Software Assurance: A Master's Level Curriculum
Transcript

Part 1: Why Do We Need This Curriculum?

Julia Allen: Welcome to CERT's Podcast Series: Security For Business Leaders. The CERT program is part of the Software Engineering Institute, a federally funded research and development center at Carnegie Mellon University in Pittsburgh, Pennsylvania.

You can find out more about us at cert.org. Show notes for today's conversation are available at the podcast website. My name is Julia Allen. I'm a senior researcher at CERT working on operational resilience and software assurance.

Today three of my colleagues and I will be discussing CERT's new Master of Software Assurance curriculum the need we hope that it satisfies, its content, and how academic institutions can take advantage of it. So let me introduce you to my three colleagues. First up joining me today is Nancy Mead who's our fearless leader on the curriculum development team. Welcome back, Nancy. Glad to have you with us.

Nancy Mead: Hey Julia. It's great to be with you again.

Julia Allen: And Tom Hilburn, our Professor Emeritus for Software Engineering at Embry-Riddle Aeronautical University. Thanks for joining us Tom.

Tom Hilburn: This is great. Glad to be back with you guys.

Julia Allen: And last but certainly not least, Rick Linger who is a senior researcher at CERT focusing on next generation software engineering. It's good to have you on the podcast, Rick.

Rick Linger: Thanks, Julia. I'm very happy to be here.

Julia Allen: And in the interest of full disclosure, I also participated as a member of the curriculum team but I'm clearly serving as moderator for today's podcast.

So Tom, I think for the benefit of our listeners, why don't we start out with one of the basics, which is in the scope or the body of the curriculum how do we define what we mean by "software assurance?"

Tom Hilburn: Sure Julia. We did a little study and research about common definitions and so on. One of the things we looked at was the Committee on National Security Systems. They had a definition.

We used that as a basis and extended it and we came up with this definition. I'll just read it out for you. "Application of technologies and processes to achieve a required level of confidence that software, systems, and services function in the intended manner, they are free from accidental or intentional vulnerabilities, they provide security capabilities appropriate to the threat environment, and they recover from intrusions and failures." That's it.

Julia Allen: Okay, and you said we elaborated on some of the tried and true definitive sources. Can you say a little bit about why we felt we needed a little bit more robust definition?

Tom Hilburn: Well first of all this was going out to a large community of people, but the focus again was on improving the practice of software assurance and so we felt we needed to cover some features that weren't fully defined in some of the research we did.

Copyright 20010 by Carnegie Mellon University

Julia Allen: Excellent. So Nancy, why don't you get us started on the next question and maybe, Rick, at an appropriate point feel free to join in. So why did we decide to develop or what is the purpose of this curriculum? And in your experience as you look across the community which you've been involved in for many, many years, the education community, why did you feel that we needed to initiate this work?

Nancy Mead: Julia, as you know, about five years ago we started to work with Department of Homeland Security to improve the way that people were developing software in the first place so that it would be more assured. And to that end we developed the Build Security In website, which provides advice to practitioners so that they can hopefully develop software that's more secure and assured in the first place.

What we realized was we needed to do more than that, and so we wanted to get the folks at an earlier place in the pipeline, probably when they were still involved in the educational process or maybe coming back to get a masters degree after they had already been out in the field for a while.

So it became pretty clear to us that we needed to impact not just practitioners who were in the workforce, but also improve education so that we could get folks with more skills, with more knowledge, to develop the kind of assured software that we would like to see. And that's what really motivated us to do this in part.

The second thing that motivated us was the work that we and others have done in developing curricula for Master of Software Engineering. Again, as you know, we did this work at the SEI about 20 years ago, and then more recently there was a graduate software engineering curriculum model that was developed. And so we thought this would be a good time to jump in and do some curriculum work in the software assurance area so that we could help our colleagues at universities, the faculty members primarily, to provide offerings that were more in tune with our desire to improve the practice of software assurance. And that was pretty much it in terms of motivation.

Julia Allen: Okay and Rick, I know that you were involved in helping us expand the definition that Tom talked about and trying to tease out and identify the areas of special and unique emphasis in curriculum. Can you say a little bit about that?

Rick Linger: Yes, I'd be happy to. There are a number of special and unique areas of study that emerged out of this and I think that they tend to reflect the realities of today's IT environment. For example, in this program we certainly emphasize software itself, but we also emphasize services because so much in the way of computing capability is obtained through services in today's world.

We emphasize software development but we also talk a great deal about acquisition because so much software and capability is acquired today through various environments and methods, COTS software and so on, service-oriented computing kinds of things, and cloud computing. So acquisition is really a key feature of this program.

Computing system security is really a hallmark of this program but in addition there's a correct functionality of software. And the idea here is that defective software can't be known to be secure and so the correct functioning of software relates both, not only to just to security but it also relates to the capability of an organization to have the right kinds of computing capabilities and have them be dependable and have confidence in them. So security and correct functionality go hand-in-hand.

Another area that received special emphasis is what we call software analytics. A great deal of assurance is going to be involved with, as I said, software that's been acquired or obtained in various ways and that wasn't necessarily developed by the organization. And so the ability to analyze software as it comes in and to make sure that it has the right security properties, the right

Copyright 20010 by Carnegie Mellon University

functionality this is really going to be a key skill. And it's I think one of the areas that really strongly distinguishes this master's program from some other programs in, say, software engineering or computer science.

And software analytics is involved with technology that can be applied to help understand and analyze software perhaps in source form, perhaps in binary form, to make sure that it has the right kinds of properties.

Another area of emphasis that emerged is system operations. It's not just enough to field a system that has hopefully the right security properties and capabilities, but you also have to monitor and assess how it's doing in operational use. And so this is another focus area in the program.

And then finally there's a need here to be able to produce auditable evidence of assurance. How do we know that a particular software system has the right security and functionality properties and is going to function properly in operational use? So the ability to produce rigorous evidence of assurance processes and outcomes we think is going to be very important in this program.

Julia Allen: Thanks, Rick, very much. So Nancy to continue setting the stage, you mentioned obviously the Department of Homeland Security but who were the principal players both in sponsoring the work? And if you could just let folks know our team, the team members who actually developed the curriculum.

Nancy Mead: Sure. The sponsor of the work, our primary contact point and promoter of the work, is Joe Jarzombeck with the Department of Homeland Security National Cyber Security Division. And he's been involved in the project or the set of projects almost as long as I have.

The participants in the development -- and actually if you look at it broadly, we had a wide spectrum. We had the development team and then we had some outside folks who were involved, including a set of invited reviewers. We had a public review and then early on we did a survey of a lot of leaders to try to find out what characteristics they were looking for in people who were going to be involved in software assurance in their organizations.

So that's kind of the backdrop; we had a whole lot of folks involved. But the primary players included myself, you of course, and Rick from the SEI; and in addition Jennifer Kent who was our editor here did a great job of taking some of our mumblings and turning it into nice-looking prose.

In Addition, we had Mark Ardis with the Stevens Institute of Technology. Mark was one of the original developers of the Master of Software Engineering Reference Curriculum way back around 1989 and 1990. So he's been involved in this area for a long time.

Tom Hilburn, who's on the call with us today, is at Embry-Riddle Aeronautical University. Tom and I worked together quite a bit over the years on software engineering. And with Tom at Embry-Riddle, is Andrew Kornecki who has really given us a lot of different perspectives because he's got an interest in control systems and in safety and so he brought a lot of that perspective into the team. And then finally Jim McDonald at Monmouth University. Like Tom, Jim was involved in some of the earlier work in software engineering education and training and so most of these collaborations go back quite a long time.

Part 2: Curriculum Knowledge Areas and Courses

Julia Allen: Excellent. Well that certainly serves as -- I thank all of you for providing the background and the motivation. And so let's turn our attention now to digging into the curriculum a little bit, talking about some of its content, some of the courses.

Copyright 20010 by Carnegie Mellon University

Rick, let me start with you. You had talked about the areas of special emphasis earlier and I know many of these end up being manifest in some of the courses and sources and knowledge areas that we developed. We considered a wide range of existing material. But could you say a little bit about, touch briefly on the knowledge areas that the curriculum covers just to set the stage? And then I'll be asking Tom to talk about the courses in a little bit more detail.

Rick Linger: Okay great. The way that this material was organized, it generally fell into two major categories. The first category is assurance process and management and there are four areas within that that I'll touch briefly on. And the second is assurance product and technology -- more of the technical side of things. So this break between a management and process perspective on the one hand and a product and technology focus on the other hand seemed like a natural partitioning of the subject matter. And as you'll see in a minute here when Tom talks more about the courses, these things map pretty well into the course content. So let me just say a couple of things about the first area, the process and management.

The first major topic here is assurance across life cycles. It's very important that we focus on all of the phases of life cycles from requirements and specifications through design, implementation, testing, evaluation, operational use. All of these activities have assurance components to them. And so we want to make sure that those assurance activities are integrated into each of those life cycle phases in an appropriate way.

The second area is risk management where we want to be concerned about understanding, identifying, classifying risks that information systems are subject to, analyzing potential impacts and consequences, understanding what methods are available for risk mitigation and so forth. So that seemed like a nice encapsulation of that aspect.

The third area is assurance management. We want to be concerned about how we make a business case for assurance. I mean obviously assurance is something that goes on in parallel with lots of other resource-consuming activities in system development and operation. So business case is important, cost-benefit models, ROI, and so forth. Another aspect to this is compliance justification. I mean there are emerging laws and regulations that are affecting how systems must be assured and what their performance needs to be, so those things are important as well.

And then finally, the fourth area in assurance process and management has to do with assessing assurance. And here we talk about how do you measure the level of assurance that you've been able to achieve, what measurement processes and frameworks can be applied to that purpose? We introduce notions of business survivability and continuity here. And we talk about monitoring systems to ensure that the assurance levels that we think we have were actually being obtained.

Julia Allen: So this last one, assurance assessment, really does tie back to your earlier comments about auditable evidence, correct?

Rick Linger: Yes, absolutely. That's a key part of it. And those four areas taken together form a package of process and management topics that we think are going to be very important from a business and organizational perspective.

And then the second major area as I mentioned, assurance product and technology -- there are three areas under that. And the first is system security assurance which, as I mentioned of course, is an absolute centerpiece of this work. In this area we're looking at analysis of threats, potential attack methods, techniques for defense. We're looking at how you design standard techniques such as access control and privilege management, authentication, and so on and so forth. And we also talk about ethics and integrity in how we deal with security issues in these systems, and how we deal with malicious content, and so on.

Copyright 20010 by Carnegie Mellon University

So system security is a key part of this. And then paralleling this from the functional side, another area is system functionality assurance. And this is where we are wanting to make sure, to the maximum extent possible, the systems are correct and functioning properly. We look at rigorous methods for requirements and specification design, verification testing kinds of activities.

And we introduce a topic area that we call assured software analytics where we talk about how to analyze a software component at a very detailed level. We actually talk about methods to transform programs from spaghetti logic into structured form for improved understandability. We talk about methods of functional analysis to abstract out functional and security properties of software and those kinds of ideas.

And then the final area under product and technology is system operational assurance. And this is where we move into questions of how do you monitor system operations? What monitoring technologies are available? How do you evaluate what you're learning from that monitoring? How do you ensure that the levels of assurance that you think you have are actually there and being experienced in system use? And how do you respond to adverse events in operational use? And how do you maintain business survivability? So those three areas comprise the assurance product and technology part of this program.

Julia Allen: Great, thank you very much, Rick, for that summary. Much appreciated.

So Tom, of course you know it's great to have all these sources that we considered, all this good input, chunked out into meaningful knowledge areas but what makes up a curriculum are courses. So could you just -- in the interest of time we're not going to describe all the courses but could you talk a little bit about how we evolved going from knowledge areas to courses and how those various content areas are packaged?

Tom Hilburn: Sure. These courses might be seen as the end products of the curriculum activity we went through and there are several components that influence a course. One is we very early on started talking about student outcomes. What do we want our students to be able to do when they finish a master's program in software assurance?

Parallel and interactive with that was the development of the knowledge areas that Rick just talked about and the outcomes and the knowledge areas are very closely related. If you read through the seven outcomes, they add more detail about what we expect students to be able to do in these various areas that Rick just discussed.

In addition, we had certain expected knowledge, prerequisite knowledge, in order to achieve the outcomes in the curriculum. You put these together and produced a curriculum architecture that basically consisted of two stems.

One would be to create a master's program that is focused on software assurance; we call it a stand- alone program. And for that we described nine courses, actually eight courses that covered the body of knowledge. They're very closely related to, some of the titles of those courses are very closely related to some of the topics that Rick mentioned. For instance, we have a course in assurance management, one, and assurance assessment, operational assurance, system security assurance, a course in software analytics.

We also have three development courses that go through the life cycle: requirements, architecture design, construction, testing, that sort of thing. And then to top it off, we have a capstone experience course. It's nine courses and that pretty much makes up a master's degree.

But we also realized that this is a difficult thing to start a new master's degree from scratch. Add nine to ten courses is a difficult thing for many. So we also had a subset of that, six courses, seven

Copyright 20010 by Carnegie Mellon University

with the capstone, that could become a track in an (under)graduate program. And basically we, in some cases, merge topics from the eight courses that cover the body of knowledge and in some cases -- obviously a master's program in, for instance, software engineering may have existing courses that cover some of the parts of the body of knowledge.

Part 3: What Students and Employers Can Expect; Getting Started

Julia Allen: Great. So Rick, back over to you. Tom has started to talk about student outcomes. I mean obviously the whole idea behind this, as Nancy said in her earlier remarks, is to prepare students to really engage in this type of work throughout their professional career.

So if you were going to hire someone coming out of this curriculum, just briefly, what would you expect them to have learned? To be able to demonstrate? What skills would you expect them to have?

Rick Linger: Yes, that's a really good question. We all know that modern organizations, whether they're in the private side of the economy or in government or whatever, are pretty much software-defined and software-enabled.

Software systems are just so ubiquitous today and so essential to operations. And this curriculum is designed to pull together the kinds of skills and capabilities that a person will need to deal with the security and the quality of those systems in a comprehensive way.

In a modern organization, software assurance issues cut across many parts of the organization; many people and processes are involved. And so a graduate of this program needs the capability to function well on the business side of things to look at management and assessment of assurance and across many people and processes. So that's one reason that we have this focus on the process and management side.

But a graduate here is also going to have the capability to get into the technical side of things and to understand system requirements and specifications, how they fit into the needs of the business, and to look at the system artifacts and requirement specifications, designs, and so on to assess quality and security there at a technical level.

And so I think that the way I look at this, a graduate of this program is going to come into an organization with the kinds of skills to take over all of those what are now pretty much disparate and typically dispersed activities in an organization and be able to coalesce them and aggregate them into a focal point and take responsibility for software assurance at an organizational level, both in terms of the business side and the technical side.

Julia Allen: Excellent, excellent. Well Tom the rubber always meets the road in terms of getting started -- like going from knowledge areas to courses, okay I've got courses I want to teach. But as you said, getting new courses into an existing curriculum and getting certainly a new curriculum at the master's level established in an academic institution is really tough. So if someone listens to this podcast, reads our report, learns about the curriculum and would like to get the ball rolling, what would you recommend as some good first places to start?

Tom Hilburn: Julia, the first thing I'd recommend is that you get a copy of the curriculum document and read it. I mean that's -- read through the outcomes, the body of knowledge, the curriculum architecture, and look at the course descriptions.

So get familiar with it also. I think that might motivate you to say "Ah, what are we doing in our curriculum? What could we do differently?" First of all, maybe you would like to achieve some of those outcomes in your curriculum if you're a faculty member. And the question is, “What are you

Copyright 20010 by Carnegie Mellon University

doing in your existing curriculum that supports that and what could you change in your curriculum to improve it?"

If some of those courses could be offered with no prerequisite courses in your current curriculum -- there is some prerequisite knowledge. You've got to have basic computing and software engineering capabilities and some awareness of security issues. That would be at the undergraduate level. And with that, we've got several courses that you could choose out of, that you could pick any one of those. For instance, Development I which has an emphasis on requirement -- problem definition, requirements, that sort of thing. And there are no -- we don't require any other courses out of the eight courses that we describe as a prerequisite for that. And there are several others in that area that you could introduce very early into your curriculum.

And then a second thing you might consider if you've got some room is to create a track. The other part which is more substantial is to create a new degree program. If you're interested in that we certainly would like to offer some help and we'd be glad to interact over that.

Julia Allen: Excellent, thank you, Tom. So Nancy a couple of questions for you as we come to our close. Clearly, there's a body of work that's been developed here which is only the beginning of this process of actually getting this new curriculum and some of the new courses out into our academic institutions. So what plans do you have and what plans does the team have for the future for the coming year?

Nancy Mead: Actually the team has quite a number of plans. When you engage in a large project like this, the first thing that you think of after you've completed the first phase of the project is, "Well, how will we get this established out in the field?" So our focus coming up is going to be on transition.

One of the things that Tom mentioned is the fact that you could offer an entire degree program in software assurance, or you could offer it as a specialization, for example, within a software engineering master's degree. One of the things that we want to look at is specializations within other degree programs. And we hope that we will see a specialization within an IT program and perhaps an adaptation of the curriculum work to international degree programs which look somewhat different from our degree programs here in the U.S. We've had interest from quite a number of folks both here in the U.S. and internationally in installing some of the courses and some of the specialization that we've tried to develop within the curricula.

Let's see, our second set of activities Are really in the nature of supporting folks who are trying to implement this. We are developing a set of course outlines that are more detailed than the discussion in the curriculum and we're hoping that that will be a first step toward providing a set of resources. Right now we're focusing on the basic resources of books and materials and videos and so on that could be supportive. But long term, we hope to see complete sets of course materials or perhaps even complete courses available to our constituents who are in education.

And then longer term, we want to drive this down further so there will probably be some thinking about what we do at the community college level, perhaps what we do in high schools. One thing that we didn't mention earlier because it wasn't really germane at that point in time was we've also looked at what a set of undergraduate courses in this area might look like if there were a concentration on software assurance.

So the current work includes both the Master of Software Assurance curriculum document and also a document that describes undergraduate courses. So those exist and then we have a whole slew of these future activities to try to engage folks and actually offering courses, specializations, and degree programs.

Copyright 20010 by Carnegie Mellon University

One thing that's going to be really important is outreach and so we're working on forming a network of educators who are interested in these offerings and going forward, trying to provide support and mentoring to them.

Julia Allen: Excellent. And certainly last but not least, where can our listeners learn more about this, find our various reports and resources that we've identified for the curriculum?

Nancy Mead: Great question. The curriculum documents themselves are available right now in two places. One is on the SEI website as technical reports. But more importantly on the CERT website, we have the two curriculum documents. Volume I is the Master of Software Assurance Reference Curriculum, and then Volume II is the undergraduate course outlines that I briefly referred to. In addition we've got a PowerPoint presentation that's available for download and that's primarily for raising awareness among faculty.

Julia Allen: Great. Well, Nancy, thank you so much for your time today, for pulling our team together to provide this comprehensive description of the curriculum. Sure do appreciate your being here.

Nancy Mead: Well, it was a pleasure and as usual the team did most of the work and I just sat back and enjoyed it so that's perfect.

Tom Hilburn: Not exactly.

Julia Allen: We know what's not true. And Tom thank you very much for joining us today.

Tom Hilburn: Yes, enjoyed it.

Julia Allen: And Rick, great to have you on the podcast and appreciate all the comprehensive description. Thank you.

Rick Linger: Thank you, Julia.

Copyright 20010 by Carnegie Mellon University