An Interview with

STEVEN B. LIPNER

OH 406

Conducted by Jeffrey R. Yost

on

15 August 2012

Computer Security History Project

Redmond, Washington

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

Steven B. Lipner Interview

15 August 2012

Oral History 406

Abstract

Steven B. Lipner is a computer security pioneer with more than 40 years of experience as a researcher, development manager, and general manager in IT Security. He helped form and served on the Anderson Panel for the Air Force in the early 1970s (was MITRE's representative), oversaw path breaking computer security high assurance mathematical model work at MITRE later that decade, was a leader in Digital Equipment Corporation's (DEC) effort to build an A1 (TCSEC certification) system in the 1980s, and led the creation of Microsoft's Security Development Lifecycle in the 2000s. This interview focuses primarily on Lipner's involvement on the Anderson Panel, his work at MITRE, and his work at DEC.

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 University of Minnesota, and I'm here this morning, on August 15, 2012, on the Microsoft campus in Redmond, Washington with Steve Lipner. This is an interview for CBI's NSF-sponsored project, "Building an Infrastructure for Computer Security History." Steve, I'd like to begin with a few biographical questions. Can you tell me where you were born, where you grew up?

Lipner: Born in Independence, Kansas during World War II. My father was stationed there. Grew up mostly in Texas, up through the start of high school. Graduated from high school in Arizona; in Phoenix.

Yost: Who would you say were your greatest influences early in life, so pre-college?

Lipner: Parents, certainly. Had an uncle who was a petroleum engineer for I don't remember which one; one of the oil companies in Texas. Got interested in electronics, I think, through him, as I recall.

Yost: And when did this interest in electronics develop?

Lipner: Well, you know, back then people built electronics and were amateur radio operators and stuff, so probably late elementary school; junior high, high school kind of age; about 11, 12, 13 years old.

Yost: Did you enjoy mathematics or have an especial affinity for math?

Lipner: I did fine in high school and enjoyed mathematics; if I remember right, the mathematics teaching was better than the science teaching where I went to school. I remember the high school physics was pretty terrible, back in that era. But some, good math teachers, and other science teachers were pretty good.

Yost: And in middle school and high school, did you yourself do ham radio operation?

Lipner: Yes. Licensed, probably from age 13 or 14; continued as a ham radio operator through high school; kept a license for a while but never really active after I graduated from high school. Kept saying I was going to get back into it and never did.

Yost: You went to Harvard, is that correct?

Lipner: No, I went to M.I.T. I went to one of the Harvard Executive Programs much later.

Yost: Oh, okay. So at M.I.T., when you entered there, what were your educational and early career thoughts and goals?

Lipner: You go to sort of an average public high school in the Southwest in the 1950s, you didn't really have much of an idea what engineering was about. So I went with various ideas, you know, major in electrical engineering; major in mechanical engineering; wound up majoring in civil engineering because I thought they were doing some interesting things, including some interesting things with computers for design. The courses were more practical, somewhat more applied. At M.I.T. electrical engineering is computer science as well, now; but was not, then. Everything was sort of formative.

Yost: As an undergraduate, did you take E.E. courses?

Lipner: I did, yes.

Yost: Did you have exposure to computing systems as an undergrad?

Lipner: Yes, I started programming; taking programming courses as a sophomore, I think, and took a number of them both in the civil engineering department and electrical engineering. I took some; digital logic, digital design courses as an undergraduate, as well.

Yost: Do you recall what programming courses you took?

Lipner: Oh, yes. I mean, I took (pause)

Yost: FORTRAN?

Lipner: FORTRAN, IBM 7090 Assembler, actually had a part time job doing system programming on an IBM 7040; still remember some of the awful programming constructs that was used then in the IBSYS operating system; self-modifying code taken to levels that would horrify people today. But FAP, or FORTRAN Assembler Program, the assembly language for the IBM 7090 or 7094 back then; M.I.T. taught with a language called MAD, Michigan Algorithm Decoder.

Yost: The language out of University of Michigan. And what years were you an undergraduate?

Lipner: 1961-1965.

Yost: Okay. And did you have exposures to CTSS, as an undergrad?

Lipner: A little. I took a traffic simulation course in civil engineering as a senior, and we did the simulation runs on GPSS I think, GPSS or Simscript. Those runs were on CTSS, so that much exposure. We tried to put a remote access facility on the 7040 and had hellacious problems with making the communications controllers work.

Yost: Any exposure to the early work with Multics, at that time?

Lipner: Not while I was at M.I.T. at all. When I was in graduate school, we did some work with CP/67, the VMM. CP/67 was the initial virtual machine monitor time sharing package out of IBM that became VM370. And we had a 67 in graduate school; I did a lot of work using CP.

Yost: And did you go straight from undergraduate to graduate school?

Lipner: Yes. Still in civil engineering.

Yost: At M.I.T.?

Lipner: At M.I.T., yes.

Yost: And so, in graduate school you definitely knew about Multics?

Lipner: In graduate school I knew about Multics; I didn't have a lot to do with it. In graduate school, I mostly worked on a project called ICES, Integrated Civil Engineering System. And that was a civil engineering application package for the 360, and I actually built some system software for that using an ICES modified FORTRAN language and maybe some 360 Assembler. And then ran a little development group that built one of the application packages for ICES, a highway geometry package called ICES COGO; and also taught FORTRAN and Assembler to undergraduate and graduate students while I was in graduate school.

Yost: While you were in graduate school, what were your thoughts about where your career would go?

Lipner: Well, I thought I was going to get a Ph.D., which I did not. And so, you know, a lot of people tended to stay on the M.I.T. faculty so I thought I might do that. I talked to people in civil engineering firms, or architecture and engineering firms; didn't talk to any government agencies that employed people in civil engineering. On ICES and some other projects, we had some connections with some of the defense contractors that were interested in computer applications and I met people from some of them.

Yost: Which ones?

Lipner: Well, McDonnell Douglas was one and there was a fellow from the MITRE Corporation who participated on one of our projects. I got to be pretty good friends with him. I decided finally I wasn't going to stay on and beat my head against the Ph.D. program. I wound up talking to folks I knew and got a good offer from MITRE, and so took that in 1969, went to work there.

Yost: And what was your initial job title, and what were your initial responsibilities?

Lipner: Well MITRE, back then at least, everybody was a member of the technical staff so that was the job I had then. And actually, interesting, initially; the first project I worked on there was a review for the Air Force. I was part of a team doing a review for the Air Force of a system called ADEPT-50. I don’t know if you’ve run across that or not, but that was one of the pioneering, multilevel secure; or would be multilevel secure operating systems.

Yost: Yes.

Lipner: The Air Force was trying to figure out whether to deploy it, or support it, or sustain it. And I haven’t reread the report we did for that 40 years ago; but I think we wound up basically being critical enough about it so that the Air Force didn’t go forward with it.

Yost: Can you talk a bit more about the ADEPT-50 and do you recall what security design features that system had?

Lipner: I do. I was on that study very early, it was my first assignment at MITRE. I wasn’t really into or aware of security issues at the time. I came back and studied ADEPT-50 after I started to get into security. It had a mandatory security model, of a sort, and it labeled subjects, it labeled objects, labeled terminals, what have you; attempted to prevent information from being declassified in violation of policy. It had what they called the High Water Mark security policy and in retrospect, I recall that it had what we’d call now covert channel vulnerabilities. That was before people thought about covert channel vulnerabilities, of course. But it made a decent attempt at controlling access to labeled information and it’s certainly an interesting pioneering system of the time.

Yost: And were you evaluating this yourself or as part of a team?

Lipner: It was a team. It was like five or six of us and I was not much of a part of the team. The team had been operating, I was a new hire and I came in on Monday, and they plopped me into the team as part of that review. The review was not focused on the security, as I recall; more on how’s the development going. Is it going to be something that people can rely on? Sustainability, etcetera. I think they actually contracted with Jerry Saltzer, who was probably then still a graduate student or assistant professor, to do a review of the operating system architecture and maybe the security stuff.

Yost: To your knowledge, did MITRE have anyone on the technical staff that had expertise in computer security at that time?

Lipner: Yes. Although he wasn’t involved in that study, there was a fellow named Ed Bensley. You know the name?

Yost: No I don’t.

Lipner: So Ed was an associate department manager, MITRE old-timer, and he was one of the folks who participated in the DSB [Defense Science Board] Report; the Ware Report. That something you're familiar with?

Yost: Definitely, I've read the Ware Report several. And I've seen that list of names, Bensley just didn't register immediately.

Lipner: Yes. But Ed was off on assignment in the U.K., I think, and so he was not; I don't recall that he was part of the ADEPT review.

Yost: Roughly how large was the technical staff at MITRE at that point?

Lipner: The whole staff was probably between 1500 and 2000 people.

Yost: And were there divisions within the technical staff and was there an area that focused specifically on computers, or computers and software at that point?

Lipner: Yes, there was a division that did what they called the command control systems, and that was where the computing software research, or computing software expertise was centralized.

Yost: That extended directly from SAGE and SAGE system integration?

Lipner: Well the whole company extended from SAGE. But, you know, I suspect if you go back to MITRE this week, essentially everybody in every division is doing computing based stuff in one way or another. Back in that era, there was a computing command control division, a division that focused on radar, lots of signal processing engineers, and physicists, a division that focused on communication systems, and so on; and then there were mission area divisions. There was one called Tactical Systems that was helping the Air Force acquire SAGE-like systems.

Yost: Can you tell me how you became a member of the planning committee that was assembled by Roger Schell for the Air Force?

Lipner: Sure. So, you're skipping. Let me just give you a little continuity and maybe that would be better.

Yost: Definitely, yes.

Lipner: I worked on ADEPT-50 briefly, and then spent a year in Omaha working on the planning of a computer-based communications system for the Strategic Air Command. Came back in late 1970, and my manager at the time asked me to take over a small team of people because they were starting to see projects pop up dealing with computer security. And I said that I didn't know anything about computer security, which was certainly true in November of 1970. And what they really wanted was somebody with a degree in computer science, which was starting to emerge, and more of a specialization in operating systems and formal verification, which I at least knew existed [but] didn't know anything about. The guy I worked for said well, I don’t have anybody like that to assign. You did okay on the SAC assignment; why don’t you do this until we can get somebody with the right background and credentials. And that was in November or December 1970, and basically, they never quite got around to replacing me. And so we had three projects, initially. One of them was a multi-level security requirement for the Military Airlift Command; one of them was a multi-level security requirement for the Air Force data service center at the Pentagon—it was called something else back then — and a third one, which I’m not going to remember, and that was three peoples’ worth of work. And so we fooled around trying to figure out what the problem was; what the solutions might be; what you might do. If you’re interested, I’ve actually got a copy of the first paper I wrote on the Military Airlift Command Project in early 1971. I can share that with you.

Yost: Yes, we definitely would be interested in seeing that—and getting a copy for our collections if possible.

Lipner: I got MITRE to de-control it for me on its 40th anniversary, and so I have a copy of that. So we did that stuff; we got independent research money, or Air Force funded MITRE research money to start looking at what we called a Secure Communications Processor Technology. We did a project that sort of explored some of the issues, if you were going to try to do multilevel security; micro-coded up a machine to do some things; can’t remember all the details of what that was about. Morrie Gasser, which is a name you know, was hired and worked on that project. And Roger [Schell] came in; he got his

Ph.D. from M.I.T. in 1971 and joined the Air Force side that was sponsoring some of our work in the summer of 1971. And Roger came out of the Multics Project and he basically thought Multics was the solution, what was the problem. Multics was a very good system for the day, and so we started thinking about multilevel security and what could we do. I can't remember all the projects in that era, but we actually had a colonel who Roger worked for, and this guy's view; he was sort of a government science bureaucrat more than anything else. His view was if you've got a problem and you want to move people in the direction of a solution, what you do is you get a panel of eminent players and assemble that, and use them—a little cynical, but I think probably it was his agenda—use them to tell you some variation on what you already know. And that was the way we'd get the support to do the things we thought ought to happen. So Roger and I, sort of combed the list of people who were playing in the field at the time, to try to get folks who could both contribute and help us sell a solution. And there was a lot of thinking behind how we put the committee together. I don't remember exactly when I met Jim Anderson, but it was in the context of putting that together. We got a couple of people from NSA because that was obviously an influential organization, and then various other players; you know, Clark Weisman of ADEPT-50 fame, and so on; you have the list.

Yost: Yes, I'd like to go through the individuals and whatever you recall about them; why they were chosen for the committee and their contributions to the committee. First, Ted Glaser.

Lipner: Ted was, if I remember right; Ted was a consultant at NSA. I seem to recall he was on the Ware Report committee.

Yost: You're right.

Lipner: Yes. Jim knew him; he was also involved in Multics, so that was like four reasons to pick him.

Yost: He was officially the chair, but as I understand it, James Anderson really led the thing.

Lipner: That's funny. I mean, you say that—and of course, you're right—it was really Jim who drove it; Jim, Roger, and I, were probably the people who drove what went into it; probably in that order; or maybe Roger, Jim, and I.

Yost: And what were your first impressions of James Anderson?

Lipner: Interesting character; very pragmatic, technically very good. I think the last time I saw Jim was probably in the late 1990s, so it's been a long time. But thinking back that far, very smart, approachable, probably not as deep in formal computer science then or subsequently. But a very practical, smart guy.

Yost: And you've talked a bit about Roger. Melvin Conway?

Lipner: Mel had been in the Air Force at ESC [Electronic Systems Division – now the Electronic Systems Center]; I don't know where we got him. I think Ted may have nominated him. He wasn't a super active contributor. Smart enough guy; I still quote Conway's Law [software reflects the organizational structure that produced it] when I'm talking with people about building systems, but not a deep computer security guy.

Yost: And then from NSA, there were three people; Dan Edwards, Hilda Faust, and Bruce Peters.

Lipner: No, you're confusing Bruce with Bernie Peters;. Bruce was DIA. [Defense Intelligence Agency]

Yost: Dan Edwards and Hilda Faust from NSA.

Lipner: That's right. There was a Bernie Peters from NSA, who you're probably confusing.

Yost: Right that is who I was thinking of. He's the one that authored a paper with Willis Ware at the 1967 Spring Joint Computer Conference.

Lipner: That's right. Yes, so Hilda was more of a research manager at NSA, and one of the old-line security folks. Dan had a master's in engineering degree from M.I.T. Very much a deep technical, software and security guy; very smart.

Yost: Can you talk a bit about their contributions to the work in computer security?

Lipner: The way we worked; Jim and Roger and I sort of drove the agenda. It wasn't, if I recall correctly, it wasn't joint writing or joint brainstorming to come up with positions. It was more hearing from folks because we visited people or talked to people about their problems and approaches. But the development of the product was more driven by Jim and Roger and me, and then reviewed by the rest of the committee members.

Yost: So did the three of you meet more frequently than the whole committee?

Lipner: Yes.

Yost: So how often did you the three of you meet and then how often did the larger committee meet?

Lipner: Long time ago. I mean, I seem to recall Roger and I were sort of talking together on the phone or in each others' office every day, or every other day. Jim lived in Philadelphia, he was probably was up at ESC weekly, and then the committee probably met monthly or every six weeks. I think we had e-mail. Roger bought us Multics accounts and we did use that for some amount of communication, I think. I may have the timing wrong but I don't think so. So that sort of gives you a sense of the dynamics. Hilda probably focused a little bit more on the programmatic aspects because what we were recommending was an R&D program. Hilda focused on the programmatic aspects and on the crypto; to the extent that we overlapped into crypto. And Dan, more on the technology aspects, what'll it really take to build secure systems?

Yost: And Eldred Nelson?

Lipner: Eldred was a; perhaps you've read about him in books about Los Alamos and World War II, he was a physicist student of Oppenheimer's. He worked for TRW. He had built a procedurally secured system, basically an isolated system at TRW and published a paper about that. We got him on the committee and I don't remember how he came in; who nominated him. He [was] more focused on the R&D programmatic aspects of the report. I mean, he was a research manager at TRW so I think he was probably helpful in putting together the programmatic recommendations.

Yost: And Bruce Peters?

Lipner: Bruce Peters, he was at the Defense Intelligence Agency. I think they had a system then; or were building a system called DIAOLS the DIA online system. And I think he was involved in the security engineering of it. And so he knew some of the operational as well as some of the technical perspective.

Yost: Charles Rose.

Lipner: Chuck was a professor at Case Western with Ted and I think he was also a government consultant at the time. They were building a secure design system called LOGOS, and he was probably the Principle Investigator, or one of the Principle Investigators on that. And, you know, pretty much into the technology, sort of how you build it; it would've been formal verification if we knew enough about it to do that. So he was definitely focused more on the technology.

Yost: Anything you recall about LOGOS?

Lipner: Hardware and software design system with automation built on PDP-10s, whatever they were then; design graphics package; analysis packages. Never used it myself; probably knew more about it 40 years ago than I do today. But I think they were funded by ARPA and probably cancelled in 1975 or 1976. I'm sure they wrote papers about them but I don't have them.

Yost: Clark Weisman?

Lipner: Clark, senior manager at a defense company and a technologist, so he was very interested in how you build secure systems. You know, what it would take to build a secure operating system; lot of understanding of that. A fair amount of bitterness about

MITRE’s role in cancelling Air Force support for ADEPT-50. Clark’s view was that if the Air Force had gone forward operationally with ADEPT-50 we wouldn’t have the problems we have today—today being 1973—and wanted to see security research programs continued, both for business and technical reasons.

Yost: And what about the committee bringing in other experts, as well, to meet with and give opinions?

Lipner: We met with users; user organizations. I remember; and you can get a list of those in Volume II. I mean, the requirements working group report, was based on meetings with those user organizations. I can’t remember whether that was the full committee, or just Roger, Jim and me; or something in between. But we got that input. I don’t recall that we brought in research experts broadly to talk to the committee. There wasn’t much of a research community back then. A significant amount of the ongoing computer security research at the time was being conducted by the committee members. There were some others, There were some others doing research in this area, but I don’t recall that we asked them to speak to the committee.

Yost: The committee obviously got started after the Defense Science Board Ware Committee made their report.

Lipner: Right.

Yost: That report was heavy on problem definition on security in a multilevel environment; open environment and really didn’t see or offer much of a solution path at that time. I recall there was an emphasis on the benefits of keeping research open rather than classified; that industry might be helpful in finding solutions. When the committee got started was there a sense that it could be successful in coming up with systems that would work in such an environment that it could find solutions to the types of problems the Ware Report identifies?

Lipner: Roger certainly thought Multics would solve the problem, or some evolution of Multics would solve the problem; thought that was the right path. Of the rest of the committee members, I’d say probably that the technologists were relatively optimistic. And who would the technologists be? They’d be Roger, me, maybe Jim, Ted, Chuck; and then the program people, probably more cautious. We have to do this, but how far we’ll get is probably an open question. NSA people might have known how hard it would really be but were supportive of continuing research.

Yost: How did discussion of security kernel and reference monitor come about as a possible path?

Lipner: I’m pretty sure that Roger must have introduced that. I didn’t and I doubt Jim did. You have the Air Force reports? I mean, do you have the stuff that the Air Force produced?

Yost: We have some of the ones, a portion of the ones that weren't classified.

Lipner: There was essentially nothing there that was classified.

Yost: Most of the ones we have are those Matt Bishop collected and made available online.

Lipner: There was a report that pointed the way toward designing a secure system that Roger and maybe Gerry Popek and maybe others authored in early 1973. And that work was going on around the time that the; when did the Anderson Report come out? Was it 1972 or 1973?

Yost: October 1972.

Lipner: 1972, yes. So that work was going on, basically, in parallel with the Anderson Report work. So I think Roger thought that that would be a sensible way to do things. I think that's probably right; that's probably something he pushed for.

Yost: What was your initial assessment of that pathway?

Lipner: Well, it's hard to argue with the logic of that approach; and of course, you don't know all the problems of doing it until you've tried it; and so it made sense to me. I was certainly a supporter. We tried; we kicked off another Air Force research project in 1973, the effort that produced the MITRE security kernel for the PDP-1145. You know, the sort of thinking there; and could we do it. But we kept learning more as we went.

Yost: Overall, can you give a sense of the committee's reaction to that approach?

Lipner: I don't have a strong sense; a long time since those meetings; I mean, obviously, the committee was positive enough because everybody signed the report. Who was skeptical; I don't recall enormous skepticism at the time so I guess I would say people were supportive or supportive enough to give it a try. But, I mean, was there skepticism? Probably. But I think we all felt that it was an important problem and we had to do research to get moving.

Yost: Were there any alternative approaches that you recall that individuals on the committee backed and if so, do you remember what those were and who those individuals were?

Lipner: Not on the committee. The thing that came up later as sort of an alternative approach—and NSA was funding this—was more of a focus on building a capability system because the PSOS [Provably Secure Operating System] Project was going on at SRI at about the same time and we didn't have anybody from SRI on the committee. Hilda and Dan certainly knew about it; Roger and I, Jim knew about it. But we didn't take the path of saying that that was going to be the solution and there wasn't even a lot of discussion about it as an alternative.

Yost: And you mentioned that a parallel project that got started; the contract was it in 1973 for the MITRE security (pause)

Lipner: The PDP-11 kernel.

Yost: Can you talk about that project?

Lipner: Well, you could get three peoples' worth of money to do basically an experimental project at MITRE, out of the Air Force contract with MITRE. So we made a proposal; Roger supported it; we said okay, we'll get one of these new PDP-11/45 computers, which has memory management. You know, not all minicomputers did back then. And so using that, we ought to be able to implement a rudimentary security kernel and demonstrate what a multilevel system would look like. And we; the guy who did the engineering there, design and engineering, was a fellow named Lee Schiller. As I recall, the Bell-LaPadula model we had around that same time, but we weren't far enough to understand all the ramifications of the *-property and covert channels, storage channels. I can't remember for sure how the file system there was structured. It was sort of, a mini-Multics; but I can't remember whether it had a file directory hierarchy or not. The documentation would tell you. So I don't know whether it was storage channel free or not. But we built it and we built a simulated multilevel Air Force application. We demonstrated it. It was not a viable system from an operational standpoint. But it was an interesting prototype; we built it and demonstrated it; wrote the papers and then everybody moved on.

Yost: Were you the lead person in terms of they reported to you on this effort?

Lipner: Yes, I was the group manager for all the security efforts at MITRE in Bedford at that period.

Yost: And how many technical staff were working under you at MITRE in the 1972-73 period?

Lipner: It got to be as many as 15, 17, 18 people.

Yost: And when you first started with that in early 1970, how large was it?

Lipner: I think three of us.

Yost: And do you recall who the other two people were?

Lipner: No I don't. Neither of the other two folks stayed with the security projects – they were MITRE software engineers who happened to be available for assignment when the security projects kicked off. The people from MITRE who you have heard of all got involved later. Bell-LaPadula; Bell was hired in 1971; he worked on the secure communications project. LaPadula was a long-standing MITRE employee. He got involved probably late 1971-72. They did the model work after we had started to ramp up; probably after the Anderson panel was at least started.

Yost: In terms of writing the first volume of the Anderson Report, who was principally involved with that?

Lipner: Actually writing the document; Jim.

Yost: And then did you and Roger have closer input on it than the other committee members?

Lipner: I seem to recall so. I mean, there was probably; you know, as I say, there was Jim, Roger, me, Ted and then the other committee members more in the reviewing and commenting role, I think.

Yost: And at what point was the contract given that Bell and LaPadula begin their work?

Lipner: I mean, all these things were MITRE projects, and at that time, MITRE work program was somewhat discretionary to the Air Force. We probably started the Bell-LaPadula work; when did Volume I come out?

Yost: I think it was October 1972.

Lipner: Okay, so we probably started in the fall of 1971 on that. And Volume I; you know when Volume I came out we sort of thought we had a failure because it was basically a lot of formalism and not much you could act on. And then if I remember right, it's Volume II that introduced the *-property; and that was the point at which we said—and we in this case was the MITRE management, that was me and the people I worked for—who said these guys are on to something. They'd given us a set of rules that would actually allow us to meet the multilevel security requirement and figure out how to build a system or figure out what system could be built. So the Volume II report was a breakthrough, and then Volume III was a refinement in terms of getting the directory hierarchy; just starting to move into storage channels. And the Multics interpretation was more just a cleanup and elaboration. But Volume II was the real breakthrough.

Yost: And the Air Force provided the contract for Bell and LaPadula's work and at the same time, did they also provide a parallel research contract to Case Western?

Lipner: Yes.

Yost: Can you talk about what you know of that path, the Case Western research, and how it differed from Bell-LaPadula?

Lipner: Yes. The difference between Case Western and Bell-LaPadula; Bell, for sure, and I were involved in monitoring and managing the Case Western work for the Air

Force and there was pretty complete transfer of information—what we were doing, what we were thinking—from MITRE to Case Western, and the other way. Case Western was working; I'm not a formal models guy, okay? And not a verification guy. So what Case Western did was to provide a different set of formalism for the same fundamental requirement set, is the way I thought of it. And some of that, I think, was to sort of keep Ted Glaser's team engaged and in the game; some of it was to maybe tie in some of the LOGOS research, for all I remember. I don't remember what all the rationale was for that contract.

Yost: So it would be best characterized as collaborative and cooperative, but not setting up two competing teams?

Lipner: There was an element of two competing teams but we had access to each other's work. MITRE was both doing the research itself in that security model space, and also monitoring and managing contracts for the Air Force, so we were helping to manage the Case Western contract.

Yost: Can you characterize both David Bell and LaPadula, their backgrounds, and whatever you can say about their work?

Lipner: I think they're both still alive, to the best of my knowledge.

Yost: I have an upcoming interview with David Bell and we plan to interview Leonard Lapadula as well, but it is also helpful to get perspectives from peers.

Lipner: Dave was a fresh Ph.D. from Vanderbilt when we hired him in 1971 and very much a theoretical mathematician, I think, category theory. So we introduced him to this stuff. We said alright, we're going to need mathematical capabilities let's hire a mathematician and I think that was literally the basis for bringing him onboard. And he was very much a theoretical mathematician and getting him involved in the technical realities of all this, what are sort of the more engineering realities of all this stuff, what's the problem, how do you think about it? Len was; I think he also was a mathematician, I'm not sure. As I said, he was a longstanding MITRE employee, he had been there, gone away to one of the little minicomputer startups in the Boston area, come back, worked on command control systems before he got involved in the security space. The model development work was very much a collaborative effort. I think Dave was probably more the inspiration for what came out. I don't know which of them invented the "star"-property; probably Dave, but I'm not sure.

Yost: Were there ultimately any key insights or solutions that were delivered on the Case Western project or was that kind of a path that was abandoned because of the success of Bell-Lapadula?

Lipner: Well, I mean, there was a series of reports, which I assume you have. The Case Western work—there are other people who probably know more, can provide more authoritative answers than I can—but the Case Western work carried with it sort of a modeling or development formalism that I think development groups didn't buy into, or didn't follow up on. So I don't know that it went farther. It may have, but I'm not aware that it was as influential as Bell-LaPadula was.

Yost: You mentioned how Roger came out of M.I.T. with Multics. Were there key design principles with Multics that were influential to Bell-LaPadula?

Lipner: No. Well, I mean, subjects and objects I guess. But I'm not sure those came out of Multics; they may have. Better off to ask Bell.

Yost: Were you directly involved with the Air Force MITRE Tiger Team efforts in testing of Multics?

Lipner: Yes.

Yost: Can you discuss that?

Lipner: Yes. Paul Karger probably joined the Air Force side in 1972, that sound right? And there were a bunch of Tiger Team efforts going on. It was a hot area. As I said, we'd gotten up to about 15-18 MITRE staff by 1973-74. One of the original problems we had is the Air Force Data Service Center multilevel security requirement and they had GE 635s. And so it was not a stretch to recommend that they go to GE Multics computers as a next generation. Of course, Honeywell won the Worldwide Military Command Control System (WWMCCS) procurement, so people understood those kinds of computers back then. And Roger was instrumental in recommending that the Data Service Center look at Multics to meet their requirement. And then we also wanted to get the Bell-LaPadula model built in with the operating system. Honeywell, having to sell us Multics, they probably would have been happier to sell us more 6080s, whatever the next non-Multics computer was. They didn't want to undertake additional development work and so the objective of the Tiger Team was to demonstrate that there was additional development work required. And so we did.

Yost: So the Honeywell Corporation felt that the Honeywell Multics system was already secure.

Lipner: Yes. I mean, that was what you'd expect if you didn't know any better. So this was just a demonstration that there were some other things you need to do. The claim at the time was that the Multics penetration work didn't require inside knowledge of Multics. Technically, it didn't, if you had the Multics source code and enough understanding you could probably do what we did. But certainly having Roger knowing sort of where the design trade-off bodies were buried was an accelerator in finding the vulnerabilities that were used. Rome Air Development Center had a Multics system. So we did the testing on that; and then we moved the demonstration to the M.I.T. system, and then demonstrated to M.I.T. and Honeywell management. Paul did the bulk of the penetration coding; one of the mathematicians figured out how to crack the password file;

I built the code that cracked the password file; Roger helped me debug it after we discovered that the Honeywell PL/1 compiler did not function as documented. Basically, all the security in the Multics password file encryption was because of a bug in the Multics PL/1 compiler. So we had to figure out that bug to reverse engineer it to break the password encryption. And so we did that.

Yost: And what was the result or outcome of showing Honeywell these vulnerabilities?

Lipner: Well I think that made Honeywell a little more cooperative in the Multics security enhancement and more willing to propose and implement the Multics security enhancement project that resulted in the, what is it, the Access Isolation Mechanism, which is what they called their mandatory security package. You have the Whitmore Report, the design analysis?

Yost: I don't believe we do have that.

Lipner: Okay. You probably need to send somebody, or raid my library at home, at some point.

Yost: Yes, we definitely have an archival collection development effort parallel to these oral histories and I'll have our archivist contact you.

Lipner: Okay. Yes, that would probably be good. I have this stuff at home and I refer to individual reports once every five to 10 years. I've advised my wife that it shouldn't be taken to the dump when I die. I mean, if you have a collection and it's reasonable, it's archival, we ought to figure out how to take advantage of that.

Yost: Yes, we definitely have the infrastructure, and thirty-plus years experience, to provide a permanent archive for such types of documents.

Lipner: But what I was going to say, there was a joint project — MITRE, Air Force, Honeywell — not M.I.T., that specified the various security enhancements that went into Multics before it went operational.

Yost: That was the Honeywell Level 68?

Lipner: Well, the processor Honeywell was building was probably the 6180; but there were a batch of changes to the operating system software to add multilevel security, and we also addressed things like backups, and we talked about delivery procedures. If you consider it's 1974-75, it's a reasonably robust set of recommendations that we came up with. To the best of my knowledge, those were all implemented. I mean, the system was operational multilevel secret and top secret for probably a number of years; I don't know, 10, 20 years.

Yost: Did Honeywell have any computer security experts on staff?

Lipner: Yes. The Multics development group in Cambridge had some very good security folks; there were also some good security folks out of Honeywell in Minneapolis; they sort of came from the crypto-building world. You know the name Earl Boebert; you’ve run into him? I mean, he led; he wound up being the leader for that part of the effort.

Yost: Between the three organizations, roughly how many people worked on Multics security enhancements?

Lipner: There were probably three or four of us from MITRE; maybe two or three from the Air Force; probably three or four from Honeywell on the design analysis. And then I don’t know how many developers actually worked on building the operating system modifications. The whole Cambridge Information Systems Lab was maybe 40 or 50 people. Seems hard to believe that was an operating system development group back then. So I don’t know how many of them would’ve worked on implementing the changes.

Yost: Besides you, who was involved in this effort from MITRE?

Lipner: Morrie Gasser, again; Ed Burke; I think that’s it. There may be other names.

Yost: And can you describe the outcome of that effort?

Lipner: The design analysis report specifies the changes that were required to make Multics into a multilevel system for the Data Service Center. So, you know, labeling on files, labeling on segments; probably auditing, labeling on terminals, end user accounts. Morrie built a password generator for them that I think shipped with the product. Probably still has existence somewhere on the Internet, if you look for it. Delivery; we talked about how they could be sure that they could trust what they were getting. It's interesting, one of the things I worry about [now]; the call I was on before I came to get you this morning, was talking about supply chain security. And we had supply chain security considerations in the Multics design analysis report in 1975. Some things never change.

Yost: So I understand the roughly 1976, 1977 period is when the Air Force discontinued in-house computer research effort. What impact did this have on MITRE's work in computer security.

Lipner: Some, but not as much as we'd feared. I had moved to a MITRE command control department; you know, to make higher management at MITRE, you couldn't over specialize in those days. And so I had moved to a MITRE command control project that involved my moving to Germany and I was basically only involved in security sort of watching, from 1977 to 1981, when I left MITRE. The MITRE program with the Air Force dropped off precipitously with the cancellation, but we had made connections along the way with NSA and DARPA and so they managed to piece together a program that kept going after that. And in fact, the work that led to The Orange Book for DARPA and NSA was kicked off in that period, as well as helping manage some of the big, multilevel prototypes that were built in that era. KSOS [Kernelized Secure Operating System] was one; the SCOMP [Secure Communications Processor] was another. So MITRE stayed involved with those things, working for I think DARPA and NSA, rather than the Air Force contracts.

Yost: Can you describe your association with Project Guardian?

Lipner: Well, Guardian was the project that was cancelled in 1976-1977. We built the Air Force Data Service Center Multics, and that was deployed and went operational. But that was sort of adding the mandatory security model onto the existing Multics operation and patching the holes. We had this notion of a high assurance system, real reference monitor, minimized, always invoked, and self-protecting. And to get a high degree of assurance for that we felt that you had to have a lot less code than even in the Multics operating system kernel, or Multics operating system. And so Guardian started as a project to figure out what would that minimal operating system kernel look like? How would you restructure the rest of the system to work around it and still be Multics? And that involved a set of Ph.D. theses, master's theses mostly, work at M.I.T. on how to minimize and remove from the kernel various parts of the operating system. MITRE did some design work in that area. Honeywell did some, as well. SCOMP was started; supposed to be a front-end communication processor, a terminal concentrator, for that secure Multics system. I actually wrote the SCOMP RFP statement of work, 1974-75.

Yost: Can you discuss the SCOMP effort?

Lipner: Not in a lot more detail than that. Honeywell had this minicomputer operation, and so we said alright, so we'll build a secure minicomputer to do that (the front-end to the secure Multics) and then they specified hardware modifications; basically, they added paging and segmentation to what I suspect was a 16-bit minicomputer. I wasn't managing that program directly. But that went on through twists and turns and survived with, I guess, DARPA funding after Guardian was cancelled.

Yost: The Navy was also involved with SCOMP, weren't they?

Lipner: I don't recall that they were, but that doesn't mean they weren't. I mean, after Guardian was cancelled, all the contractors were trying to figure out where they'd get funding. MITRE was trying to figure out where they could get funding and sustain a program. The people in the government who believed that some of this was important were trying to patch together a program.

Yost: Do you recall implementation of SCOMP? Where within DOD it was implemented?

Lipner: No. That was after I was off the programs.

Yost: So in 1981, you left MITRE to join DEC; can you talk about that transition?

Lipner: Yes. I’d come back from Germany and then was running the tactical command control department, or one of the big programs in tactical command control at MITRE. The picture over there. Basically acquiring a tactical, deployable version of SAGE. And, you know, it wasn’t especially what I was interested in doing; and I kept in touch with my friends in the security business. So Karger had gone to work for DEC; he had left the Air Force, gone to work for DEC; and they were trying to build a security group in corporate research that was advocated by their government marketing group; but part of the corporate research and engineering organization. And they had a couple, three people and some projects, looking for somebody to run that. So they offered me a job as an engineering manager in 1981 and I took it. They had built a set of enhancements, Multics AIM-like add-ons to VMS. They were fooling with some network security stuff, and they were trying to figure out what to do next. And so I joined them in 1981. Did I send you that; the paper that Mary Ellen Zurko and I wrote for computer; for [*IEEE*] *Security and Privacy*, rather?

Yost: On DEC and [interrupted]

Lipner: On SVS. So the story’s sort of in there. I mean, Paul and I went to Oakland in 1981. We went to out dinner after the conference. We were talking about how do you get to high assurance. We thought that VMS product development would pick up the mandatory security enhancements that we had; that Paul and Joe Tardo had built for VMS and ship them in the product. And they eventually did, after a lot of twists and turns. But as with Multics, that was not a high assurance system. How do you get to high assurance? And so we said what can we do? And the logic is in the paper if you build your own system from scratch, then you've got to support all the applications and achieve ongoing compatibility with the mainstream operating system; and that's very difficult and costly; so you wind up driven to a virtual machine monitor. We said all right, what would it take to build a virtual machine monitor? We spent much of the next nine years doing that.

Yost: I have some questions on that but maybe we could take a quick break.

Lipner: Sure.

[BREAK IN INTERVIEW]

Yost: So we've just begun to talk about DEC and SVS. What was your impression of Digital Equipment Corporation and its commitment to computer security when you arrived?

Lipner: So, Digital; I mean; so the government marketing group, they sort of believed the people in the government who were telling them that this was going to be important. At that time, it was viewed as a government market problem. The government was a significant market segment for DEC but not an overwhelming one. The engineering organization, was more skeptical. They thought security was important, but lots of other things were too, probably a more balanced standpoint—security is important, performance is important, getting the next new processor out is important, making money is important. So security went into the pool of tradeoffs. And of course, any company, any commercial vendor; basically, has a lot of requirements and these tend to be threaded through the operating system group. And so all requirements funneled into the VMS group and they tried to balance them off and make the right decisions.

Yost: Besides you and Paul Karger, who else was a part of the development team?

Lipner: Initially, there was a fellow named Joe Tardo. He did a Ph.D. somewhere in California and he was trying to do some formal verification and crypto stuff. He moved into the networking group maybe a year or two after I went there. And then, there's a fellow named Drew Mason, who we drafted; borrowed from one of the PDP-11 operating system groups; and he came to work for us. And then a fellow named Cliff Kahn, and a woman named Sarah Thigpen, and Tim Leonard, who did the; he was part of the VAX architecture group. He did the prototype microcode. That's where we started the effort. The SVS group started with Paul and me actually on the team; Drew, sort of on loan for a while; and then grew to include those other people. And then when we said all right, we have a running prototype, we're now actually going to build a product it grew to like 40 or 50 people and when the project was cancelled and the Oakland paper was written, Paul listed everybody on the team as coauthors except me, because he was very angry with me. And so you can go through that sort of ending list of contributors and I can name some of the key, the other key contributors.

Yost: Was there any sense that within DEC, this system could have possibilities in the commercial market, applications beyond the government marketing group?

Lipner: Well, those of us in the development group thought it could. I think people across the company were unsure.

Yost: What were the key influences of past computer security work to you, Paul, and others in visioning VAX SDS?

Lipner: The Anderson Report, Bell-LaPadula, Roger’s work at the Naval Postgraduate School, this funny paper I wrote in 1971 talking about virtual machine monitors. We had looked at virtual machine monitors several times at MITRE. Those, maybe, to an extent, Popek. You know, the Popek kernel is a VMM kernel for the PDP-11 that was built at UCLA. We didn’t know much about KVM at the time, but we knew about the project. I think those were probably the key influences.

Yost: And in seeking a top trusted criteria rating, I believe it was 1983 that the first version of The Orange Book came out, is that right?

Lipner: That’s right.

Yost: At what point did it evolve from a research to a development effort?

Lipner: I think we formally made the transition and said all right, we're actually going to try to make a product out of this, in 1984; and probably we got—I'm not sure—we'd been talking to NSA under The Orange Book and precursor programs all along and I think we got an initial evaluation team assigned in 1984.

Yost: From the start, was the plan to build an A1?

Lipner: Pretty much, yes. We were going to minimize the trusted code, so it was either going to be B3 or A1. And we felt like if you're going to do a B3 and do all that effort, you might as well do the formal verification, too.

Yost: Had you had direct involvement with putting together The Orange Book, were you a consultant or an advisor, at any time.

Lipner: Yes, if you've looked at The Orange Book, you know my name is in there as an acknowledgement. Basically, the development of The Orange Book was done by government; NSA people and government contractors; MITRE; that was probably before Aerospace was involved in security so it's probably just MITRE. But then the people who were developing The Orange Book reached out to some of us in industry who had been involved in security research work through the 1970s and who they felt could provide perspective on what would or wouldn't work; and how things ought to be structured and evaluated. So I was part of that group of people that got drafts for review, commented on drafts, provided input on various ideas.

Yost: And when you embarked on a product development effort, were you aware of other efforts to build A1 systems at that time?

Lipner: We kind of presumed, I think, that the efforts that were ongoing, the research or prototype efforts that were ongoing, would also try to transform themselves into A1 products. So SCOMP, which was still at Honeywell; KSOS, which was at Ford Aerospace; I think KVM-370 was probably finished and dead by then; but we probably assumed it would go forward.

Yost: What did you see as the greatest technical challenge in developing SVS?

Lipner: Performance. When we started, we knew that performance was going to be the toughie. That, and a couple of things we underestimated, which we learned about as we went. The security evaluation and hardware support were things that sort of bit us, to varying degrees along the way.

Yost: Could you elaborate on each those?

Lipner: Sure. Security evaluation, I don’t think we realized how awful covert channels were going to wind up being. The word “nightmare” comes to mind; or maybe “death of a thousand cuts." (Laughs.) Okay? Just bad. Covert storage channels are easy. I mean, you're building a minimized system, and you're putting all the objects, you put classifications on them, you're done. But covert timing channels in a shared system, my God! Just horrible! We never; we did some innovative work, and it was brilliant. But the answer is it didn't matter. So that was one problem.

Yost: It didn't matter in the sense that that was too tough to overcome?

Lipner: There were still covert side channels. (Laughs.) And then evaluation; the only way anybody can build a high assurance system that's compatible with existing applications is to build a virtual machine monitor. Well, this was back in the timesharing days. You're going to build a virtual machine monitor, then you're going to run copies of the standard operating system in the virtual machines, and you're going to have to put multiple users on each of those copies or else the overhead of virtualization is going to kill you. And so that overlaps into performance, but that performance choice got us into a prolonged argument with the evaluation center about whether it did or didn't have A1 discretionary access control because the discretionary access control in the virtual machine at a single security level, is being enforced by the non-A1 off-the-shelf operating system. And we argued back and forth about that and I think we wasted time trying to come up with solutions and arguments and alternatives. We wound up kind of agreeing to ignore the problem, which is something that often happens with evaluations. What that means is that we documented how you could use the system with one user per virtual machine, and we knew that end users wouldn't use it that way.

Yost: Can you discuss the process and kind of the structure for the evaluation process?

Lipner: For the evaluation process?

Yost: To get A1 certification, is it the National Computer Security Center that's doing the evaluation?

Lipner: Yes.

Yost: Are they doing it themselves or is it contracted out?

Lipner: It's a mixture; I mean, it was government funded and it was a mixture of NSA and non-profit employees and independent consultants to non-profits. So MITRE, IDA, Aerospace—we never had anybody in from Aerospace that got in—actually, we did, I take that back; Aerospace, and then independent consultants who were free of conflict of interest, and subcontracting to one of them. And the mechanics were that you'd take The Orange Book, and you'd come up with an idea, and you'd sit down at an informal meeting with an evaluation team and you'd have a discussion, and they'd provide feedback and you'd go away and follow-up on the feedback; and eventually, you produce the system and you shared it with the evaluation team, including documentation, source code, formal specs, formal verification results. We had people onsite from a couple of evaluation teams, for a while, basically sitting in with our development group and just trying to understand what we were doing and how we were doing it. So that was the way the process worked. And we never got to the penetration test. They'd eventually take a finished copy of the system and do a penetration test on it. You know, walk through your design docs, and your source code, and so on.

Yost: In your co-written article on the lessons from VAX SVS, you stated that the effectiveness of the discretionary access controls proved to be the only major area of disagreement between your development team and the evaluators. Can you elaborate on that?

Lipner: That's the issue I talked about, multi user.

Yost: Okay. And you talked about security evaluation in some depth. Did you have anything to add about hardware problems?

Lipner: Yes. I said performance and security evaluation and hardware. Hardware, when we started in 1981, 1982, 1983, VAXes were micro coded processors. Well, back up. You know about Popek and Goldberg and virtualization? Okay. So the VAX; the vanilla VAX processor is not virtualizable and you couldn't sort of hack your way around that, and so we had to modify the VAX processor. The VAXes, up through the 8800 series, the processor logic was implemented in microcode. So you talk to the processor group; they build a special micro code load; you run that and all of a sudden the return instruction is privileged and life is good. But then; and there were a couple of other hacks and micro code changes; and we knew we wanted to minimize them. We actually were part of the group that was responsible for the VAX architecture for most of the time that we were working on the project. And so, you know, it was easy and straightforward to figure out what things made sense to do and were doable in the micro coded VAXes. But by the late 1980s we were starting to build microprocessor VAXes, and they were also micro programmed, but now, the micro code was on the chip and so you had to basically produce a separate chip—a separate item with your micro code in it. So now somehow you've got this separate production run of chips and then you've got a separate production run of boards, because you've got to have something to populate the chips onto. To sell,, just the logistics were awful. The project was cancelled before all that awfulness came home to roost. But by 1988, 1989, 1990 we knew it was going to be bad.

Yost: There were test customers?

Lipner: There were field test customers.

Yost: Can you say who those were?

Lipner: I don't think I want to do that, okay? There were about 12, 15 of them—government agencies—U.S. and countries that we could export to; and big defense contractors. And the feedback was positive; don't know whether I put that in the paper or not , there was a rumor that one of them actually took the Beta test version and ran it operational multilevel for a while. Which I wouldn’t have done, if it’d been me, but it was their data.

Yost: Can you give me a sense of the financial commitment that DEC made to this effort? What was the budget?

Lipner: I don’t think that could be sensitive anymore; it was; you know, the total cost probably in the range of $20 million, over 10 years.

Yost: How does that compare to other efforts for high assurance systems in the 1980s?

Lipner: I don’t know. Trying to remember what the Guardian budget was projected to be, and I don’t. That, of course, was 1970s; probably a little less than $20 million. And I have no idea how that compared to what other people were spending. I suspect some spent more but that’s just a guess; depended on what they did. Actually I was talking to a colleague about another effort, and I think he said they spent more than $20 million on their effort for comparable time, not quite a comparable system but same sort of environment.

Yost: Are you aware of anyone who did research and published on the economics of developing high assurance and evaluation of the cost of developing such systems versus the potential markets for such systems?

Lipner: No. You mean then, or now, or anytime?

Yost: In the 1980s or 1990s.

Lipner: There was an English or Irish researcher who did some work on history, formal verification, and so on, and that might include some discussion of economics. I am unlikely to think of his name but I believe Carl Landwehr might have worked with him and remember.

.

Yost: VMM, the security kernel for the VAX architecture, that you presented at the IEEE Symposium on Security and Privacy was awarded best paper. In your opinion, what was the longer term impact of lessons from the VAX SVS and to the understanding of high assurance computing?

Lipner: Well, hard to say. I worked at Trusted Information Systems in the 1990s; 1994 to 1997, when they were doing the TMach project. That was not a VMM, it was an attempt to build basically a kernel and emulator whose interface would be a secured UNIX. And it took a somewhat different approach and I don't know that there were any particular lessons learned from SVS that were reflected there. I wasn't part of that project team so I didn't have full visibility. You know it crashed and burned, too. They were doing it with government money. And then otherwise, I don't know; that's pretty much the only effort I know of in the 1990s. I've read about other projects, sort of research high assurance systems. I was at a NATO Information Assurance Symposium last year and there's some French company that sells a high assurance virtual machine monitor for the PC architecture. (Laughs.) I was talking to this guy at the booth and I asked him a couple of questions. I asked him if he'd read the paper by Karger and he says, "oh yes, Karger. That is a very good paper. We're familiar with that." So maybe somebody picked it up somewhere. But not a lot.

Yost: Can you talk about the contributions of Peter Conklin, Paul Karger, and Andrew Mason to the project?

Lipner: Yes. So Peter—I don't know how we got hooked up with him—when we did the design analysis for SVS I was very new to the company and I don't know quite know how we got connected to Peter. But he was the original VMS architect and he wound up being the Alpha architecture person; and he was very senior and a very good engineer. He just focused on what do you do, I mean, good technical guy. But what he helped us with was engineering process; what are you trying to do, what's the simplest way; to be very clear about what you're trying to do and then focus on the simplest, most effective way of doing what you're trying to do and don't bring any extra baggage. He was super helpful. Paul drove a lot–essentially all – of the detailed design of how do we build a high assurance VMM. He did all the digging into the past art of virtualization; what are the ways to build a memory manager; what are the ways to make the VAX processor work? And so on. Paul was responsible for memory management and file system in the prototype phase. And Drew was responsible for schedulers and, I think, probably I/O.

And that was building a prototype to run virtual VAXes. And that was basically what we did. Paul was very much the visionary and had a view of the entire system through that prototype period. Paul went to graduate school at Cambridge in 1984 and Drew, at that point, took over as the senior architect, senior visionary. You know, keep the entire system in mind role; and drove that through to where we had a system that we could send to field test. And Peter, basically, helped us make some really key early decisions; sort of dropped in and out to see how we were doing. But his contributions to the project were probably all made by the end of 1981, but they were super, super important. If we'd tried to emulate VAX Unibus I/O in the virtual machine monitor we still wouldn't be done.

Yost: What were the primary reasons that DEC terminated the project?

Lipner: Not enough people wanted to buy it. So the reason you build a VMM is to minimize your development and maximize your compatibility with what people want, with application software. PCs were an important factor by the late 1980s but there was still a distinction between the PC and the high-end workstation market. And so if we said well, alright, our target is the high-end workstation market, we still needed graphics, a graphical user interface, and networking, you know, beyond what we were able to do with the asynchronous DECnet hack. And to build a multilevel secure graphics package, and to build secure networking, that would have been essentially another research project of the scale of the covert channel project or bigger. So, at that point, we would have had to pick a target; that is what networking; what graphics; go off and design and build that. By that time, the graphic workstations would've had better networking and better graphics, so we would've been back into the catching up trap that the VMM approach was designed to avoid. It might have converged, we might have been able to do something that would've achieved currency or parity. But if we had, it probably would not have been 'til the late 1990s. You know, DEC was imploding for reasons that didn't have anything to do with the security projects. It was just clear to me that we weren't going to get there. If we had shipped SVS we would not have been able to continue it. From the standpoint of continuing the project, I probably could have made the decision to go ahead and sell it to customers. I wouldn't have sold many, but it would've been out there and then we would have been obligated to support it. That would have kept the product going but it would have been irresponsible to the business. And so, from a business perspective, the right thing to do was to pull the plug.

Yost: Supporting it would have been a major ongoing effort.

Lipner: Yes, or at least a significant ongoing effort. I didn't even mention the hardware; the microprocessor and board issues that I talked about earlier, but that would've been a cost. If that cost had driven revenue up then you might have said we'll do that; but you wouldn't have done that without the networking and graphics investments. So probably we would've had to build a chip; used that version to sell a few as time sharing systems; tried to build the graphics and networking; and then build another chip that we actually would've sold the graphics and networking on. I think top management would have just blanched at the level of financial commitment we were asking them to make. Couldn't have happened.

Yost: Some long term computer security researchers we've spoken with have likened perspectives or approaches to computer security almost to be like religions, and high assurance being one of those that has its diehard supporters. Do you think this is accurate and did you see yourself at the time as kind of a stalwart supporter of high assurance? And that was what computer security was all about?

Lipner: Oh, probably. I mean, I certainly; yes. Technically, I believed we really did some great theoretical work in the 1980s but you've got to decide. Are you going to do great theoretical work or are you going to make a difference. And at the end of the day, I think I did the right thing for the DEC business in making that decision to pull the plug. If you could figure out how to solve the problems of making products, making systems as secure as we claimed to be making, that would be great but I don't know. I mean, to the best of my knowledge, none of those high assurance systems really was exposed to the sorts of attacks that we see at Black Hat every year, or DEFCON. What we know about the Bell-LaPadula model and covert channels is not encouraging. Hard problems. People today talk about covert channels, and they mean inferring the key in another VM that's just doing a cryptographic computation innocently. We were talking about covert channels where you have a malicious piece of software in another VM trying to signal you. Blocking that is a hard problem.

Yost: And has there been work done subsequently that has significantly addressed that problem?

Lipner: There may have been but I haven’t seen anything. (Laughs.)

Yost: So you left DEC and returned to the MITRE Corporation?

Lipner: Returned to MITRE for a couple of years. Didn’t work on security, except sort of peripherally. Went to TIS in 1994. Ran the firewall business there for three years.

Yost: When you arrived at TIS can you describe the corporation? Steve Walker started that, didn’t he?

Lipner: Right. And it was very much Steve’s company; it was Steve’s company all the way through. And they had the TMach Project, which was an NSA contract. And then they had a bunch of other government projects; some odds and ends from the Navy, I think; and some subcontracts from big government. And then a lot of DARPA research.

Yost: Roughly, what was the size of the company, how many people worked there?

Lipner: About a hundred people.

Yost: A hundred people.

Lipner: Yes, just a hundred.

Yost: And you worked on the firewall technology?

Lipner: Yes.

Yost: Can you discuss that project?

Lipner: Marcus Ranum and Fred Avolio had come from DEC; and DEC had built a firewall called SEAL [Secure External Access Link] and TIS got a contract from DARPA. I believe it was connected with putting the White House on the internet. But they got a contract to look at network security research, and so Marcus built a better, or a different firewall as a C program that ran on UNIX. And TIS released that for free, as a free download on the Internet. It was a proxy firewall and we had proxies for the common applications, TELNET, FTP. We didn't have the HTTP proxy until later. And so we released the thing as a free download, and companies started to come to TIS and say well, that's fine, but I don't want to build it myself. You come and install it for me. So we started a consulting business doing that for a fixed price. And then we said alright, we'll make it a product. And Steve suggested we just pre-package it on a box, on a PC; ran UNIX on a PC. And so we sold those for a while; and then we started selling software kits for some major UNIX distributions. That turned out to be a pretty successful business.

Yost: So the company had both significant business in the government system area, but also was selling to corporations and the private sector with a firewall.

Lipner: Yes. Firewall was used heavily by some government agencies, government contractors, private sector. But we just sold it commercially to anybody who wanted to buy it.

Yost: Who were the primary competitors in the firewall business at that time?

Lipner: Well, CheckPoint was the one that was a competitor, and survived. And then there was a company called Raptor; a company called V-1 that started out re-selling our firewall technology; there was a company in Canada; those are probably the key ones. We were too slow to build a graphical user interface and we probably didn't do enough marketing soon enough. Oh, whatchamacallit, Secure Computing, which was another government contractor turned commercial. It's funny, Network Associates bought TIS for the firewall business, and then it eventually sold the firewall business to Secure Computing and then McAfee, the descendant of Network Associates bought Secure Computing. I don't know where all that—the TIS firewall business—stands, whether any of that still survives.

Yost: Was all your work on the firewall business at TIS?

Lipner: Essentially all; I did some odds and ends of government consulting.

Yost: Did you have any involvement in Trusted Xenix?

Lipner: No. I'm trying to remember whether I was responsible for it for a brief period but I don't think I was.

Yost: In doing a little bit of research on TIS, I noticed David Bell, and Marv Schaefer, Steve Crocker, Carl Ellison and other luminaries were all there, a lot of the star talent of early computer security research was there. Were there any other companies with such a list of prominent contributors to computer security?

Lipner: Maybe Secure Computing, where Earl was the star there. There's probably the list in there of other people, more spread out, I think.

Yost: Then in May, 1997 you left TIS to join MITRETech . . .

Lipner: MITRETech, yes.

Yost: Did that have any connection with MITRE?

Lipner: MITRE Tech is a non-profit; it calls itself Noblis, now. It's a non-profit; spinoff from MITRE. It took the non-defense government work that MITRE had been doing, and allowed, basically; MITRE to cut itself back to what’s called a Federally Funded Research and Development Center, which is a special designation in the federal contracting world that has certain preferences from a procurements perspective, but also certain limitations. They had accumulated some other work outside that FFRDC space, and I think they were taking some heat for that. So they spun it all off to MITRETech. MITRETech took some of the technology that MITRE had developed, spun it off into a profit-making company called Concept 5 that, I think, disappeared; don’t know when. And MITRETech, as Noblis, still exists; they’re not an FFRDC I don’t believe. They’re a nonprofit government contractor and I think they have some work for various government agencies.

Yost: And what lead to your leaving in 1999 to become Lead Program Manager for Microsoft security response team?

Lipner: Well, you know, I was running a division at MITRETech, and it was about 100 people. MITRETech was funny; I mean, they weren’t an FFRDC, which would have gotten them procurement preferences, but they weren’t willing to compete which they’d have to do to be successful without the preferences. I didn’t understand how that business model was going to work after I got into it and saw what they were trying to do. So actually, it was funny, a friend of mine who I worked with at DEC had wound up at Microsoft. Wanted to move back east; came to work for me at MITRETech for a while; and then left and went back to work for Microsoft; turned around and said, you ought to come to work for Microsoft. And so I said well, I'll talk to them. They had the Security Response Center job open and that sounded sort of exciting. And it was.

Yost: 2002, I understand Microsoft made a big security push. Can you talk about that?

Lipner: Yes. We had the awful worms of 2001 and so Craig Mundie, who was a CTO here—Chief Research and Strategy Officer now—Craig and Bill worked together to pull together this notion of trustworthy computing and basically, a commitment to trying to do things right, from a security and privacy perspective especially. And Bill released that statement, that commitment, in early 2002. In parallel with that, some of us were trying to figure out well, what can we do to make a big difference to Windows soon? And we came up with the idea of stopping development and saying all right, for some defined period, nobody's allowed to work on anything but security. And so I took that to my manager, and his manager, and then his manager; making more and more detailed proposals, never being told "no." Being told to come back with more details. About the third meeting, discussions got to exactly when would we stop and how would we organize what we're going to do. So we stopped all development in February and March of 2002, on Windows. And that was sort of the birth of the big change in Microsoft commitment to doing secure development. And it's not A1 systems, and the people who have the religion —to quote your question—think I sold my soul. But we made a big difference.

Yost: What do you see as the primary accomplishments of the security development life cycle at Microsoft?

Lipner: It gives us a way to build software that people will use, and to drive the security vulnerability rate down, and susceptibility to attack down, in an effective way. It's not perfect but we're able to build products that are more secure than they would be otherwise; much more secure than they would be otherwise. And still build products that are competitive. Anecdotally, Scott Charney, who's our vice president here, talks about how in late 2002-2003, no Microsoft executive could have any conversation with any enterprise customer about anything but security. We don't have those conversations anymore because it's not something that people feel they need to complain about.

Yost: Has Microsoft built its security capabilities primarily internally or have there been significant acquisitions?

Lipner: Big acquisition, before I got here, was a company called, I think it was, Intrinsa. And that was a code analysis, static analysis company. And that's the basis for— I think the name is right—that's the basis for some of our static analysis tools. Other than that, not a lot of acquisition. When I talk about security at Microsoft, I'm talking about building products securely. Michael Howard, co-author of our book, talks about building secure products and building security products. I worry about building secure products, not building security products. We do that too, but (pause)

Yost: Building the security into the products is the focus.

Lipner: Right. Operating system, database system, Office, and so on.

Yost: Are there topics I haven't covered, questions I haven't asked that you think are important to understanding the history of computer security and your role in it?

Lipner: Twenty years from now if you ask somebody about what I did—forecasting is hard, especially about the future—but 20 years from now if you ask people what I did in my career, I think what they're going to remember is my time at Microsoft and the SDL, and not the A1 work. The A1 work and the stuff around it will probably be a footnote. That may be right or wrong, but that's the way it is. I mean, we have companies; Adobe, Cisco, EMC, that have taken the SDL work, process control companies, utilities have taken the work we did on the SDL and picked it up and made it part of their mainstream development process. So that's not the quest for perfect security but it's improving security and doing that in a way that people use. And I think that's much more important in the real world than the high assurance research. Research is interesting, but we haven't figured out how to make all those research results make a difference.

Yost: Were there important influences outside of Microsoft, in how you; or rather you and a team of two others, developed the SDL [Security Development Lifecycle]?

Lipner: Well, SDL; Michael's and my names are on the book. And Michael and a fellow named Eric Bidstrup and I did the development of the first version. But that was a reflection of the experiences of the security pushes and stuff that we had done before. And threat modeling came from a couple of other folks; can't remember their names; we published the first threat modeling paper on the SDL blog a few years ago but we're like on our fourth or fifth version of threat modeling. Finally have one that is effective. When you ask thousands of developers to do it; you know, we update SDL every year or so. And so what goes in is based on the experience of development groups that either find new tools to drive vulnerabilities out, or find new attacks that you need tools to detect, or find new ways that make it harder to exploit vulnerabilities that remain. And so it's sort of a living thing. The people all the way down this hall, and most of the ones on that other one, are involved either in developing SDL or helping product groups apply it, or building the automatic tooling that enables them to apply it.

Yost: Thank you very much. This has been extremely helpful.

Lipner: Thank you, Jeff.