

An Interview with
W. EARL BOEBERT

OH 460

Conducted by Jeffrey R. Yost

on

28 April 2015

Computer Security History Project

Albuquerque, New Mexico

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

W. Earl Boebert Interview

28 April 2015

Oral History 460

Abstract

Computer security pioneer Earl Boebert discusses his education at Stanford University before the bulk of the interview focuses on his work within the Air Force and at Honeywell. Among the topics he discusses are the Air Force Undergraduate Navigator Training System, efforts to save and market Multics (and the inherent challenges given GE's existing systems and the economics of the mainframe business), PSOS, Sidewinder, the formation of Secure Computing Corporation. Also discussed is his role in the broader computer security research community including serving on many National Research Council committees, including the one producing the influential 1991 *Computers at Risk*.

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 this afternoon on April 28, 2015 with Earl Boebert in Albuquerque, New Mexico. This is for CBI's NSF-funded study "Building an Infrastructure for Computer Security History." Earl, can you begin by just giving me some basic biographical information, where and when you were born?

Boebert: I was born in Elko, Nevada in 1939. My father was a railroad policeman. I'm an only child.

Yost: Did you grow up there as well?

Boebert: Until I was nine years old. And then my father was promoted and we moved to the San Francisco bay area. I grew up there, in Albany, next door to Berkeley.

Yost: Were there particular areas of interest you had in school, pre-college?

Boebert: Just about anything mechanical; model airplanes, model cars, cars, you know. Any of that stuff.

Yost: And who were your greatest influences in your early years, so pre-college?

Boebert: Well, let's see. My father and mother, and some of my peer group; it depends on the age, of course, you're talking about. A lot of the teachers. My high school was

basically a manual arts school so I learned — probably the single biggest influence in high school was the mechanical drawing, the drafting teacher.

Yost: Can you tell me about the process you went through in deciding where to go to college?

Boebert: [Chuckle] It's kinda funny. I was recruited by both Berkeley and Stanford, and decided mostly on Stanford because my mother was a graduate of the San Jose State Teachers College, and San Jose Teachers College women thought Stanford men were the cat's meow. [Laughs.] And so she was ecstatic that I got in.

Yost: When you started did you have an idea of what you wanted to major in?

Boebert: Yes, mechanical engineering. I wanted to go off to El Segundo and design landing gears for Convair.

Yost: What was your first introduction to computing at Stanford?

Boebert: I needed a job, and on the job postings there was busboy and night computer operator. I didn't know what a night computer operator did but I sure as hell knew what a busboy did. [Laughs.] So I went to what was then — it was 1958 — it was then known as the Stanford Computation Center. A guy there, Al "Jazzbo" Collins was the chief machine operator. We called him Jazzbo because there was a New York disc jockey

called Al Collins. And he was an ex-street kid and I was an ex-street kid and we sort of hit it off, so he hired me.

Yost: What types of things did you work on there?

Boebert: First thing was a computer operator. It was an IBM 650, a 533 card reader punch, 402 printer. They sold time on it to various commercial outfits. The biggest user was a civil engineering firm that ran a package called COGO, coordinate geometry, which calculated areas of lots and stuff like that. So it was a question of you got the program deck, the customer dropped the deck of data off; and you put it on the back of the program deck; you set up the console according to the instructions; you press start; read the program; read the cards; it ground away; punched a bunch of cards; took the cards out; put them in the 402; 402 listed them; put the whole thing together in a nice package and put it in the out box.

Yost: And so during the day, part of the time, the computer was used for faculty and students, but the remainder of the time they're selling computer time.

Boebert: Yes, at night they sold time on it. And they did that also when they upgraded to the Burroughs 220, and I was night operator for the Burroughs 220 except for the nights when the bank ran it.

Yost: From one of your web pages I saw that you automated half time card stunts for Stanford football games. Can you tell me about that?

Boebert: Yes. That was 1961, I think. I'm terrible about the dates but anyway, Larry Tessler wrote that up and he put us in for pioneers of computer graphics, Larry Breed and I. And SIGGRAPH [ACM Special Interest Group on GRAPHics and Interactive Techniques] got all huffy and said that human pixels didn't count, which was really too bad. Larry did all the really creative work, the mathematics, and he did some of the first, if not *the* first, raster scan fill algorithms and things like that. I did the I/O. I ran two IBM 407s full speed for about six hours printing the tickets. There was a data compression algorithm that I was very proud of. Basically, it discarded duplicate instructions. So I learned a lot from [that]. It was interesting. I learned a great deal about human interfaces from that because the users; I learned that if you give a user an interface they're gonna exploit it. And so the way the thing worked was you had the card section, which was a little — I forget the dimensions — but say 300 seats x 300 seats square, right? Well, of course, since you were on the computer, Larry had an arbitrarily large — I don't know — 1,000 x 2,000 seat area, of which this was just a window. So what the people devising the stunts would discover on their own was that they could park pieces of card stunts off in the unseen area. And one of the commands was an instantaneous transpose and then when they wanted it to appear, they'd transpose it down into the window where it showed up. When they wanted it to disappear they'd transpose it back out into the unseen area. And that was an interesting education to watch that.

Yost: Can you tell me about your decision to enter the Air Force upon graduating?

Boebert: I was essentially talent-spotted by a group that included the head of the Stanford University Press; the editor of the Stanford University Press, a guy by the name of Jess Bell, and he'd been a crypt analyst in the Korean War, breaking Chinese codes. And there were several Air Force officers attending Stanford on an Air Force program called AFIT, Air Force Institute of Technology, which was a thing where Air Force officers went and got their graduate degrees. And of course, this was the height of the Cold War and the Air Force needed computer people, and so I was; I mean, I was a terrible student. It took me five years to get out.

Yost: And did you graduate in mechanical engineering?

Boebert: No, math; I transferred to math. [laughs] Some guy told me years ago — I bumped into somebody and I can't remember who it was — I got out by means of what I always refer to as the term "escaped"; I escaped Stanford by an undergraduate honors program in history and a math lab, writing a compiler. And supposedly the story is that when George Forsythe went up before the academic senate to argue for a computer science department at Stanford, he used me as an example and he said that if anybody like Boebert can get a degree in mathematics [laughing heartily] there's something wrong with your setup. So they decided that the computer geeks could have their own playground in the university. I don't know whether that's true or not but he was a great guy, by the way, Forsythe. He recruited the LISP AI guy — he just passed away recently;

from M.I.T. — McCarthy — he recruited McCarthy. Supposedly, Forsythe went to McCarthy and said in typical Stanford fashion, we'll give you a ton of money. Then McCarthy said I like it at M.I.T. And they went back and in another typical Stanford fashion, they said well we'll give you a house on campus; and McCarthy said that's okay, I like where I live in Cambridge. And Forsythe went back to McCarthy and said we'll give you your own personal IBM 7090, and McCarthy said, "Sold." So Larry Breed, Roger Moore, and I staged that machine at IBM research in San Jose, before they moved it on campus. That was one of the last things I did before I went into the Air Force; as machine operators, ran diagnostics and stuff.

Yost: You entered the Air Force as an electronic data processor specialist, is that right?

Boebert: Yes.

Yost: Can you talk about the early assignments you had?

Boebert: Well, it was Curtis LeMay's Air Force, which was a pretty hard ass organization, is the best way to describe it. Air Training Command made the initial assignments, upon commissioning, and ADP officers were in such high demand that Air Training Command assigned me to Air Training Command, so I basically went across town from Lackland Air Force Base up to Randolph Air Force Base, to the headquarters. As a second lieutenant I was in a major's slot. As I recall, there were about 200 slots for ADP officers in the Air Force, and there were 25 of us, all junior, and we automated the

whole damned outfit, basically converted it from punch card to computer equipment, or first generation computer equipment to computer equipment. The big advantage I had was that Air Training Command had a Burroughs 220, which I had programmed for several years at Stanford, so I just went right in. Had a shop of about, I guess, about a dozen — at the end, up to 30 people — and they managed the conversion of a whole bunch of assembly code, Burroughs 220 programs, to COBOL, when they brought the Honeywell 800s into the Air Training Command. ATC was, in those days, was a big operation. It was as big as IBM worldwide so it was like being at headquarters IBM in terms of the scope of what you did.

Yost: What was the gender mix? Was it almost all male?

Boebert: Oh, yeah. I mean, this was just before the Cuban missile crisis. There was one; I mean, other than nurses, there was one female officer at the headquarters. Yes, I tell people I don't need to watch *Mad Men*, I was there.

Yost: At some of the smaller companies there was a mix of women programmers early on, like at the start-up computer services firm Computer Usage —

Boebert: Oh, at Honeywell there was a very high mix of women programmers and I recruited and hired them by choice because they'd had to work harder to get where they were.

Yost: What were the earliest discussions that you recall in the Air Force about computer security, that you were privy to?

Boebert: When I hit the place, I mean, Air Training Command, the two number one Soviet Intelligence targets in the United States Military were Fort Huachuca and Air Training Command. Well, I guess there was also the Naval Electronics place in San Diego. So that'd make three. And that's because in Air Training Command, everything was there, right? I mean, there was a manual for every single piece of equipment the Air Force had because we trained all the techs, so it was a big concern. So that started it, I guess. Most of that was physical security stuff and later, let's see, first — I gotta figure out what I can say here [laughs] — first exchange with NSA was probably 1965, about when the Honeywell machines came in. We had a crypto room and had to do stuff.

Yost: And so with physical security, this wasn't a time-sharing environment?

Boebert: No, these were all batch processing machines so there were questions of; most of it was TEMPEST and also like I said, we had crypto links that had to be managed.

Yost: Was the risk with electronic emanations a serious risk?

Boebert: Yes, that was the concern, right.

Yost: What led you to leave the Air Force and join Honeywell in 1966?

Boebert: One of my mentors in the Air Force was a World War II veteran — no, he was a Korean War veteran — I had several mentors who were World War II veterans, which was a pretty amazing experience when I look back on it. They were pretty amazing guys. We were just ramping up — see I was in the Air Force for the Cuban Missile Crisis, the Kennedy assassination, and the Tonkin Gulf incident — so it was a transitional era, I guess. After the Tonkin Gulf incident he went out to set up a TACAN installation. TACAN was the air traffic control for military aircraft is the easiest way to describe it. And so he came back. I was on a pretty good career track in the Air Force. He came back — of course he fought in Asia — he came back and he said lieutenant, get out, you'll throw your life away for nothing. You'll serve your country better on the outside than ending up dead in Southeast Asia, and he was right.

Yost: And were there a number of different companies you considered before going to Honeywell?

Boebert: Oh yes.

Yost: Can you talk about that process?

Boebert: I interviewed with the whole cycle, but we were a Honeywell installation by that time because the major air commands had Honeywell 800 computers and in the same building, actually right next door in the computer room, was a Burroughs B5000 that ran

the military personnel center. So I went to interview with Burroughs, interviewed with UNIVAC, interviewed with RCA, interviewed with Honeywell. The obvious one was Honeywell.

Yost: So did the Air Force have any IBM systems or did they just have the Honeywell and the Burroughs?

Boebert: There were IBM machines at the Pentagon but IBM did not win any of the Air Force wide automation contracts.

Yost: When you were in the Air Force did you have much interaction with SDC?

Boebert: No.

Yost: Or the Rand Corporation?

Boebert: No.

Yost: What was your initial position when you were hired at Honeywell?

Boebert: I was a first level supervisor; a group manager in an advanced development group for what was then the Honeywell EDP Division in beautiful downtown Wellesley Hills, Massachusetts.

Yost: Can you describe that early work?

Boebert: We were designing a successor machine to the Honeywell 200. I had a whole slew of assignments. I decided in retrospect that one of the problems in being a quick study is people pile on stuff so I never lacked for something to do. We had the team that was designing the advanced machine and there were basically three of us — a few more — who were the architects for the thing, and I designed the programming language for it. It was a ILIF machine, tagged architecture. So the operands carried tags and the instructions were all operand-neutral. And then the logic examined the tags of the operands so ADD was scale; fixed point ADD if it was a scale; fixed point operand; and it was floating point ADD if it was a floating point operand; and conversion rules; and all that stuff. It would've been an interesting machine if they'd have built it, but they didn't.

Yost: How large was the group you were supervising?

Boebert: Ten, eight, something like that.

Yost: And in the 1966 period to the end of the decade, before the acquisition of GE's computer division, were there discussions about computer security at Honeywell, and if so, can you talk about those?

Boebert: Oh, yes. Mostly tied in with classified work, so it's not much I can say. But yes, there was activity.

Yost: Can you tell me about the undergraduate navigator training system?

Boebert: Boy, what a monster that thing was!

Yost: You were one of three lead designers?

Boebert: Yes. Well, basically, it simulated the navigator system on a transport aircraft. It was so big that the Air Force built a building to house it in, at Mather Air Force Base, with the disk drives in the basement. The requirement was that you should be able to calculate the radar equation; calculate and display a simulated radar return for an aircraft flying at Mach 2 across the United States. The United States was digitized — oh, I used to remember the numbers — well, it was a gigabyte of storage, which in the late 1960s, was like almost all; like doubling the amount of storage in the world, alright? I mean, that's an exaggeration, but it was huge. Okay, it was 12 IBM 2314 disk drives. So you carried reflectants, and you carried altitude, and then you had to carry what they called cultural, which was buildings. And you had to have the ability to generate weather. The

problem with weather is you can't simulate a thunderstorm with a 20,000-foot tall building because the radar can look under it, so you had to have it. So there was a ton of this. So the thing was built out of the marine system's division in West Covina, California. And they were the simulator; they did simulation and training stuff. By that time I had transferred to a corporate-wide advanced development group [at] headquarters in Minneapolis in December of 1969. One of my first assignments was to analyze their system for distributing the data on the disk. I analyzed it and wrote a nice memo that said this sucks, this will not work. They didn't understand how a moving head disk worked and so they were taking average access times and they were trying to design this thing on average access time instead of worst-case, which is you've got the arm in the middle and the thing you want is on the outside, okay? So they cogitated for a while; then they sent another one up to Minneapolis and I wrote another memo saying this thing sucks, it won't work. I think this cycle happened like three times and then they called my boss, who is a guy by the name of Jim Renier, who ended up as CEO of Honeywell later on. And they said, can you send this smartass down here to tell us how to make this thing work? [Laughs.] So as we used to say in Honeywell jargon in the day, I got mailed to West Covina. They're a great bunch of guys, and we sat down and devised — they refined it but I did the first one — and basically the idea was, you have an airplane sitting on a square, and you had all the adjacent squares; and what you want to do is make sure that you distribute the database in such a fashion that no matter which square the airplane hops onto, that square is under the head of a disk arm on the disk. And you gotta do this for 55 different airplanes that are flying through this database. So it was a humongous problem and it was an exercise; there wasn't anything particularly theoretical or analytic

about it. It was just kind of *ad hoc* engineering but we got that worked [out] and we demonstrated it.

Yost: Was there significant learning from what was achieved with SAGE?

Boebert: No, we did it all; we spun it up on our own. Nobody knew anything about any of this stuff. The Air Force, or the government, was just building the digitized maps of the United States. That was a wonderful exercise. So we get these reels of tape down to West Covina and the way they do the map is they'd get a twin engine Army airplane with a side-looking radar on it, and you'd fly patterns back and forth, right? And then he'd land and then night would fall, day would come, different temperature, slightly different airplane, slightly different pattern, and he would fly back and forth. Now when you pasted these things together you got discontinuities. Now if you know anything about radar, you think about the radar return from a three-foot-high shelf running all the way across the state of Iowa, right? [Laughs.] So we brought this thing up and the display was on what's called a PPI, Plan Program Indicator, the old traditional circular, you know, World War II kind of display. And they'd bring this thing up and there'd be like this gash running right across the radar return. So Honeywell became the world's leading expert on smoothing map data and they had a team of mathematicians working in West Covina smoothing out the data. But that was an extremely successful system. It lasted for as long as there were navigators in the Air Force. I think I went and got the article; I think 20,000 — something like that — navigators were trained. We can look it up.

Yost: How large was the group in West Covina?

Boebert: Oh, it was big. I mean, they were building customer; you have to realize it was a custom special purpose computer that calculated the radar equation, which is why Honeywell got the bid, so that was not general purpose software. That was a hard wired machine. And so there must have been 250 people at West Covina; it was a big, big operation. I don't know exactly.

Yost: And there really wasn't any precursors to learn from in doing it, you just had to [pause]

Boebert: No, this was a straight, clean sheet of paper operation because no one had been able to calculate the radar equation in real time before, so nobody had done radar simulations with a computer before UNTS arrived. Before they did that, they used transparencies and a flying spot scanner. So, if you can imagine, the instructors were moving this little beam of light across a big sheet of plastic that has the various densities that's supposed to be the radar return for where you are. And then that beam of light — it was all analog — that beam of light would go off and be displayed on the PPI display. Nobody had digitized any of that stuff before. We had looked at various designs for the storage, one of which was all solid state. If we did the solid state one it was going to take something like the entire semiconductor production of Silicon Valley for 18 months or something like that [laughs] so it was pretty wild.

Yost: Can you tell me more about the programming effort on that project?

Boebert: I didn't have that much to do with the programming effort. They were Honeywell 316s, as I recall; 150 of them. And they were all programmed in assembly code.

Yost: Was it difficult to get code running well and was there a long debugging process?

Boebert: Oh yes. But — and this transferred to the project management seminars that I devised — but what I learned from that was that this was a disciplined bunch of guys. They'd done trainers, fire control systems, bunch of no-foolin' technology, which is what I tended to gravitate towards. And they did a very strict disciplines waterfall development cycle, which they could afford to do in those days because there was a lot of money around. And so the software activity looked like a hardware activity; specs, test plans, all that stuff.

Yost: And I understand it was used for the better part of four decades. That's amazing.

Boebert: Yes it is. I forget when they converted it off the disks, but somebody basically rewrote it as a, you know, you didn't need a building anymore to put the thing in.

[Laughs.]

Yost: How long were you in West Covina?

Boebert: I shuttled back and forth. I was Western Airlines' number one [customer].

[Laughs.] Rode the Western Airline red-eye back and forth every week basically for the better part of a year, I guess.

Yost: What were your thoughts when you first learned that Honeywell might be acquiring General Electric's computer division?

Boebert: We didn't hear about it until it was a done deal. So there I was. [Laughs.] And that's how I got involved in Multics, because as an advanced development group, they were — as the M&A guys call it — going through the chocolate box looking at this little piece and that little piece and deciding what are we going to with this, what are we going to do with that? So there was this Multics thing sitting out in Cambridge so they decided that our little group in Minneapolis should be the ones to take a bite out of that particular piece of chocolate and decide what to do with it. My immediate supervisor in those days was kind of a jerk and very self-confident, so he's; I mean, the stuff came in boxes and piles of paper, and of course, all the GE guys were terrified and fearing for their jobs. And it was an emotionally terrible situation to have to manage in. So he threw this description and the stuff that the guys at CISL [Cambridge Information Systems Laboratory] had written about what they were doing, sort of their proposal about keep us, you know? In my lab on my desk is a memo that said fly out to Cambridge and come back and write a memo that says that we should shut this thing down. And so I flew out to Cambridge and got the tour, and talked to Charlie Clingen, and John Gintell [?],

Bensoussan, and Corbató, and so forth. Luckily — I guess luckily — that previous system design exercise that I'd gone through when I first joined the company led me to think out a lot of these problems of linkages and descriptors, and what you could more or less call quote/unquote advanced computer architectures for the period. So I sorta understood this stuff. And so I went back and wrote a memo that said we should exploit this thing. So that was one of Multics many near-death experiences; that was their first one. Tom Van Vleck's got the memo someplace in the Multics archives. And my boss was very un-amused but I didn't care.

Yost: At that time, what did you see as the potential market growth for Multics?

Boebert: At that time, we had done and we had been involved with a system that they tried to productize and never succeeded in doing it in the 800 days — and I can't get my chronology straight but it was before Multics — and this was a system for Metropolitan Life. It was a remote job entry system so you had a little batch computer at every office and every night the — whatever the hell insurance agents do — but anyhow, the guy doing whatever paperwork had to interact with headquarters. This guy would punch up and stick in this terminal, and then while he was home the central machine would pull these things and suck this up and do all the updating, the policy writing and commission calculating; whatever the hell an insurance company does. Well I mean I knew what they did, but all that stuff. And so it was very clear that the remote access and what they called the computer utility model, which Honeywell legal refused to let us use because they thought that then that would cause the government to try to regulate it. [Laughs.] This

was clearly the wave of the future and actually I thought of interactive terminals, so basically I thought really of just a massive remote job entry architecture, using; it looked to me that the match was good between the smaller, cheaper — well small, they'd fit in this room, you know but — for the day, the smaller, cheaper Honeywell machines, a central multi-user one, and immediately tumbled to the notion that process isolation was the way to go. Before the merger, we had an account at Dartmouth and we played with the Dartmouth Time-Sharing system in this group and had written some applications. And had captured other peoples' processes, and you know, it was very clear that architecturally, process isolation was essential. And here it was, and so this was the thing to do. Of course this generated the undying animosity of the Phoenix organization that had the competing operating system, GCOS, which we called God's Chosen Operating System, and referred to them as the dinosaur factory. [Laughs.] Phoenix. So that started the multi-year award, replace GCOS with Multics, which we never succeeded in doing.

Yost: Were there security design elements with GCOS?

Boebert: Well they claimed there were but they all sucked. [Laughs.]

Yost: And it was the experience, in part, that you had with the Dartmouth Time-Sharing system that recognized the need and the value of computers that had a security design —

Boebert: It was just kind of obvious that you had to keep these streams separated and, I guess it was just ingrained in me because we had done multi-level processing in the Air

Force where, you know, there you were separated temporally instead of electronically, but it was the old take down all the tapes, purge the machine, run the classified application; take down all the tapes, purge the machine, purge the machine again, purge the machine again; run a test to make sure you purged the machine; sign the piece of paper to attest that you purged the machine; and since you signed the piece of paper, you go ahead and purge yourself [laughs] just to make sure that nobody from the Air Force Office of Special Investigations sneaks in and sees one of the bits.

Yost: In going out to Cambridge and going to those meetings, and being told to write the memo, did you have some significant discussions about the security model with Multics, with the principle —

Boebert: Oh yeah. And I think Roger Schell had; they were primed with that stuff because Roger had just graduated. Let's see; Roger was an avid student, I believe [pause]

Yost: Was this in 1970?

Boebert: Yes, this would've been early 1970. So the whole bandwagon was starting to rumble up. Paul Karger I think had just gone to work with Roger at Hanscom, and so all that stuff was cranking up.

Yost: Several years earlier, Willis Ware headed a session at the Spring Joint Computer Conference in 1967 on computer security and privacy. Were you aware of that?

Boebert: Oh yes, yes. We were aware of that stuff, and ADEPT-50, and we knew Marv Schaefer.

Yost: And Clark Weissman?

Boebert: Oh yes, and Clark Weissman; and some classified work with Ted Glaser. These are all SDC people. And Ted became a long term friend.

Yost: Was he still at MIT or had he moved to Case?

Boebert: I believe Ted — I've been with Ted up until he died; worked on so many different projects — I can't remember. I think by that time he was at SDC but I don't know. And I don't know exactly when we came [pause]

Yost: So were many or most of the GE employees that worked Multics kept?

Boebert: Yes. We kept them and that was an unholy battle. It was a good thing that I had learned organizational politics from a Machiavellian Italian colonel from New York City. [Laughs.] So it was just dismal and of no technical interest whatever, let's put it that way.

Yost: Okay. Were there fundamentally different cultures between the computing group at GE and at Honeywell and if so, what were the primary differences?

Boebert: Oh yes. Now obviously I'm biased because I came out of Honeywell, but it was a very straight arrow operation and GE was like Italy under the Borgias. So after we got knifed in the back a couple of times, and lied to, and what they considered in their culture a satisfactory standard of behavior, we knew what we were in for.

Yost: What were your thoughts on the Multics security design as it existed then and can you compare and contrast it with your knowledge of ADEPT-50?

Boebert: I didn't know that much about ADEPT-50, other than some background study and conversations with Marv Schaefer. Dan Edwards had come up with Trojan horse idea. Somewhat it was contemporaneous with that so once you saw that it was clear that there had to be a further control on privilege and management of bindings than the Multics discretionary access control model provided. So that was kind of the simple problem; that was the problem on the table of the day.

Yost: In the early 1970s was there continued substantial interaction between your group at Honeywell and Corbató, and Jerry Saltzer, and others?

Boebert: Oh yes. We morphed into a thing called Multics Special Projects, MSP, which I was the manager of, and I chose that name because it was the airport code for Minneapolis. And we were the marketing; we were the federal marketing and liaison to

commercial marketing group to productize Multics so we were on the point of the political battles, which as I said, was dismal. And we eventually lost.

Yost: Was there interaction with the Anderson committee?

Boebert: Oh yes. I knew Jim very well. I met Jim Anderson [pause]; he was part of the gang so it must have been like, I don't know, the first party Roger Schell threw at Hanscom. Jim was there and it turned out we had common friends because he knew the guy who hired me as a machine operator at Stanford when Al Collins had worked for Burroughs. Both Anderson and Collins worked for Burroughs. Computer business was pretty small in those days, certainly as far as senior people. I mean, if you think of the pyramid, you know, I wrote my first program in 1958. I don't know how many programmers there were in the United States but there weren't a lot, certainly not by today's standards. And so it was like real mysterious stuff.

Yost: SDC had about 800 and I think that was probably a third to half of them.

Boebert: Yes. So I would guess that in 1958 there were probably no more than two, three thousand in the United States at most, sprinkled around all these places. But we all knew each other.

Yost: At that time, even IBM was well under 500, in terms of programmers.

Boebert: Oh yes, because they were still basically; I mean, they had the 7090s but they didn't; I mean, they were loaders. IBSYS, IBJOB, oh God! [Sigh.]

Yost: So in interacting with Roger Schell and Jim Anderson and the committee, did you see the Air Force and what they were trying to do as a really essential market opportunity for Multics?

Boebert: Oh yes. Our business plan was to; that was actually a thing that Renier had gotten Honeywell doing and after the whole thing collapsed and I washed up on the beach, I ended up — thanks to Jim Renier — at the Systems and Research Center in Honeywell. But the idea was to leverage government investment; leverage government contracts into commercial business and that was a fundamental Honeywell business model. And we were on the same page with the customer on that because the customer wanted this technology to be distributed.

Yost: What was your opinion of Bell-LaPadula when the papers came out?

Boebert: Humph. That's all I have to say. [Laughs.] I did not think that embedding a 17th century bureaucracy into the guts of your computer system was the way to solve the problem. How's that? [Laughing.]

Yost: Fair enough.

Boebert: Not that I necessarily had a better idea but it just didn't; and then we found the various problems with it, logical problems with it.

Yost: What about what came out of the Anderson Committee with the secure kernel concept?

Boebert: That was all Multics. That was obvious. I mean, that's the only way to describe it; it was just obvious. You just kind of sit there, watch the briefing, nodded your head and said yeah yeah yeah; next slide, next slide, yeah. Next slide, fine; that's the way you want to do it.

Yost: In those first years, in leading this Multics team, what kind of process was there for ongoing security enhancements?

Boebert: We got the B2 contract, and so the process was pretty much the Honeywell business model. Now, that makes it sound like a helluva lot more organized that it really was, okay? You have to realize that as the manager of Multics Special Projects, I was sort of conducting a marketing effort in a war zone because we had Phoenix doing everything they could to co-opt, pervert, etcetera, what we were doing. So it's hard for me to claim that this was a really cool planned; I mean it was a mad scramble, is the best way to describe it. Marketing is always a mad scramble but this was a two dimensional mad scramble because I had the rear area to deal with. The problem from their point of view was that their empire disappeared if the Honeywell customer base moved to Multics.

Now, you have to understand the economics of the computer business at the time to understand why this battle existed because in today's world if you went to Apple and said hey, I got a new product and this is going to obsolete your product, Tim Cook would say of course, that's our business model, right? [Laughs.] Well, in those days that's not the way that the large-scale computer business worked. Large-scale computer business was essentially modeled on heavy industrial equipment and it was one hundred percent — almost one hundred percent — a lease business. So you had economically, for each installation, you had the infamous and deadly hockey stick cash flow curve, which said that you started out way in the hole, because Honeywell spent x million bucks building one of these beasts, and then the customers got it for — pick a figure — \$500 a month and that's, you know, that's not right. But anyhow, they got it for some fixed amount per month and then you slowly climbed out of this hole. And after a rather extended period of time — Honeywell was very aggressive in their leasing arrangements — but say; oh let's just pick a number; three years. Three years you were break even. Everything after that was gravy so the business inducement was to keep the iron in the customers shop as long as possible, and this is the business model that made IBM rich, okay? Well obviously, if you come up with something that everybody — and oh, the installed base, the GE guys called the installed base the PARC, Purchased and Rental Computers — well, there were no purchased computers to speak of so they were all rental computers. And the thing they used against us constantly was churning the PARC, which was if you had something that was really new and cool, all these guys that were running these dumb old GECOS machines were gonna say okay, take it back. They don't own it; it's a lease, right? Maybe a three-year lease, maybe a five-year lease, but at the end of the lease period, they're

gonna want the new one. Well you're gonna get killed on the cash flow because now you're making a jillion new Multics machines, you follow me? You're going to bankrupt the company with this thing. So it was, I guess in retrospect, we were foolish and young enthusiastic technocrats because economically it was an impossible proposition. And so what they had — and then of course from the point of view of the ex-GE business, the GECOS business, they were dying on the vine because there's no place for them to go. So that was the nature of the problem. So we were off doing this crazy thing and it's actually — again in retrospect — it's amazing that we kept the thing alive as long as we did. [Laughs] But it finally got to the point where the hardware situation was impossible and then Honeywell decided to pull the plug on the computer business, and we were all surplused.

Yost: Can you tell me about your involvement with the Project Guardian proposal and that effort?

Boebert: We were the customer side of it. We did the mechanics of the pasting it together and all that stuff.

Yost: Was there extensive interaction with Saltzer and others?

Boebert: Oh yes. I mean, we were basically the marketing guys, but yes.

Yost: I believe it was in 1974 [that] Tom Van Vleck made the move from MIT to Honeywell. Was that a common transition back in the day?

Boebert: Yes, it was a pretty porous barrier.

Yost: With some people going to Honeywell from MIT, and were other going from Honeywell to MIT? —

Boebert: Most of them came; most of them we recruited into our place. I mean, there was this screwy two-headed organization. Multics Special Projects was the free-floating liaison, marketing group reporting to corporate headquarters, and using their political clout with the corporate management to keep the rest of it going. But CISL technically reported back to the dinosaur factory. So I had a small group resident at CISL that were actually paid out of Minneapolis, but all the Honeywellers can tell, Clingen and all Clingen's people, they all were stuck with Phoenix as a management so it was a mess.

Yost: The Air Force, of course, was a driver and very interested in the development of security features.

Boebert: Right.

Yost: Were there other parts of the federal government or companies actually pushing for certain security features?

Boebert: Well, the big car companies; GM and Ford were receptive and we got installations in there. And then there was a sort of parallel classified track with NSA that did stuff and was interested in it. Finally ended up as the National Computer Security Center.

Yost: With the auto companies and potentially other corporate customers, can you talk about how they viewed computer security relative to tradeoffs with efficiency of operations?

Boebert: Sure. When you have; in the case of GM and Ford, who were both organized in pretty much the same way, in the classic concept of the corporation model, which said that if you pit the divisions of your corporation against each other they'll become so agile and efficient that they'll eat up the competition, okay? So the biggest secret in Chevrolet Division of General Motors [was that] since the Chevrolet Division of General Motors made hubcaps for Buick, the Buick Division; the biggest secret in the Chevrolet Division of General Motors was how much it cost them to make a hubcap so the Buick VP could beat down the Chevy VP on this internal cost transfer. So it was . . .

Yost: Accounting information was multi-level like government classified documents . . .

Boebert: . . . it was totally multilevel, okay? It was just like a classic multilevel security problem because these divisions were at each other's throats. And Ford was very much the same way, although GM was much more bureaucratic.

Yost: And were there customers in some other industries? Insurance, banking?

Boebert: Yes, we talked to a lot of banks. God, I talked to a lot of banks. And of course, bankers are the worst customers in the world because what you do is you pitch them on a concept and then they come back and say well, our analysts have looked at it and we've decided that this machine should cost \$4.95, you know. [Laughs.] It's just impossible. No, we never really; you know, we had the Air Force installation, we had the National Computer Security Center installation; I'd have to go through the installations, but the ones that I was most focused on were the government ones because that was our pipeline of technology, supposedly, and technology; money that we were charged with nurturing.

Yost: Can you talk about your vision of how the computer security research community was evolving in the 1970s?

Boebert: IBM was fighting it tooth and nail and so IBM had their guy — God, I can't think of his name, he was one of their ex-salesmen; had a P-51 for his private airplane — and his job was going around telling companies they didn't need it and it was a people problem. They ran ads saying it was a people problem, and they went to conferences saying it was a people problem, so that was another battle we fought.

Yost: That was Bob Courtney?

Boebert: Courtney, yes! God yes! Good ol' Courtney. Yes, a *foeman* worthy of your steel was Bob Courtney. He was very, very good. Excellent schmoozer. We'd come at him with technical arguments in the various arenas, conferences and so forth, and they'd run ads saying; I guess the famous one — and I haven't got a copy anymore but you should dig it up — is the one with the yellow line down the middle of the road. Have you ever seen that? 'You don't have a computer security problem if everybody stays on their side of the road.' The question is of course, what if they don't? [Laughs.] But yes. So there was huge resistance to any kind of technical computer security problem, on the part of the dominant supplier of equipment of the day.

Yost: So in 1976, IBM comes out with RACF. What was your reaction?

Boebert: Well sure, that was clearly a defensive; you know, it was a joke; it was a defensive; yes, we've got this thing. It was like the difference between a VW and the early subcompacts that Detroit put out. Yes, we've got a little car, you know. It's a Chevy Chevette, you know, this godawful thing [laughing] designed by people that didn't want to have anything to do with that particular problem space. That was sort of RACF.

Yost: In the development laboratory, they actually had security by default, but they stripped it of that because they wanted a weaker product that customers could claim they

were doing something with security. And I believe the Foreign Corrupt Practices Act for international business required that there be some type of access control.

Boebert: Right, something that did something, but who cares what as long as the box is checked?

Yost: That might not even be turned on.

Boebert: Exactly.

Yost: You mentioned the National Company Security Center, the DoD center that NSA.

Boebert: Right.

Yost: Did you have any involvement or were you consulted with regard to the development of the criteria for the Orange Book?

Boebert: Oh, yes. We were up to our ears in the criteria development.

Yost: Can you expand upon that?

Boebert: Well, you mean all the time I spent in Baltimore? Developed a taste for crab cakes, anyway. I guess the criteria went through what I refer to as The Life of Brian Syndrome, if you know the movie.

Yost: Great movie.

Boebert: You say a couple of reasonable things in the public square and you come back a little later and discover to your horror that you founded a religion. Okay. The criteria was intended as a guidebook for sensible evaluation in the manner that NSA evaluated their own equipment. So it basically said look at this, look at this, look at this, look at this; go back, have a sensible discussion with sensible people — some of which I've been in got pretty brutal because you're talking about lives and wars at stake — but anyhow, get together and decide whether all things considered — it's kind of a risk management deal — all things considered, based on what we know about the threat and what this thing does; and all the other jillion factors that go into it; we can say yeah, you can roll this sucker out, okay? And then it turned into a religion. And as a consequence, which we didn't realize, as the interface to the commercial world, was fatally flawed. The thing that killed the criteria was the existence of the constantly escalating threat. And so you had the phenomena of criteria creep, which is; let's say I'm Honeywell and I'm going to go get a rating. And so I get my rating in 1973 and my rating is B2, and that's B2 based on the threat level of 1973, and to get that rating I spend a million dollars. Now in 1975, CDC comes along and wants a B2 rating. Now you've got a real problem because a 1975 B2, sensibly, is not the same as a 1973 B2 because the threat's increased. And so the

raters, the criteria evaluators, come in and say okay, within the interpretation of these various gray areas, of which there are multitudes, you've got to do more than Honeywell did and of course, then CDC screams you're being unfair. You following me? It just couldn't work, and didn't work. That eventually killed the whole idea of criteria, so far as I'm concerned. So you're back to the old informal way that people used to evaluate crypto gear, which is you take a bunch of smart people with bad attitudes and they tear into the thing every which way they can. And when they get through with it they come up with a list of vulnerabilities, and then somebody in management says I accept that list as being most of which we can cover with so I accept that list. That says this machine is secure as long as you lock it in a safe every night, or never let it out of your sight, or you know, strap it to your ankle or whatever the hell the rule is.

Yost: Can you talk a bit about the Tiger Team efforts of Roger Schell?

Boebert: We did that, and that was out of that tradition.

Yost: In your mind is that a very useful technique to [pause]

Boebert: Oh, it's essential; adversary analysis is essential. But the problem is that you have to be in an environment where the upper management understands the problem. Checking boxes — and safety has the same problem — checking boxes is a narcotic, right? Something that the human brain wants to do really badly, and so you; well, you can just go on and on with the instances where — I don't care whether it's physical security

or safety or whathaveyou — where all the boxes are checked and something horrible happens.

Yost: Did you attend some of the early IEEE Security and Privacy Symposia?

Boebert: Oh yes.

Yost: Can you tell me about your view of that event and that group that came together, what was achieved?

Boebert: It was the only game in town for the longest time, and the Claremont was a cool place, and we were all friends, and I don't know what else to say. The problem was it could come up with a variety of point solutions or interesting observations or whathaveyou, but you really needed a project like Multics to learn the real lessons because all the real lessons come from systems integration. So if you're not building a big thing that's going to have to work, and live in a hostile environment, then you're not engineering. Okay? I mean, you're making components and that's fine, but two wings, four engines, a group of instruments and so forth do not make an airplane.

Yost: Right. Can you tell me about the SCOMP Project?

Boebert: Yes. That was the initial of a long evolution that ended up with Sidewinder. It's one blur in my mind. SCOMP started out with an observation by Marv Schaefer when he

did KVM, that the whole exercise was futile because the communications process was the weak link on the system. So you had this nice isolation box, and then in front of it all the wires came in from something that had no isolation at all. So we decided; the front end of Multics was a machine called 355, I think that's what it was called. And it's this room-size communication multiplexer, right? So we got the bright idea — I don't know who came up with it, I didn't, but somebody did, might've been Jerry Whitmore — that we would do a thing called the SFEP, the Secure Front End Processor. And that was, as I recall, the first nubbin of the idea and now we're getting into the blur where I can't remember which one was what contract and who did what. But anyhow, so we kicked that around and I think we got a lot of help from Morrie Gasser. Somehow the consensus arose that this thing was to be a mini-Multics so it was supposed to have something like rings, descriptor segments, and all this stuff because we understood that architecture and it was sort of the same problem. So we were going to make a mini-Multics and that morphed into SCOMP when we decided to make it a freestanding computer instead of just a communications front end.

Yost: At the start of the project was there the goal of developing an A1 system?

Boebert: Yes, whatever A1 meant. We just wanted to make it hard to attack. And work.

Yost: Can you talk about the certification process?

Boebert: I wasn't that involved with it because by that time the project had moved down to Honeywell St. Pete operation, and those guys worked getting-the-boxes-checked stuff.

Yost: What were the greatest challenges to overcome in developing SCOMP?

Boebert: [Ha] UNIX. We had the only machine in which pointers were not integers so every piece of C code in existence broke. [Laughs.] We read the spec, you know; bigger fools us. But yes, as I recall it was pretty straightforward engineering exercise. As I said, kind of classic, straight waterfall; write the spec, so forth and so on.

Yost: What did you see as the market for such a system?

Boebert: Well, there were a lot of little communication boxes running around the world in those days. BBN, you know, they were building IMPs like crazy and that sort of stuff was pretty much it. Again, you had to go back to that business model. We had a contract to build a nifty widget, and then once we got a nifty widget we'd sort of figure out what to do with it. So that was the leveraging investment thing.

Yost: Can you tell me about your role in contributing to the design and verification software for the Saab J37B autopilot?

Boebert: We did theoretical work on modeling and setting up the test plans, and stuff like that. And of course, you couldn't call it software. Saab threatened to cancel the

contract if Honeywell called what was in that airplane software or anyone that worked on it a programmer. [Laughs.] So we were all engineers and it was all firmware. And then later on we did a; after the aircraft was flying, a guy by the name of Ed Rang [?] and I did a R&D project jointly with SRI to try to prove an invariant, and we failed. The invariant of interest was a scaled fixed point arithmetic system, and what we devised as a problem was to analytically demonstrate that the code would never either underflow or overflow, and we couldn't do it. That was using SRI stuff; what the hell did they call that specification language. SPECIAL. Anyhow, that was the Peter Neumann group stuff.

Yost: What about the Mark 48 torpedo?

Boebert: I can't talk about that. It was straight ahead engineering.

Yost: The space shuttle main engine controller security?

Boebert: That was static analysis of pseudo code that was then converted into assembly. On all of those; by that time I was a Honeywell fellow. Honeywell was a dual track, dual ladder company, and I was kind of a journeyman software project consultant. Renier referred to us as the priesthood; he said we only showed up at births, marriages, and deaths. [Laughs.] We'd go in and help an engineering group get started, or be involved in the merger of a couple of projects, or be involved in going in and shutting them down, so I saw a lot of stuff.

Yost: In 1984, the research group, the Secure Computing Technology Center, was formed. Can you talk about that organizationally, and was that kind of a continuation of SCOMP?

Boebert: That was actually; SFEP begets SCOMP; and SCOMP begat PSOS, and PSOS begat the Secure Ada Target, and the Secure Ada Target begat LOCK, and LOCK begat the Standard Mail Guard and Sidewinder, and that was all one continuous thing. So you have to go back to PSOS. SRI had sold NSA on the Provably Secure Operating System project and we bid the hardware, and won. The engineering was out of Ford Aerospace so you had a three-headed monster. You had SRI doing the analytic work, Honeywell doing the hardware, and Ford Aerospace as the system engineers. Ford Aerospace was worthless. So Dick Kain and I — you know Dick is emeritus, you ought to get him, too.

Yost: Yes, we have plans to.

Boebert: So Dick Kain and I came up with — this is a capability machine, ticket oriented architecture — so we came up with the idea of grafting a ticket oriented architecture onto the front end of Multics for efficiency; on the front end of a descriptor architecture. So in the PSOS design there was this magic phrase about presenting the capability to the machine. Presenting the capability to the machine involved essentially an extended version of the Multics. So you would take the ticket and you would look at the ticket and make sure it was a good ticket, and then you'd bind it to a process as a descriptor because now you could cache stuff and you knew how to do things fast. That was the architecture.

Well, so we're sitting here and we've got this concept but we need a lot of details in order to build the hardware, right? Like what's the register set, what do the operands look like, how big is bleeping word? All this stuff. [Laughs.] So Dick and I would go to these meetings and Ford Aerospace would dither, and dither, and dither, and finally, we said okay, you won't tell us what the next layer above us is so we'll design one so we can get going. So we designed one, and then we'd go to these meetings and they'd dither, and dither, and dither about the next layer up, and then we'd say we designed it. When Dick and I found ourselves designing the file system the NSA contract monitor, who was a really tough, smart lady by the name of Hilda Faust; Hilda said in one — from my point of view — one wonderful meeting — and NSA could do this in those days because of the freedom they had in their contracts — we all sat down with Hilda Faust, and Ken Shottling was there, and Matt McClarty, and Dan Edwards was there, and they're all sitting there with these kind of funny expressions on their faces; and the Ford Aerospace guy is about to stand up and give another dithering lecture about how bad everything is, and how many problems they have, and so forth; and Hilda says, before you begin, you're fired and Honeywell has the contract. [Laughing.] And it was in our office, it was in the Honeywell office in Minneapolis where she did this! So we ended up working on PSOS, and then for a variety of reasons, it was decided not to proceed with the analytic stuff. But we had all this technology so I came up with the idea, because Ada was big and of course, I'd been on the Ada project, which means my signature resides in some dark corner of the Académie Française because Ichbiah came around with the Ada manuals afterwards and had the presentation copy that we all signed. So Ada was looking like it was going somewhere so we said why don't we do an Ada target, a secure box that runs

Ada. And that went along for a while, and that morphed into LOCK, and then LOCK eventually became the Standard Mail Guard, and then moved on to the other hardware and became Sidewinder.

Yost: Before we move on to talk about the Secure Computing Corporation, can you tell me about the course that you created on systems engineering and project management?

Boebert: It started out when I was in the Honeywell Systems and Research Center, which is where I ended up after Multics collapsed. One fine day, at the staff meeting of the Honeywell Aerospace and Defense Group, the vice president for Aerospace and Defense discovered that every single project on his critical projects list of things that were in desperate trouble was a software project. [Laughs.] So they said do something about this; and he turned to Renier — who I think was the head of systems and research at the time — but anyhow, he turned to his buddy and said fix this. And Renier said I know just the guy. And so Carl — I spent so much time with the guy and I can't remember his name but anyhow, I can see him right there in front of me — but anyhow, another Honeywell engineering manager from St. Petersburg and I put together a software management seminar. It ran two days and it was systems engineering and software engineering from the aerospace perspective, and by today's standards it was sort of "screw agile make it work" approach to life. [Laughs.] Not the sort of thing I think the Silicon Valley types would appreciate, but this was for building no-fooling systems, stuff that goes boom. And so it was, like I say, straight waterfall, small steps frequent reviews, management of change — which we call configuration management — test plans, partial integration

testing, all of that stuff. We gave it all around the company and then we started giving it to customers. And then my partner dropped out and it became my baby, and so then there was a time when you could not be a project manager at the Naval Weapons Center in China Lake unless you'd gone to this course. So I gave it and then, because of financial problems, I got permission from Honeywell to give it on the outside on vacation time. So it became a second job for me, second income, and it was run out of an outfit called State of the Art Seminars in beautiful downtown Los Angeles Airport by a guy by the name of Hy Silver, who had been the proposal manager for the space shuttle; the North American Proposal Manager for Space Shuttle. So he gave a how-to-win government contracts seminar and then I gave what to do after you've won the contract. [Laughs.] And I gave that all over the world.

Yost: I saw from your web page, 13 countries, three thousand students.

Boebert: Yes, 3,000 people. It was a big hit.

Yost: So how did Secure Computing come about?

Boebert: You mean as a corporation?

Yost: As a corporation, yes.

Boebert: Well, in 1989? I'd have to look it up. Was it 1989? Something like that.

Yost: Yes, I think 1989.

Boebert: Yes. Jim Renier was then CEO of Honeywell and he decided supposedly largely under pressure from his second wife, that Honeywell was going to get out of the military business. When the word of that came around we knew we were doomed because we were running the LOCK project and we knew exactly what the spreadsheet jockeys in corporate was going to do with us, which was milk the project for every last buck and then fire us all. So I picked up; I went to a Honeywell VP by the name of Roger Heinisch, who was liaison to the intelligence community and we discussed our problem. I walked out of his office with what I interpreted as his okay to put SCTC up for sale. And so I then went back to my office that very afternoon, picked up the phone and called everybody I knew in the intelligence community and told them that the Secure Computing Technology Center was up for sale by Honeywell. The first that the Honeywell vice president in charge of mergers and acquisition learned about this was when Bobby Inman's right hand man called him up and said how much do you guys want for SCTC? [Laughing.] The M&A guy was not stupid. Instead of saying the obvious, which was what's SCTC? Because we were this little five-million-dollar-a-year operation buried down in the bowels of the aerospace and defense group. I mean, they were talking about selling off all kinds of military electronics — big chunks of the company — so he finally found us and they came up with a price and so we spun off. There was a usual bunch of machinations in the spinoff exercise, but we had this contract to be transferred to the new owner and we had a very happy customer, and so we were a decent, attractive

property. We had an interim bunch and then finally we ended up under the wing of a venture capital group that kept us going until we went public, which was just about the time I left. Let's see, I left right before they went public.

Yost: So they spun off the company, it wasn't acquired.

Boebert: No. It was purchased by investors. That's not technically, legally true because these guys had an existing company, quote/unquote, but that company had no business. It was a shell. But that's the best way to describe it; for all intents and purposes, it was purchased by a shell corporation and then finally transitioned to being under the wing of an investment banking firm, and then went public. By that time I was gone.

Yost: So you were technical founder, and had the title of Chief Scientist?

Boebert: Right.

Yost: Can you talk about the work you did for the company?

Boebert: We did the LOCK Project. That was the biggest thing; and we did some crypto equipment, and we did everything we could get our hands on, basically.

Yost: How large a company did it become?

Boebert: I think when I left it was several hundred; might've been that big.

Yost: And when it started, it was . . . ?

Boebert: Fifty.

Yost: You served on a number of the NRC committees . . .

Boebert: Ten of them.

Yost: . . . including the top ranking 1991 committee that came up with *Computers at Risk*.

Boebert: Yes. I'm a national associate, which gives me bathroom privileges whenever I'm in Washington, D.C.

Yost: Do you know how that committee was formed for *Computers at Risk*, was [David] Clark selected as chair and did he choose the committee members?

Boebert: Yes.

Yost: One thing I noticed in looking at the list is it's primarily — there's a handful of academics but it's primarily industry people — and other than Willis Ware, who's at

Rand, not technically government, but there wasn't significant government representation.

Boebert: No, that was the point.

Yost: Was it in some respects a response to TCSEC not really working? Or not being influential to industry?

Boebert: It's hard to say because it was a DARPA idea, and I don't know what motivated DARPA to fund it. We didn't care because most national academy studies ran like runaway grand juries. We took the money and the customer would say now this is the narrow problem we'd like you to look at, and we'd go yes, thank you very much [pff-f-t] and go look up the real problem. So I just really don't know where it came from.

Yost: Can you talk about the process for the committee's work? Did certain people specialize in certain areas that became chapters?

Boebert: Yes, committee work is sort of like corporations. Each committee has their own different culture, but yes, that's basically it. People would contribute chunks and then there's this real grind to assemble the chunks into something like a coherent document.

Yost: I think it was chapter six, really, [that] seems to be the first significant analysis on the economics of computer security.

Boebert: Yes.

Yost: Can you talk a bit — was that seen as an essential goal of the committee?

Boebert: I don't think it was from DARPA's point of view but it certainly was when the committee went to work on the real problem. [Laughs.] The question was, how you gonna get this stuff out there? I mean, how are you gonna get people to do this? There was so much; the good committees — and that was an excellent committee — the good committees are very dynamic; the ideas are popping off people in the sessions. So to say who came up with what, that answer is I don't remember. But yes, that arose, like most of the things that come from the good committees, that arose organically from the discussion and the interaction.

Yost: What do you think was the impact of that report when it came out in 1991?

Boebert: Same as the impact of everything dealing with computer security, nothing. [Laughs.] I mean the stuff that; the issues were raised in chapter six have never been solved.

Yost: You were on about a half dozen other NRC committees?

Boebert: I think ten total.

Yost: Are there others that stand out in your mind, ones that you're especially proud of the work that was done?

Boebert: I was happy with all of them. Again, they all resemble each other so much. I mean, you go to that place, you sit in sort of the same rooms, and talk with many of the same people so separating them out in my mind is a little tough. No, I think we did a workmanlike job on all the ones I was on. The process pretty much insures that. I've reviewed committee reports that were duds, but I was lucky to never be on one that was a dud.

Yost: Can you talk about Sidewinder and what you see as the impact of it?

Boebert: Well, we shipped a lot of them and to our knowledge, nobody's broken one. So I guess it did pretty much what it was supposed to do.

Yost: What became of Secure Computing Corporation?

Boebert: Let's see. McAfee bought it, and then Intel bought McAfee, and so they're now, I think, there's still people that go all the way back to the original days but they're now all working for Intel.

Yost: In 1995 you joined Sandia National Labs.

Boebert: Yes.

Yost: Was that to principally work on computer security?

Boebert: No. Stuff.

Yost: Can you talk at all about your work?

Boebert: Not really.

Yost: Okay. Before we conclude are there any topics I haven't brought up that you'd like to discuss?

Boebert: No, I think that's pretty well; I would, going back to the college days I mentioned Forsythe; another powerful influence was [Richard] Hamming. I did the numbers for his first book, and he was another tough old bird. So the transition to the Air Force, after working with Hamming, the transition to the Air Force was fairly straightforward. [Laughs.]

Yost: Were there any individuals in computer security that you considered mentors in your early years working in the field?

Boebert: Colleagues that we were lucky to be kind of . . .

Yost: Learning from each other?

Boebert: Yes, the gang, as I said; the gang of 11 as somebody once called us. You know, Anderson, and Roger, and me, Steve Kent, and a bunch of other people formed a kind of a core group. That's about it.

Yost: Thank you very much.