offered the position, but when you started how did they view this research specialization?

Kemmerer: They were thrilled by it. They really liked it. And they liked it not, I don't think, because of computer security, it's because I'm a systems guy. I mean, I was an

operating system guy. At the time, there were a lot of theory guys, but systems guys were hard to hire, and I think that's what they liked. And I did computer security so that was okay; I was also doing software engineering.

Yost: Did you teach courses on operating systems?

Kemmerer: That's interesting; I have never taught an operating systems class and Giovanni and Chris both teach operating systems classes. You have to understand that when I hired in; well, when I interviewed, there were 4-1/2 faculty in the department. 2-1/2 of them were full professors, and it was half because the other half of one was in electrical engineering. So 2-1/2 were full and two were assistant professors. I was hired in, in July 1979. The department was started in December of 1978, and so they had some people working in electrical engineering doing computer science, and some people in math doing computer science, and there was a little bit of friction between them. The university decided the way to solve that was to form one department, which is a computer science department. So I was the first one actually hired from the outside into the Computer Science Department; the others formed it. One of those senior professors was a guy named Ed Coffman, who had been at Bell Labs and then went to Penn State, and came here. Len Kleinrock's first Ph.D. — and his first student, I think; that's maybe who it was but I'm not sure. Anyway, Ed Coffman left before I started. He decided that he wanted to live in the New York area and he went back there. So now we had 1-1/2 full professors and three assistant professors. And of those two assistant professors, one was in software engineering and the other was in programming languages and did systems.

And those two didn't get tenure or left before they got tenure. So I was the first one who got tenure; the whole school didn't have a history of tenuring computer science people, so I felt good about that. But computer security, we didn't have anybody until Giovanni came.

Yost: I didn't ask you about the early publications that you and Popek and Walker did. Some high profile publications like your 1980 article on the UCLA Secure UNIX kernel in *CACM*, the flagship journal of the ACM. Can you talk about the reception to that research by the broader community?

Kemmerer: That was damn good work, first off; and it was well received. And when SOSP, you know that's when you still used to have it up north by Monterey, at a great venue, and no, everybody; I mean, all of the information we got back on that was really good, really good feedback. One of the things I remember from that meeting was I met a guy named Bjarne Stroustrup. Does that mean anything to you?

Yost: C++

Kemmerer: Yes. He was at Bell Labs at the time. He saw the formal methods work we were doing and how we presented that, he was very interested and said hey, I've got this new — it wasn't called C++ then — and you know, it's abstract. Because when I did the data secure UNIX work, the refinement part, I defined the kernel as an abstract type and each of the operating system calls as a call on the abstract data type; and there was theory

that was out there already. So you know, you would assume the entry assertion and you have to prove that the exit assertion holds. And if you prove an initial condition, then by finite induction you can conclude that the reachable states satisfy whatever your secure invariant is. And C++ works that way, when you're working in modules. So Bjarne was very excited about it and said I have a system — which I don't remember what he called it — but I remember him sending me a big nine track tape of the system to run on UNIX — which I don't think I ever ran, we were all too busy for that. But no, that work was well received. The COS-NFE work, the secure network front end, I wrote specs for that. We were doing covert channels on that. That's where I developed a method called the Shared Resource Matrix approach to covert channels. And I remember what happened was there was COS-NFE, the NFE was a network front end and it was a proprietary network front end developed by DTI. Because it was proprietary, I couldn't publish on it, okay? So I came up with this approach to finding covert channels; and, in that paper, I also made the distinction between timing channels and storage channels. I defined those things, because there were papers out where they mixed those and so I made a point of saying here's the difference between them. The problem was, in order to do this, I came up with a toy example and I showed on the toy example what channels you could find. And then I stated that we ran this on a real system, network front end, you know, DTI's COS-NFE, here's the size of the system, and we found some problems, but because it is proprietary I can't show it. Well, even before that, I'd go to the guys at DTI, I'd say how long is it going to be proprietary? And they go probably six months, maybe. A year goes by. And then finally, I did the toy system and submitted the paper. And I remember submitting that paper to either *CACM* or to *IEEE Transactions on Software Engineering*

because there was a problem; there was no place to publish security papers. IEEE and CACM, for our SOSP paper, one of the reasons that got in was Stockton Gaines was an old-time computer security operating systems guy, was a good friend of Jerry Popek. *CACM* had editors for different departments and so he took it and made sure it got reviewed and got pushed through. There was just no place to publish security stuff. I must have submitted to *Transactions on Software Engineering,* and they just came back and said well, you know, it will never work on a real system. I said, how can you say that it will never work on real system? I'm telling you there is a real system, I'm giving you a size, I'm telling you how many channels we found, but I'm not telling you what they are. As an assistant professor, I was depressed, you know? I remember taking that paper and putting it in my file cabinet and doing nothing with it. Then John Bruno, who was our chair, came by some number of months later just to talk. He said how's it going; what are you working on? I told him about this paper and he said let me see it; let me see the reviews. He read the reviews and he said, they don't hate your paper. It was a learning lesson for me; Bruno said if they say it doesn't do this, then you go either it doesn't do it for this reason or it does do it and here's how I didn't explain it too well, blah blah blah. And in the meantime, I had met Dorothy Denning. I think I met her at one of the NBS meetings, possibly that very first one; I probably met her and Peter there. I met her and we became good friends. She was starting a journal and I think maybe I told her about this paper and I sent it to her, and it was accepted for the initial volume of *ACM Transactions on Computer Systems*. So it finally got published in there, taking the advice that Bruno gave me and doing that. But no, a big problem for security researchers was that there wasn't a place to publish. Now I had a better chance than a lot of people when I

did the formal methods, because even with my shared resource approach, the paper I wrote described how the shared resource approach works on formal specs, and it works on design level English, and it also worked on code; and I showed all three in the paper. So having the formal methods and having had system development made *IEEE Transactions on Software Engineering* a place to publish. So in terms of journals, that was the place where I would go to publish that work.

Yost: In the early years, how were the proceedings of the Oakland Conference perceived by the broader reaches of the computer science community?

Kemmerer: The problem we had and the problem computer sciences had is, you know, what counted were journal pubs, but there weren't computer security journals. So you have Oakland. NSF actually did a study on this, which was very helpful with us here; we would always make the case that our conferences, like getting into Oakland, is much harder. Today it's still that way. And now people are more accepting. The guy who really pushed that was Dave Patterson at Berkeley. He wrote an interesting reference letter for somebody and I have a copy of it. It says, "There was a computer scientist who came up for tenure and only had one journal paper." But meanwhile, Dave said, he had already done the RISC architecture and accomplishments like that. I don't think he was in the National Academy of Science, yet. But he said, "and they gave him tenure," and he blah blah blah did all this stuff; and then he said at the end of the letter, he said, "this guy is me." You know, making his case that the journals aren't what count. Even the number of pubs isn't what counts; it is how much impact did they have? So we had to do that here.

Fortunately, I was working in formal methods. I got into computer security as an application area for formal methods. When this guy named Dave Farber — not the guy named Dave Farber from Delaware — Dave Farber was a student who told me I should go talk to Popek and work for him. I was looking at that as an application for formal methods and I did that, and I broadened my research into security.

Yost: At what point did you begin to teach courses in computer security?

Kemmerer: I could give you an estimate; I'd say 1982, but it might've been earlier. Oh, maybe I don't have them here; I guess I don't have them all here. I would guess 1982 or 1983. When I came here, you asked me about operating systems. Data structures is our core course, and so I started off teaching data structures, and that was two quarters. I also taught programming languages, and then I taught a graduate class in software engineering. I probably did that for the first two years, and then I started buying out of a course, and then we had only three courses a year. So I was doing mostly programming languages and software engineering. Then I taught a graduate class in formal methods. And then somewhere like, I would guess, 1982, 1983, I taught my first computer security class and I taught it as an undergraduate class. Oh, I know what happened, before that I made the argument that we should have an undergraduate course because our software engineering class was a graduate class; in fact, I think it was a two-sequence class, and I taught one of those. And then I made the argument that we should teach software engineering at the undergraduate level, and so I was teaching software engineering and programming languages, and then eventually, I talked them into doing computer security

and I dropped the data structures classes. This is why I never got to the operating systems because it was easier to get an operating systems lecturer or somebody from the town to do that. I know I went on sabbatical in 1985; I know that I had done the course at least two times before that, maybe three. So I would guess it was probably 1982 that I started it and then I taught it every year since then. I'm wrong. I started it as a graduate class in 1982, and I made it an undergraduate class; that's what this is here. I made it an undergraduate class in 1999. And then I could do that and my formal methods. But my research always bounced between both of those. Sometimes I'd be doing mostly formal methods; sometimes I'd be doing combinations of the two.

Yost: Did you have graduate students in both areas in the 1980s?

Kemmerer: In the 1980s, I guess what I'd have to do is; I'd have to look in my publications and see what I published because I'm thinking a lot of stuff I did was just me. I'd be doing these other things, like digital sound, with a grad student, which was completely off what I did but there was nobody else that did that and so I would pick up the student, learn the area, do it with them. But in the meantime, I was doing my security research. I definitely had security people working for me in the 1980s. My first Ph.D. student, his Ph.D. was strictly formal methods. It was in a language called **ASLAN** that we developed, based on other languages that I was familiar with. We did some work for Steve Lipner when he was at DEC and they were trying for A1 certification. Before that he gave us English specs for the system and we were writing formal specifications for those English specs here, a couple of students and I. I met Steve in 1982 at the Air Force

Secure Database Summer Study that Marv Schaefer was in charge of. He was good for me, because he gave me some research money and some equipment. We wrote specifications for him, and then when they decided to build their system to be A1 certified, then they wrote their specs in Ina Jo, and I went back and I taught a class on Ina Jo for them. That was probably around 1985 or 1986, something like that. I used to do week long classes for SDC at NSA, and I did one at MITRE, and I did one at DEC but for SDC, under the auspices of SDC.

Yost: Were you among the advisors of *TCSEC*, or did I misunderstand?

Kemmerer: *TCSEC*?

Yost: Yes. The Orange Book.

Kemmerer: An advisor on it? No, I worked at SDC and the Orange Book was being developed then, and the Red Book. They sent it to SDC to have you go and decide on it. I don't remember what level Marv was at that time, but Marv would often give me things to look at. So I had looked at those. I don't show up anywhere on the Orange Book or the Red Book as being a part of it. What I did do is when we were doing this COS-NFE study, that was sort of the dry run for how the National Computer Security Center was going to do their evaluations. But there wasn't a National Computer Security Center then, because it was formed; I mean, Marv went there in the spring of 1982, and so it might have been formed at the earliest in the fall of 1981. And maybe not until spring of 1982, I

don't know, but right in that time frame. So they were doing the evaluation with us, but it wasn't a real one; it was sort of their dry run. I also was on some of the evaluation teams and things like that. The National Computer Security Center had something called the Formal Methods Advisory Board, or something, which was mostly people from MITRE and me, and maybe somebody from Aerospace, I'm not sure.

Yost: Do you recall what your reaction was to hearing there'd be a National Computer Security Center that would develop this criteria and evaluation of infrastructure?

Kemmerer: I thought it was good. I believe that if you have any system that you understand and that you can convey to me your understanding, I can write a formal specification for it. And I think formal specifications are really powerful. Not only doing the proof because you understand what you're doing, because in making you write it down formally, I poke at those questions where you think you understand it — and maybe you *do* understand it — but you hadn't thought about it and it makes you fill in those sort of things. So I thought it was great. I also thought it was interesting because we had dealt with NSA when I first started at SDC. They had a project called "Blacker." You weren't allowed to say the word "Blacker." And if you worked on Blacker, you worked in a vault, you had to go through a set of doors for it. I knew how tight all of that work was. I remember being at a meeting at SDC; we were talking, and somebody said Blacker out loud. I remember thinking oh crap, is he going to be in trouble! Well they had finally outed it, so to speak.

Yost: Was it disappointing that industry didn't participate more once the Orange Book came out, in developing systems at high levels of security, A1, B2?

Kemmerer: I think the disappointment was more with the government. Steve Lipner was saying that there wasn't an interest. But when the government was selling formal methods, they weren't selling any one approach, they were selling the whole evaluation and said we want these. I don't know if they said it straight out but it certainly was implied that if you deliver one of these they will buy it. And that wasn't happening, so I don't think you can fault the companies for not doing it. If you tell me, build me a car that can go from zero to 50 in two seconds and can stop on a dime, where I define what I mean by a dime, and I'll buy 100 of them from you, or something like that. Or I'll buy whatever you can produce, which is usually how these things are stated. Then you go and build one and they don't buy any, and it does what it's supposed to, you're going to stop building them. And the next time they say something like that, you ignore them. I think that's what happened there. There is also sort of a; NCSC never — and I don't even know what the situation with it is now — NSA was always so closed. For instance, when we did the first crypto conference here on campus in 1981, we didn't know whether they were going to come or if they were going to close us down. But they showed up and their badges said "Columbia, Maryland" or they might say "Defense Department." Most of them had their name, and Columbia, Maryland, but you knew where they were from. Nice guys and all that, but it was all closed. And there were all the different sides of NSA. Somebody who you would talk to and all of a sudden they would go over to the other side or whatever, the dark side, and you wouldn't hear from them for five years.

And that's how NSA was. Then they put the National Computer Security Center there as part of NSA, as an outward facing, open, public side that deals with, from the NSA's point of view, the unwashed, in the sense that these people can do bad things. Which maybe today we should have learned, you know with Snowden, maybe they were right. [Laughs.] But there was always this tension between NSA and NCSC because NCSC did things completely different.

Yost: I remember in interviewing Roger Schell, he related that even before it had formally been decided that that's where it would be, he was thinking that is the one place it couldn't or shouldn't be, that it shouldn't be at NSA. But that is where it ended up. And, of course, he went there early on to as second in command . . .

Kemmerer: He was assistant director, yes.

Yost: . . . central to the development of the Orange Book. Wasn't NBS discussed as a possibility, as well?

Kemmerer: I was not part of the discussion so I don't know, but NBS was clearly a player. I was on their advisory board for a number of years, too. I know I was there when they changed from NBS to NIST.

Yost: What was coming out in industry for commercial systems in the second half of the 1970s, for IBM systems, that got widespread use were access control packages RACF,

ACF2, and Top Secret. What did you think of those and did people in the computer security research community pay much attention to that?

Kemmerer: I know those systems. I know they were there. I never worked with any of them. I know that the only one I worked with at all was the one at SDC, which I can't remember the name. KVM, was it just KVM, which is an IBM product? But RACF and the others, I just knew about, because there weren't many of us that were university computer security researchers. When you found a problem, you sent a message out to 10 maybe 20 people. Dorothy Denning and Virgil Gligor, and later on, when Matt Bishop became a professor; those sort of people. A lot of that work was still held tightly. You didn't get the information on it. You'd talk about; well, at the Secure Database Summer Study in 1982, we had groups that we broke into, and those groups were supposed to decide where was the technology; you know, what was the state-of-the-art of technology going to be in secure databases. One group was, I think, a year out, another one was five years out, and the other was just sort of pie in the sky. I think I was in the median group with I think Carl Landwehr , if I'm not mistaken. But we got called to task because we came up with a security model for a secure database that wasn't Bell-LaPadula. And this was summer of 1982; in December 1982, they called our group back to NSA for a meeting — and this is actually a game changer for me, in some ways — they said why didn't you use the Bell-LaPadula model? I said we had a formal model for the database that made perfect sense. One of us, possibly me, said well why should we use Bell-LaPadula? They said because the theorems have been proved about the axioms. I'm a mathematician and that makes no sense, but that's exactly how they sold it. They called

them axioms, and the axioms were actually invariants that you were trying to prove. So proving theorems, that they're invariant in all the states, made sense. But they said the theorems are proved about the axioms and I thought, what the hell are you talking about? As a result of that meeting, Steve Lipner got a bunch of us the original Bell-LaPadula report, which Steve had signed off on; those weren't readily available before. People talked about Bell-LaPadula but they only knew it at the top level. They didn't know the details of the model. At that meeting, Carl Landwehr said he had a logician who just started working for him, and he was going to let him look at this Bell-LaPadula model to see what he thought. That logician was John McLean.

Yost: I just interviewed John.

Kemmerer: Great guy. That was John McLean. A result of that meeting was that there were a bunch of debates. I was the moderator, Dave Bell was one side, and John McLean was the other. He [McLean] came up with System Z. What came out of it was people didn't understand Bell-LaPadula. Everybody thinks they understand Bell-LaPadula, but they don't understand it all. And so it is sort of like when you say with the Anderson Report, and the Ware Report, we all know they're there but today you can get those easily; can probably get them from you guys, right? You couldn't get those then. You had to know somebody who was getting that for you, then you could sit down, look at it and see what you think, otherwise, it was all folklore. When you think about Bell-LaPadula; what I did was I took the Ina Jo language and I wrote formal specifications for Bell and LaPadula — I still have them around here somewhere — so I could understand it now

and see where things were wrong. I didn't find any big problems with it but I probably understood Bell-LaPadula better than Bell and LaPadula, at that time. So those were changing times, those debates went on for a long time. That was a big battle, it was a change. Finally we started to say, it doesn't have to be Bell-LaPadula. And just from that one meeting. Who knows what John McLean would be doing now otherwise? [Laughs.]

Yost: The Air Force program was the first to put really serious money into computer security. Was there a sense of kind of momentum and kind of a logic to, or progression to; okay, you've got the Anderson Report, then you've got Bell-LaPadula; and that's kind of viewed for some time as the state of the art with security models. Was the fact that probably more research money was put into this trajectory something that gave it a momentum and favored it over other potential models or trajectories.

Kemmerer: What were the other trajectories? You mean the Air Force put more into this than other Air Force trajectories? So what's your question? Do I think the Air Force was the main player, is that what you're saying, in computer security?

Yost: Was there a sense that Bell-LaPadula was almost kind of a sacred cow that couldn't be criticized because a good deal of resources had gone into it and projects started because of it, that it kind of had the legacy of the Anderson Report, to funding research at MITRE that tried to implement the secure kernel and reference monitor concept?

Kemmerer: I don't know. I don't have any strong feeling. There's nothing about that that drove me; in fact, I don't think I ever had an Air Force grant, I'm trying to remember. I knew the Air Force funded research and that MITRE was a big player in security.

Yost: The Air Force was giving contracts to SDC.

Kemmerer: Yes, like I said, COS-NFE was Air Force. Yes, they funded me and funded my research, but through SDC. So I'm thinking as a researcher, what did I do? It is sort of who you looked to. You looked to DoD. I mean, NSA funded me and I got a lot of private funding — not a lot, but I got some — I guess I never; it's interesting, you interviewing makes me realize how much the Air Force was involved in that, you know? That fact that it was the Air Force Summer Study. The fact that it was the Anderson Report. The fact that the Air Force was the funding for MITRE. It is, clearly; like how much of their funding comes from the Air Force? Not all of it, but certainly a large chunk, right? And maybe part of that is I don't know; how about SRI? Because in security I always think of SRI being a player, too.

Yost: Certainly, but probably a lower percentage than a place like SDC, and certainly less than MITRE. SRI I believe had a variety of funding sources for computer security—from the Air Force, Navy, NSA…

Kemmerer: I never thought of them [the Air Force] being like the Daddy Warbucks for the whole thing. But, you know, clearly they were; and if they hadn't done it, it probably would have been delayed much longer or maybe not happened, I don't know.

Yost: Can you tell me about your involvement for a secure ADA?

Kemmerer: For a secure ADA? Are you talking about the whole ADA thing?

Yost: Wasn't there a formal verification?

Kemmerer: The work I did. Oh, because before it was ADA, you know, it was the Iron Man and several other names, and I would comment on some of that work for people. But secure ADA; I built a symbolic execution system with one of my students here, which was not security related at all, but software engineering. We did a symbolic execution system for PASCAL. Basically what happened was there were a couple of symbolic execution systems, one done by somebody [Jim King] at IBM, and another one by Bill Howden at UC San Diego. There was none you could use, and I believed in symbolic execution as a way to learn about formal verification. You start with your entry assertion, you execute it, and you kind of see whether everything you know is strong enough to prove your exit assertion. I wanted to have tools. I like tools for my classroom; I like the students to have hands-on experience. So I designed a symbolic execution system for PASCAL, because PASCAL was the language that we were teaching in our introductory courses, and we used PASCAL to run on UNIX. And so I, with my students, built a

symbolic executor called UNISEX; and then we wanted to look at the problems when you're getting into multiple processors and multiple threads, and ADA was the obvious language. I had a student who worked on a symbolic executor for ADA. And basically, we looked into those hard parts. How do you deal with it when you're waiting, when you're synchronizing again? I don't remember the ADA terms. We didn't do that from a security sense. Is that in my resume or something?

Yost: No, just a question based on some preparatory research I ran across.

Kemmerer: I don't have a paper on secure ADA, do I? Every once in a while I find out about a paper I forgot about. But I did ADA symbolic execution, so what I did was formal methods for ADA. Formal methods for ADA could be used by people who were using ADA to go and write secure code, so only in that sense did I do that.

Yost: In 1988, you published a paper on critical gaps in formal verification technology. Can you tell a bit about the context of writing that paper and if you were optimistic or pessimistic?

Kemmerer: I was probably optimistic — maybe not as optimistic as — in 1988 it was?

Yost: Yes.

Kemmerer: That was Baltimore, Maryland paper, I think, maybe. Critical gaps in formal methods?

Yost: In formal verification.

Kemmerer: Okay, critical gaps in formal verification. I'm always optimistic, like I told you before, on formal verification. I think the critical gaps were we really weren't dealing with real-time, and I actually spent the next 10 years, from 1990 to 2000 or so, working on a real-time formal specification language and a real-time formal specification system, and then also how to refine those things. Very few people were doing anything with refinement. So there was real-time, there was refinement; we still had the problem that you needed a compiler; proof of a compiler where the compiler was using the same semantics that you used in the specification language, those sort of things. The real-time part was new theory that had to be developed, and like I said, I worked for 10 years on that and I think we did some really good, interesting stuff and hard stuff with the Astral system. That's probably one of the things I was thinking about, but the other gaps were down to code, and code to hardware, and doing those sort of things. But I don't remember where they were. I'd say I was positive, you know, and I might have even been given that topic. Was that at a workshop or was that at the National Computer Security Conference? I should pull up my resume, then I would know what I did when. [Laughs.] How'd you get my resume? Did I send you my resume?

Yost: No, I don't think so. But I found one online.

Kemmerer: Really.

Yost: And then also that

Kemmerer: Oh, right. Okay, there's also 2005.

Yost: I forgot where I found it. It wasn't linked to your faculty page.

Kemmerer: I know, I don't have one. My view is that if people who need to know who I am, I'm not interested in talking to them. I made an exception for you because Becky Bace said that I needed to talk to this guy. Okay, this isn't the same as doing a search. The reason I asked is because I'll be invited for a keynote or something, people say I'm trying to find your CV and I can't find it anywhere; and then they'll get like an eight-line summary that the College of Engineering made me write for something. But yes, I'm sure I was positive on it.

Yost: Speaking of Becky, you briefly mentioned your work on intrusion detection. Did that at all seem a surprising direction for you to do research, given that your work was primarily on formal methods and developing secure systems to keep people out rather than tracking people once they're in?

Kemmerer: Yes, these things are all timing and people. It is interesting the people that make me go in a different direction. I had a student named Nina Lewis, who was pursuing her Ph.D. at the time, and she was interested in risk analysis for security. Her father was actually a professor on campus here in physics, who did a lot of work on risk analysis of nuclear reactors. In fact, he went over to Chernobyl. He was one of the guys that went over there like days after the meltdown happened, to go to try to figure out what was going on. So she was interested in risk analysis for computer security. We worked on that for a while, and it turned out that risk analysis; if you're doing a nuclear reactor, is okay, I've got this valve here and we know historically that those valves fail one in however many times; and I've got something else here and I know its history. Well, computer security is absolutely the opposite, you can't depend on what it was. First off, when I'm the hacker, I do what you least expect and you can't depend on it, so you don't have this history. So then we said, okay, we're looking more at this risk analysis. Oh, they have this thing called Bayesian statistics, you know, so you sort of set some initial values and through the Bayesian inference it all gets changed to the proper values; well, it still doesn't work for computer security. So finally, I think she and I together decided this isn't a useful thing to do. But I think either she or both of us, I think both of us at one time, went to a risk analysis workshop at SRI and I think it was maybe combined with the intrusion detection people. Intrusion detection is the other side of that coin. And also, I thought this is a more fun problem to work on; this is something I can wrap my head around. There's a guy, Phil Porras, at SRI; works with Peter [Neumann] and other folks up there. He was doing a master's thesis with me, and we looked at signature-based intrusion detection. We said if we had a way, instead of having a separate rule for every

case, if we could abstract out. At that point in time, which was around 1990 or something like that, signature-based approaches would say okay, we've seen this attack and there was this audit record, followed by this audit record, and they would throw away audit records in between that didn't matter. But they would look for that exact same thing. So what we said was we should be able to abstract out from that. So let's say you do something that allows reading or whatever you're doing. And so we abstract it out, we defined the critical things about the state and the whole, and when we saw certain kind of transition to a state that satisfied these other constraints, then we could do it. So basically we had, at one point in time buffer overflows, instead of having hundreds of rules, which is what Symantec and other places were running, we had three rules that covered it. And so that was the start of our intrusion detection work. So Phil Porras and I did the original work on that. And then there was another master's student named Koral Ilgun, who actually developed the first system for Sun OS 4.3. Then I became the department chair in 1993, and I was also doing the real time formal methods stuff. Between those two, I didn't do any intrusion detection. Maybe I did something; I think we did a system for Solaris, which was the next update. But then Giovanni Vigna came to work with me and what we did was a network-based intrusion detection system; and then we did an *ad hoc* networks system. But then what we did was we said, let's build an approach, because let's look at what there is for intrusion detection. We have a source of events and something that processes those events. The event processor is looking for conditions to hold, you know, still in the state transition part. It's looking for states and transitions to match up there, where we are. And then there's a response. So basically what we built was the bare modules where you then could define here's the kind of events I'm going to

look for; here's the signatures I'm going to look for; here's the kinds of responses of what I could do; and then we're going to customize it. So we could customize one for the network, one for Apache web application, another one for sys logs, and so that's what we did, developing a family of those. So all you had to do is have this bare bones module located on a host, and everything else you could download onto them. Then we went one step forward where, when they got the alerts, they passed those up to either one or more controllers or enterprise security officers, whatever, and they tried to get the big picture out of this. And so that was kind of that intrusion detection. From that we then looked at the next step, which is if we have all these alerts, how can we correlate them to get the big picture? So we started looking at correlation, and then keeping track of assets. And then from that, we started looking at well, who wants these? If you have an enterprise here, you have certain missions that need to be accomplished that depend on these assets, then we could do cyber situation awareness. And that's kind of what we're working on now. So it all came out of that intrusion detection work. We did mostly signature-based. Like I say, we did the original work in the early 1990s, didn't do anything for about four years, and then in the late 1990s we developed more and did this family approach. Then we went into the correlation work and cyber situation awareness, but we also at the same time started looking at anomaly detection. We did some new approaches to anomaly detection, so we kind of hit all over that area.

Yost: Did you carefully study IDES and did it provide a sort of foundation?

Kemmerer: By IDES, do you mean the system at . . .

Yost: SRI.

Kemmerer: No. I mean, we looked at what they were doing but their system had; well, I don't remember which one they called IDES, because the interesting thing is Phil Porras wanted to name our system, instead of the STAT system, wanted to name it IDES. I said no, IDES is just intrusion detection system; why would you name it IDES? And then he went to work for Aerospace, and then went to SRI, and in the meantime, they had named theirs IDES. Now doesn't the IDES system do more statistical based analysis? Because I think that was the early one that Al Valdes developed, and that's more statistical based. That's more anomaly detection. They have another system that Phil did with Ulf [Lindqvist], a guy from Sweden whose last name I can't remember, and that's more of a signature-based type system. We are similar, but we were doing the work at the same time. I don't know. Obviously, Phil took the knowledge of what he had done down here and so he used that. You should build on the shoulders of others, not stand on their feet.

Yost: In 1990 you published a specification on mental health care database. Were there unique attributes to this domain, versus other areas that you'd worked on for a secure system?

Kemmerer: There are unique aspects; it got more into privacy. What's interesting about it is, remember I told you if you can understand a system and tell me about it I can write a formal spec for it? That's what happened here. I don't remember whether I was doing the

National Academy of Science study, where I looked at healthcare databases first, but somehow, I was on the National Academy of Science committee that looked at healthcare and security. And part of it was how secure are the databases? Things were pretty poor. This was a nice study, and Carl Landwehr was part of it. But then there was a database workshop and for that workshop, I wrote the specification that you are referring to. It's limited to healthcare, because they had different problems, in my mind. But in some ways, it is similar. It is like everything else; what are the invariants that you want to hold everywhere; or what are the things across states that you want to hold? Like if you don't have access now and you get access, it better be the case that you satisfy certain conditions. And this is the whole Bell-LaPadula and McLean argument; there's this transition property, too, that has to hold. Those are all the same, you're just applying it to a different system. So when I applied it to a healthcare database, number one, I had built up a lot of knowledge because I was on this National Academy of Science committee so I knew these things. I knew that if there were certain diseases, you have to report them. I also knew all the privacy issues about what you couldn't do. So what I did was I tried to capture those in my invariants. These are the critical things I have to satisfy, and what are the different operations that you would do on a database, and then how do you deal with those? So it's a domain area but you do the same kind of things with formal methods. That doesn't mean it's easy, because you have to understand the domain area. So another place that happened was with the crypto stuff. In 1981, there was a guy, Allen Gersho, who was an ECE professor. Before that, in the summer of 1980, my first Ph.D. student — I remember his last name, [Brent] Auernheimer is his last name— anyway, he said he wanted something to do for the summer and I said I know all about computer security and

network security, but I don't know much about cryptography so what I would like you to do is build up an annotated index of the crypto literature. And he did that. So part of what he did was there was this professor named Allen Gersho in ECE, who did some crypto stuff, and he had talked to him. Allen came out of Bell Labs. Anyway, it turns out that Allen Gersho and another guy decided to have this conference called Cryptography Conference 1981, on campus here, and so in the winter they came to me and they said, Dick, we are thinking about putting on this conference, and we need somebody to do local arrangements so would you be part of the founders to go do this? So basically, we did the Crypto Conference. Those two guys aren't involved in it at all anymore, but it's gone on every year and it's the top crypto conference. The reason I'm telling you this is I was sitting, watching one of the presentations of crypto protocols, and I'm looking at it. And I'm sitting there going, well what do you have here? We have a five- or six-step protocol. When you go from step here, well you know something about; so I'm thinking about it in the way I write formal specifications, and each of those steps is a change, a state change, and then there are certain properties, you can't just do those steps anywhere.

[INTERRUPTION]

Anyway, so I looked at it that way. So what I did was I started looking for some protocols and tried writing specifications for them. One of them was the oblivious transfer protocol that Manny Blum at Berkeley had done; in fact, that's the one that caught my attention. And when I wrote it, well, it depended a lot on number theory, and to write the specs for all the assumptions I had pages and pages of spec; so that didn't work. And then finally I

did something for an IBM protocol that they had for a system that was released out in the field, and they realized later that they needed two master keys instead of one and had to recall the whole system. So I wrote a paper on that. The point is that I got into the analysis of encryption protocols just because of having started with the crypto conference, because Allen asked me to, because a student wanted something to do, and then Allen asked me to go do local arrangements. I forgot something that I said with regard to the new computer security people here. In 1982, we hired a guy named Alan Konheim, who had actually worked on the DES standard. He had worked on Lucifer. He was head of the math department at IBM when they did the Lucifer development. And so he was a crypto guy, but I don't consider him a computer security guy. He was here from 1982 until about five years ago. But anyway, so from the analysis of protocols, a lot of research developed out of that.

[BREAK]

Yost: I see that in the mid-1980s — 1985-86 academic year — you were the visiting scientist at the laboratory for Computer Science at M.I.T. Can you talk about your research that year and were you collaborating with people? Jerry Saltzer or others?

Kemmerer: One of the things that happened through the crypto meeting was I got to meet all the top people in cryptography, and one of those was Ron Rivest. I had thought about going to M.I.T. for sabbatical, and John Guttag, who did formal method and who used to be at ISI, and I had talked about me doing a sabbatical. He said that would be

great; work on the formal methods research. It turned out that he had been asking for a sabbatical at M.I.T. for years, you know, because they don't just automatically give them. And they gave it to him that year. So at the crypto conference, I asked Ron if I could go with his group and he said sure. So I was in that group, which is actually the theory group. I looked at some interesting things. There was a working group there that Albert Myer, I think it was, was running, and we would meet once a week. I worked on some different things that Robin Milner had done, a different kind of formal specification work. I did various things. I grew there in terms of actually writing papers, I actually wrote my first analysis of encryption protocols paper there, even though it was something I had started on years before, but I wrote that while I was there. I was finishing up — oh, an interesting thing. So Roger Schell and I flew on an airplane back from Baltimore, Maryland to Los Angeles one time together, sitting across the aisle from each other. And I had had this idea — this would've been in 1985, 1984; probably 1983, 1984 — and I had had this idea that what we should do is we should have the people who have real formal verification systems, real ones, where they had the iron running, and have written specs for real things; we should have those people all get together. Actually, let me go back even further. In 1982, just before Marv Schaefer left to go to NCSC, SDC was bidding on a program which was called Secure Formal KSOS, but it was verified KSOS, and NSA was the sponsor of it. And NSA wanted to use a formal verification language, and I think they were favoring the Gypsy language from U-Texas. And so even though we had our own language, the back word was that they really wanted somebody to do it in Gypsy. So we flew down to Texas, to visit with Don Good for three or four days, to find out about his system and while we were on the plane, somebody from NSA sent a

message down to Don Good that — oh, they had written specs for a red to green system, or something, or a black to red system, whatever — we wanted to see those specs so that we could understand the language. While we were on the plane going down there, someone at NSA said they [Don Good's group] couldn't show them to us, even though they had shown them to other people who were bidding on this contract. So we were down there for two or three days, basically not able to do what we wanted. So what we decided was to take an example. Carl Landwehr and some other people had published a paper where they had written some examples in different specification languages and every one of us that looked at it — and one of them was our language — said that's not the way you should do it. So we said let's take a sample problem; let's define a problem, let's write the specification, let's break our group up into two people or so from SDC and two people or so from Gypsy, and one group is going to write the spec in Ina Jo, show how we'd write it and explain why to the Gypsy people; and vice versa for the other group. And then let's come back and talk about it. It was a very productive weekend. When we were leaving we said, you know, we ought to do this in a real way with a number of problems. We all agreed, yes, good idea. Marv moves back to NCSC and I said hey, we ought to do that. He said he presented it and they didn't want to do it that way. Anyway, I'm coming back on an airplane with Roger Schell and we're talking; he's sucking my brain on different things. I said I don't understand why you guys don't want to fund a formal verification systems comparison project. He goes, what do you mean we don't want to? I said, well Marv said you guys didn't want to. He said tell me about it. So I told him about it. By the time I got off the plane, he said I think we should do that. Basically, he funded us. And we looked at Affirm, which is out of ISI, which is algebraic

spec; we looked at Gypsy; we looked at SRI Special; and we looked at Ina Jo. We had the developers of each of the systems doing it. And then it was decided that they wanted me to be the sort of all-around guy and then they got some other people from SDC on it. So we would go for a week to each site. Well, first of all, on election night 1984, we were meeting back East to decide on a set of five problems that we were going to write specifications for. Then we would go for a week to each place, like I would go down to Texas, Don and his people would tell us about Gypsy, okay? And then they would have written their specs for these five different problems. Explaining why they wrote them that way, and we would ask questions. I, in the meantime, had come up with my own sample problem that I wrote in each of the languages and it was a great experience. Then we each wrote our opinions on this. When NSA sent me that grant, they sent the grant papers out here and they had the typical thing that said you can't tell people who's funding you and you can't write any papers without getting prior approval. The people over at our administration said we are a publicly funded university, we can't agree to these two. Okay? So they [NSA] removed them. So I could publish anything and I didn't have to get pre-approval, and I could tell who's funding me. So anyway, we did this thing for most of 1985, going around to the four places. We wrote up the reports. I had all the reports and basically, it was five volumes, it's a stack like this, referred to as the Kemmerer Report. Meanwhile, I left to go to M.I.T. I had my originals and I had money to pay for duplicating them and sending them out, but meanwhile, NSA contacted me and said you know, if you want, we can duplicate the reports and send them out for you. Just send us the originals and send us the mailing addresses. I had a list of about 150 addresses; half were U.S. and the rest were foreign nationals. And so fine, saves me a lot of time, right? I

know I left M.I.T. in May and went to Italy for three months. While I'm over there, Marv sends me an e-mail and says you know the Kemmerer Report is arms embargoed? So they copied it for me, but they didn't mail it to any of the foreign addresses. They didn't have any reason for the arms embargo, that was for crypto stuff, primarily. There was nothing that wasn't in the public literature. And so none of the foreign addresses got any, plus I was sitting there [Italy] with a copy with me because I didn't know that. So anyway, during that period I did a lot of work on that, and I wrote a paper about what we did. A short paper just summarizing it that I gave at NBS the following fall. But, yes, another tangent, sorry. I worked for years to get that changed, by the way, and they never did; they always kept it arms embargoed.

Yost: In 1987, you were a member of NCSC's Computer Security Curriculum Workshop. Can you talk about developing curriculum in computer security, and how that evolved, and what was achieved by that workshop?

Kemmerer: Dorothy Denning was heading that up, and Chuck Pfleeger was there, and a database guy from Naval Postgraduate School, and somebody else. But basically what we did was looked at each of the undergraduate classes, like data structures, and said how can we put an element of computer security in it? Like if you're looking at something like a page swapping algorithm, or something, how could you say where are some of the dangers in this, of knowing what's in there already. And not even at the covert channel level, but just leftover data that's in there, if you didn't erase the pages and things like that. And so for each of the areas, we defined what would maybe be a class, or a half to

two classes of topics on computer security that would tie into those things. And then the idea was to — and it was published by the NCSC — but the idea was to give those to people preparing courses so they could use them. I don't know that it had a large circulation. And after that, I think Matt Bishop did some similar work in that area; and Cynthia [Irvine] from Naval Postgraduate School, I know, was active; her and Matt. But I think that when I heard about people doing that, I always sent them copies of this to go and say hey, you might want to use this as part of what you're doing.

Yost: In 1990s, you're on the program committee of the European Symposium on Research in Computer Security. Were there fundamental differences in the way European computer security researchers approached problems and solution paths with computer security?

Kemmerer: None that I can think of. If you're talking towards the language thing, ADA was more popular in Europe than here, so if it was something where you bring in the languages that would be the case. Otherwise, no, not in 1990. In the early days, all of computer research, I think, at least most of it, they were always more theoretical and less building systems. Like if I did something with the covert channel analysis, can I now build something to help you do that? And they were primarily interested in the theory. That's changed since then.

Yost: With the Orange Book, I understand that there was a bit of a debate, to have the levels as they set them versus what was referred to as a Chinese menu. Can you talk about those alternatives and do you think the right choice was made?

Kemmerer: In making the Orange Book?

Yost: Yes.

Kemmerer: Making the Orange Book was the right choice at the time because it was a choice. Somebody's got to do something first, you know, and they did it first. I used to teach about all the different alternatives, and I have them over here on my shelf, but the Green Book, or whatever it was. They did this, then they came out with, like you say, the Chinese menu where you could do this part at this level, or this part at this level. And I believe — I haven't been following that — but I believe that now they've come up with another thing where you could just do components or something like that and not a whole system, in the U.S. But I'm not sure. I don't think they made a wrong choice, because I don't think they were choosing at the time and I think that what they did was they got that ball rolling. And then the final one that there was, it sort of took what was done in the three main approaches, which I don't remember who that was anymore, and sort of combined them. I think that's where you say you have a choice in the different ways of doing it, which probably makes sense.

Yost: You became an EIC of IEEE *Transactions in Software Engineering*. Tell me a bit about that role and what vision you have in taking that on journal, and in what years you served in that role.

Kemmerer: What vision I had?

Yost: Yes. And what years — it looks like it isn't listed on what I have..

Kemmerer: What years I did that?

Yost: Yes.

Kemmerer: Let me get my resume up here. I can't tell you without my resume. Probably figure it out. It's right here. I think I ended in 2000, and it's for four years. I think it was 1997 or 6 to 2000. I could tell you later. I can give you an up-to-date resume.
So my reason for taking it over was primarily just doing service for the community, but it was also the delay in getting things published was so long. It was probably two years plus from when somebody submitted a journal article and it got reviewed, then they'd say it needed some improvements, and it'd take about two years to do that. I was hoping that we could speed that process up and I think we did speed it up some. For Software Engineering, in general, a lot of researchers tend to take their papers, they think I'll submit this to *Transactions on Software Engineering* when it really should go to *Transactions on Databases*, or it should go to *TPAMI*, which is on graphics. They think

Software Engineering, that's everything. Software Engineering does cover a lot, and so I guess one part of my motivation was to make sure we just take software engineering, we don't take a paper that somebody should have submitted to *Transactions on Databases*, and maybe they did and it didn't get accepted and they think well, maybe the *Transactions on Software Engineering* people will take it. I wanted to make it a quality journal that you would want to read every month.

Yost: Looking back at the history of computer security, and in thinking about researchers coming into the field today, are there important lessons from the past, from the 1970s and 1980s, that you feel the next generation of computer security researchers aren't learning? And if so, what are those lessons?

Kemmerer: I think there's just a general lesson that if you're getting into any field, you should read the seminal papers that were done back in the 1970s, or whenever. I mean, you'll see people writing something and you go yes, we discussed that. And there are some colleagues of mine who actually say that, I think, more times than they should because it wasn't all solved in the 1970s so we have to learn something, too. But right now, something interesting in our group is looking at Android malware. Well, it's a different platform, sort of like when we went to microcomputers and things like that; things that you did in operating systems prior to that year, so learn what we did there and see how you could use it somewhere else. The computer security people did that with their approaches to trying to penetrate systems. What worked before? I'm not going to do exactly the same thing, it's a different system but here's what worked before. Is there a

problem with something spread across two pages? Can I gain some information from that? I'm amazed all the time where you go and read about some new problem and you say well yes, that's the basic problem — time of read, time of use. Popek and Bisby wrote a paper on that I think in 1973, and we still have that. A guy goes and he says I want to open file X with read access, and the system goes and copies X, and copies read, and goes and does an access check and says yes, he's allowed access to X. And in the meantime, he changes X to Y, and now he's got access to Y. So copy the parameters over. We were working on some security testing with a bank where that's exactly how we got in. They asked us a question and we answered another question by telling them what question they asked us because we only knew the birthday, we didn't know the social security number. Those are lessons, you know, there are basic papers out there maybe you should read. I think that's the main thing.

Yost: Before we conclude, are there any topics I haven't brought up or areas we haven't talked about that you'd like to discuss?

Kemmerer: Just if you want — you're aware of the work we did on electric voting machines we analyzed as part of a study that Matt Bishop and Dave Wagner did for the Secretary of State. We were the red team for the Sequoia system; showed how we could completely compromise it, and as a result, it was decertified. And then there was another study called Everett, which was for the Secretary of State of Ohio. In that case, we were looking at the ES&S system, and again we fully compromised that system. I don't know whether they decertified it or not. I'm not sure what they did with it because that was in

Ohio. But those are high impact. For me, as a researcher, high and medium impact was great. That was 2009. In 2011, we took over the Torpig botnet and we had 180,000 bots reporting to us every 20 minutes, giving us stolen bank credentials and information from the systems they were on. Again, high impact. We owned that for three weeks — well, for 10 days; just fun stuff. Computer security is a fun area. It's hard to believe people pay you to work in it. [Laughs.] You should pay them.

Yost: It's fascinating to do these interviews. Thank you so much for your time.

Kemmerer: If you need anything else you can let me know. If you want an updated copy of my resume or my CV . . .