



*Announcer You're listening to the Sun Microsystems Podcast Network.*

*Announcer Welcome to another edition of [Innovating@Sun](#), with your host, Hal Stern. Today's topic: Solaris Security. And now, here's Hal Stern.*

**Hal Hello, and welcome to another episode of Innovating@Sun. I'm your host, Hal Stern, Vice President of Systems Engineering. And today I'm going to talk about Solaris security. My guests are Darren Moffat, Senior Staff Engineer, and Glenn Brunette, who is a Distinguished Engineer. Welcome to the show, and why don't you start by telling us about your background and what you do at Sun.**

**Darren** My background is, I've been with Sun for ten years. The last six and a half years I've been in the Solaris Security Technologies Group. Before that I spent some time in Sun Service. Answering customer calls and dealing with escalations. And also spent some time in Solaris Sustaining. My background before Sun, I worked for the Ministry of Defense in the UK. And I can't say much more than that. From security point of view, my interest started, like many people, at university. So a long-standing interest in security and crypto for me.

**Glenn** For me, I'm actually in my eighth year at Sun. I've been working in the information security space for about fifteen years in various roles in terms of system network administration, software engineering. And most recently IT consulting. These days I'm working as the Director of the Global Systems Engineering Security Office.

**Hal Great. I guess the starting point here is that we have added a lot of security features to Solaris 10. Continuing to build on things that have been in the basic operating system going back to 2.6 and coming through Solaris 8 and Solaris 9. Most recently, things that we used to only include in the Trusted Solaris version of the operating system have found their way into the base operating system. Why don't we start by looking at what are the core capabilities that are there?**

**Darren** One of the most obvious differences that Solaris 10 brings that was previously only in Trusted Solaris was the concept of least privilege. Least privilege is about only giving applications and processes what they need to do their job. The thing that was particularly innovative and unique about what we did this for Solaris 10 was we did it in ensuring a hundred percent backward compatibility. Because Solaris has this compatibility guarantee, and we couldn't introduce a feature like this and stop people's applications from running. Although we had this in previous Trusted Solaris releases, it wasn't a hundred percent compatible. It is now. One of the other key things that is very obvious that it came from Trusted Solaris and is still very much a Trusted feature is the concept of labeling. This is traditionally for your military customer background, but it applies equally well to financial services. Imagine a trader on a trading desk. You don't want him sending an instant message over an AOL network to his friend who works for another trader about what he's doing. But you might want to use instant messaging within your trading environment. So you need to be able to control the flow of the data based on who it's coming from and where it's going to. And we do this by labeling. Again, a Trusted Solaris feature that came into Solaris. Role-based access control is the third of the big ones here. It first appeared in Solaris 8. But in Solaris 10 we start to make really heavy use of it.

Hal Glenn, anything you want to add to that.

Glenn Yes, I'd actually like to mention an add to Darren's points. In addition to being backward-compatible, the beautiful thing about how privileges and RBAC and so forth have been used in Solaris, and Solaris 10 in particular, is that they were fully integrated from the start. So, even if you're talking about something like the Service Management facility, where you're using this facility to control service behavior, service properties, and so forth, the actual how services are started leverage things like Solaris privileges. Or how you grant people the ability to start, stop, enable, or configure services is controlled through RBAC. In this way, security has continued to be integrated throughout the operating system, and that's just one example.

**Hal So if you were to turn the conversation around now. Think about this in terms of our customers. Particularly our customers now who are faced with a larger volume of content being written or being posted back to their sites. It's not just them putting their information on the Web, but also engaging in a two-way conversation with their customers, how do they put these things into use? What are they worried about, and how do we take these basic features and build a more secure system out of them?**

Darren One of the things that is quite different about where we are now is the distinction between the inside your company safe ' behind your lovely safe firewall ' and outside is blurring. As you say, it's now a two-way conversation. You want to be able to build the things that are at the perimeter where you talk to your customers out of a solid foundation. And we hope that Solaris 10, with its security features, allows you to do that in a way that you couldn't do before. Trusted Solaris has been used by military organizations a lot for doing this type of sharing. Sharing different classifications of data. You can apply those same things with Solaris 10 now that we've got the Trusted Extensions functionality in there. It's mindset change to be able to use some of these features, but it will certainly allow you to do many more flexible things with sharing with your customers if you think about what your data means to you if it goes outside. Is it acceptable, is it not. And start to not think about a system that has firewalls that are all around the outside, but think about the flows of the data. To do that, labeling is one of the ways you can deal with that.

Glenn The other thing I would point out is that if you look at the security capabilities of Solaris 10 in their entirety, and look at how well they integrate and support one another, we have an opportunity to create really an unparalleled platform for the delivery of secure services. By this I mean, if you were to take for example a Solaris system. And you create a zone in it in which you will deploy your service. You can take that zone and you can harden it using the Solaris Secure by Default capabilities. Once you deploy your service in there, the service could be deployed within the Service Management framework, where you can start it up with reduced privileges. You can control who can start and stop and enable and disable that service. In addition to that, we have the core capabilities of Solaris that have come from many years. Things like preventing code from being executed on the stack. Being able to audit actions taken by those services. We have in Solaris 10 a firewall IP filter that can further restrict what comes in and out of that zone. So, as you build up these capabilities and you think about them in total, you see that you have built a very strong platform that allows you, based on your requirements, you can pick and choose what you need, but based on your requirements, you can deliver a very strong platform that will protect your services and data.

**Hal What I'm thinking out of this is that historically, and Darren you mentioned this, the view of having the crunchy shell, the firewall that just protects the perimeter, is really not sufficient anymore, especially as we're seeing more of an emphasis on writable content. More of an emphasis on two-way conversation. That we can't just think about the inside of the network and the outside of the network. There's really a lot more transitions to that barrier.**

Darren Absolutely, Hal. And as Glenn was pointing out, one of the things that we can do is use zones on Solaris to do that type of thing. You still need to deploy the applications that provide these services in a secure way. But it's all now about the flow of the data, so we've got to be able to control that. And if you're going to talk about controlling the flow of the data, you want to know where it's going to. So we've got things like auditing as well. But it's not just about what we've shipped in Solaris itself. We've got to look at all of Sun's products here. They work together. Being able to run Sun's Web server inside a zone. Being able to run the Application Server with Portal Server inside it. All sitting inside a zone. Or if you're running with the labeling functionality from Trusted Extensions, they may be running at a distinct label. That'll give you the ability to control where the data moves without having to get down to very nitty-gritty details about firewall rules. Instead, you think about what the classification of the data means to you. Is this safe to go to a customer? Did it originate from a customer? Can these two customers see what each other did? And that's some of the things that people couldn't do before with Solaris, but now with Solaris and Trusted Extensions included in it, they can start to do these things.

**Hal Darren, your commentary about working with containers and different parts of the system being able to see each other makes me think of what we're clearly seeing as an increase in emphasis on virtualization and virtualization technologies in the market. Being able to gain greater efficiency, or being able to consolidate multiple workloads on the same system. Glenn, I guess that raises a whole other host of security issues in terms of keeping things isolated that really need to be separate from each other, even if they are sharing physical infrastructure at some level.**

Glenn Absolutely. In fact, we still need to recall that there are different levels of risk. We need to evaluate risk related to systems application data on a case-by-case basis. There may be some applications that physically can't share the same hardware, software, switches, and so forth. But then there may be cases where we could use domains. Where we can use lightweight domains, or where we can use zones. Really, that's one of the beautiful aspects of Sun's portfolio is that we provide a wealth of options for people, depending on their levels of risk and tolerance for failure. In terms of service failure, hardware failure, software failure, things of that nature. We give them a variety of options based on what their particular risk profile might be.

**Hal OK. So as we take the different layers of security mechanism that we have, and emphasis on, as you say, Glenn, the risk modeling and the threat modeling, if you were to look at Solaris 10 from a market perspective, are we ahead? Are we behind? How do you stack ranking here in terms of the basic capabilities that we offer to system administrators and to people designing networks that they want to be secure?**

Darren I think in many ways we are ahead. And we were already ahead in some portfolios, because we had some stuff from Trusted Solaris, some unbundled products. And a lot of that functionality is now available along with your Solaris 10 license. Or free[?]. You no longer have to spend an expensive Trusted Solaris product and learn a lot of extra slightly different things. It's all just now part of Solaris. With things like the labeling, our heavy use of role-based access control. As Glenn mentioned earlier, fully integrated internal service management framework, we're ahead in terms of the technology. And getting that technology usable by customers to help secure their systems. We made a couple of other bold steps in Solaris 10 of cryptographically signing almost all the binaries that make up Solaris. This is prework for stuff that's coming along later to help us do secure execution. There's stuff that's still sitting dormant in Solaris today that we can do more with in security coming along.

Glenn I would agree. I think we actually are quite far ahead, and in fact, looking out at what our customers are doing today, many of them are still not effectively leveraging the technology that we had in Solaris 8 and 9. One of the challenges that we have is that we don't want to get too far ahead of the customer, creating new and

different technology, and them not using it. So one of the really strengths of Solaris 10, I think, is that we've started to really focus more on usability. We've taken many of the hardening recommendations, for example, that have existed in the industry for many years. And have been codified by tools like the Solaris Security Toolkit for several years as well. And we've built that into Solaris. We've made it a basically no opt[?]. All you do is select, yes, I want Secure by Default in Solaris 10. Solaris dot next, it will be the default. So that we take a lot of the complexity and a lot of the concern about what do I need to do, how do I need to do it, out of the equation by building security into the system itself.

Darren One of the other things that that allows us to do is, we don't get arguments between engineering and service and a customer about what the right thing for a security point of view is. Because by putting it into Solaris rather than it being some third-party's recommendation, it provides a lot more weight and value. Customers trust it. And the whole service organization, whether it be inside Sun or done by a third party, likes this a lot. They take the choices away. They take away the ambiguity by us building more and more into Solaris.

**Hal Last comment here. What's top of your mind, what are the cool things you're working on now. For both Glenn and Darren, how are you putting this stuff into practice?**

Darren One of the biggest problems we still have to solve for Solaris, even with what we've released so far, is security of data at rest. We have two projects running now to help with that. Both of these are actually OpenSolaris projects. One of the great advantages of doing that is we get to see customer feedback on this before we start writing the code. Make sure we've actually implemented what the customers really want. In a way that even with all the requirements gathered and we've done before, we've never been able to get this direct feedback. The two projects are adding encryption support to the ZFS file system. Which was part of Solaris 10. It was a very innovative new file system. And it was designed from the start to be able to support encryption. Adding crypto to it was quite a challenging thing. To understand what customers actually need. So, although it's not there now, the whole ZFS file system was designed with this in mind. They left us hooks to put these things in. And we're now going on and developing that stuff. It's going to be a phased delivery. We're doing a smaller project as well. More aimed at laptop use. To solve a similar problem in a simpler way, that will allow people to use legacy file systems as well. As I say, both of these are OpenSolaris projects. It's solving the bigger problem of security of the data at rest.

**Hal And Glenn?**

Glenn There's actually I think been three interesting projects that have happened. One has been completed, and the other two are in progress. The first one was around minimization. Minimization, as you recall, is the act of either removing or initially not installing software that you feel or is not needed, or the operating system to provide its services or for you to manage the system and support the system. This has been one security practice that has been employed in many sites over the years. One of the challenges we've had is understanding when that type of configuration is not supported. What are the rules around supportability when you start taking software and removing it from the operating system. One of the things we accomplished this year was developing a 'rules of engagement' document for support services that outlines, with feedback from engineering, from customers, from customer engineering, what are the rules. What are we trying to do. One of the benefits that we were able to get everyone on the same page, and now we have an official support structure for minimized configurations. There's actually an info doc that customers can view to understand what is support, what is not supported, so that they don't run into problems later on. The other projects that we're working on, I just completed some initial fuzz testing of Solaris. In the past, we've done code reviews, and many types of security audits around our code. But one of the things we haven't done is actively looked at fuzz testing, which in this

particular case, we were injecting all of the binaries in Solaris to random types of output, and seeing how they reacted. In many cases, we found that there were some problems for certain programs that would core dump due to a buffer overflow or something of that nature. One of the interesting notes about this work, though, is that although we did find some programs that need to be corrected, none of them were actually privileged programs. None of them were setuid or setgid. And none of them would've exposed a privilege escalation flaw in Solaris. The last thing that we're working on, and this is something we're working on jointly with a number of U.S. government agencies, is around Solaris security recommendations. Although we have implemented Secure by Default in Solaris, that's not where the story ends. Because when you have services, you have systems, you have operational requirements, there's often a need to configure Solaris, even if you have deployed Secure by Default, you may need to make additional adjustments. That is the reason why we have these additional security guides. There's many guides been published over the years from the Center for Internet Security. The NSA. NIST and other organizations. What we've done is tried to get many of these organizations together and develop a common set of security requirements for Solaris, and we're actively working on that and hope to have something published in the near future. More than that, we've taken the feedback that we've learned from this process, and we're working to integrate as many of those additional requirements as we can into future Solaris defaults where possible.

**Hal** Great. Certainly, as far as we come, there's always a new set of threats or a lot of work being done in this space we can always go and apply to the state of our software. Again, Darren Moffat, Senior Staff Engineer, Glenn Brunette, Distinguished Engineer. Both hard at work in the Solaris security space. I'd like to thank you for joining us today in this episode of Innovating at Sun. And I've been your host, Hal Stern.

*Announcer* You've been listening to *Innovating@Sun*. Join us next time for the latest in innovation from Sun Microsystems. Only on the Sun Microsystems Podcast Network.