

IBM Podcast

[MUSIC]

MATHENY: Welcome to this IBM podcast, Trust and Transparency, Managing Secure Outsource Relationships. I'm Angelique Matheny with IBM.

Security is always hung up on or been hung up by the weakest link. The complexity of today's systems has compounded the difficulty of defending them, not just because of the number of components but because of the many sources from which they're drawn.

Whether you're integrating an outsource or open sourced application or if you're interfacing with an outsource service provider, security must be one of the areas where there is a clear focus and detailed requirements.

In this podcast, Jack Danahy, Worldwide Security Executive for Rational, talks about some recent events where more security transparency would have helped and describes ways in which Rational can help to ensure the security and reputations of our clients. Hi, Jack. Welcome to the podcast. Thanks for joining us.

DANAHY: Thanks, Angelique, happy to be here.

MATHENY: Let's jump right in. Jack, what have you seen as new security problems as organizations outsource more components and services?

DANAHY: I think that what we're beginning to see is that sometimes a breakdown in communication is causing there to be a new set of security mistakes being made in terms of the way the people have either components themselves or services implemented.

The ongoing move towards more outsourced services as people take advantage of geographical disparity in cost, and sometimes as they try and find subject matter experts, which are different than the folks they have in house, they're ending up moving a lot of what used to be critical internal functions to folks who aren't necessarily as well versed in corporate culture or in the business of the companies that are engaging them.

And this can sometimes cause there to be a real breakdown in an understanding of the business purpose and sometimes the business priority in terms of the security of these applications as they're being constructed or as they're being managed.

Let me give you an example. Ordinarily within a particular industry or within a particular company, there's an

understanding of what types of data are important, how an application is likely to be used, maybe other kinds of applications that that particular service is going to touch.

And so from an internal perspective, there are a lot of judgment calls that get made that will sort of naturally be the right ones. There's an understanding of the way the systems are going to work together. There's an understanding why it's important and how it works. So typically, there is either internal or external judgments made about whether or not the application is going to be secure enough.

When I remove that function, when I remove that function of actually developing the software and perhaps developing the service away from that central base of common understanding, then I run the risk that there will be a disconnect: a disconnect in the way that people understand whether or not this application has to be secured in a particular way, and whether the data itself will be handled in a way that's appropriate to the people who are sharing it with me.

And so with this relationship, as this relationship grows with these outsource providers, new risks enter into it. And the risks really fall not on the organization necessarily developing it, but in the organization which is asking the development to be done, the organization which is

going to deploy it.

I'm going to give you an example from a long, long time ago which was a case that happened here in the U.S. called McPherson versus the Buick Motor Company. And to make a long story short, what happened was Buick built a very particular car and there was a wooden wheel that it used that it purchased from one of its providers.

And a gentleman was actually driving it, Mr. McPherson, was driving his car, and the wheel failed. And coincidentally enough he was on the way to the hospital with a sick friend.

But what ended up happening was the wheel itself failed, and it was an accident. And Mr. McPherson was injured. And so he sued. He sued Buick.

And what Buick said was, listen, we didn't make the wheel, among other things. We didn't make the wheel. And so, therefore, we can't be held responsible for the wheel's failure.

What the court came down and said was, listen, you were sort of the last stop in the line. You used the wheel. You understood that motoring itself, particularly back in this day and age, which is back in the early part of the century, 1916, roads were likely to be a little bit dangerous. And so you understood the risk better than anyone else that

McPherson would be under using your vehicle.

And you understood what the wheel was going to have to do, and what the stresses it would have to bear up under. And so therefore you, the organization which sold the vehicle, are responsible for all those different pieces, because you were sort of the last stop in the line. You were the last place where that could have been checked to see if it was okay.

As we extend this forward, we see numerous cases where there's data loss, where there's privacy challenges, et cetera, where people's individual information is sometimes lost, sometimes services are disabled.

And these things happen because they haven't necessarily managed that relationship, but they provide it well enough so that that provider understands exactly what they're supposed to be doing and what the security concerns are around the way in which they're going to get it done.

This communication back and forth is of really utmost impact on both of these kinds of organizations. What we find is that the organization doing the development typically benefits from having those kind of requirements up front. It's not that much harder to do things in a different way to make it more secure. It's just harder to do it after the

fact.

And for the organizations which are actually acquiring the service or the application itself, what they're going to see is a difficult choice between deploying something on time, which is insecure, as it gets delivered from an outsource provider.

Or perhaps the challenge of actually deciding, I'm going to slip my dates and I'm going to miss all these schedules is sometimes too tough for people to make that decision.

And so for everybody's sake, these kinds of new challenges are arising as we're moving more and more of this development and service provision to outsource people, it really relates to understanding the communication of that security and understanding of the business values that are being driven by this particular application.

MATHENY: You know, Jack, everyone is worried about cloud computing and security. Are you as concerned about security in the cloud?

DANAHY: For me, the most interesting part about cloud security is all the great questions that are being asked before it really takes off.

I mean, there's a lot of great cloud infrastructure out there. There are a lot of organizations working to understand better how cloud-related services can separate sometimes some of the more logistical functions away from organizations so they can focus on value additional, and the way in which centralized architectures and infrastructures can relieve the burden on some organizations who don't want to do all that work themselves.

And the thing I like about it is there are so many questions being asked about security as these systems were implemented. If I look back at what was going on, as we're all working in the Internet, and Internet security back in the mid-1990s, it was beginning to take off, everything was about feeds, and speeds, and how fast can it get connected, and how many different people would come and touch what I've built.

These days, that is better understood, but also because of those experiences, people understand the risks of having inadequate security as these systems roll out.

So personally I believe that as cloud infrastructures gather more momentum, and as you find them taking on more of an important role, central to the execution of business function for a lot of these large organizations, it's actually going to be more secure -- it's going to be more

secure for the same reason that consistency breeds security.

As organizations focus their talents on what they do best and allow cloud service providers to focus on the provision of hosting, or storage, or you name it in terms of a service, they will be very much focused on ensuring that that service itself secure in satisfaction of all of these new questions that are being asked as you and I sit here and talk today.

And I think that the outcome will be ultimately that the cloud may turn out to be even more secure than those individually deployed architectures and interfaces that we have seen to this point in time.

MATHENY: There's a long-running debate about the security about open source versus proprietary software and applications. Which do you think is more secure?

DANAHY: I love that question, because it speaks to the heart of what we're talking about today, which is sort of learning how to manage in some cases an outsourced relationship.

When I think about open source code, there's so much value there. Right? But the providers are sort of an outsourced company where you don't know anybody -- that an open source

package is written by really smart folks who understand what they're building, but two things, number one, they're not really building it for you.

They're building it according to a certain set of functional requirements. They're building it according to what they think is most important, what the market is sort of telling the pack is most important. And they're also building it according to their best practices, which may be different than your own.

And so, in this case it's sort of like an outsourcer, but it's an outsourcer where you don't even have the opportunity, necessarily, to ask them how they're building it or why they're building it.

Now, the second piece of it is there's no real sense of responsibility to you as an individual organization from the open source community. They are responsible for building the best thing that they can do along their own sight lines, along what they think is the most important challenges to solve, whether it's performance, functionality or security, you pick it. And those won't necessarily match up with all of the priorities of the organizations who intend to deploy it.

So the way I look at it, it's much the same as ensuring a

nice relationship between yourself and an outsource provider. It's about understanding what you're getting. And so in this way, if I stop and I think about the best practices that I'd use to integrate outsourced software packages or services inside my own organization, it's mostly about insight and understanding.

I want to learn what goes on inside an open source package.

I want to be able to take a look at it. I want to be able to read about experiences other folks have had with it, and I want to be able to learn for myself whether or not the way that it behaves and the way it does its job is going to be in keeping with the type of security or privacy or confidentiality which I intend to offer to my own clients, to my own prospects, perhaps to my internal users.

And so, is it more or less secure? It really is so contextual, right, that I don't believe that anyone would adhere to the fact that they are naturally more flawed or naturally incredibly less flawed than less open packages.

But I think that the fact that they are out there and that they're understandable, and I can look inside them, because typically the source code is available there as well, it's an opportunity in the open source community to learn more about the package before you decide to use it...

...to measure packages against one another and to ensure that whatever package you choose is one that's going to match up with the type of security concerns that you or your users are going to have moving forward.

So, again, I would just recommend that as organizations decide which to use, don't go by a question necessarily of is it more or less secure, take it on as part of your process, understand whether it's secure or not.

Understand whether it's secure enough for the purposes that you're going to put it to, because even in a single organization there may be many different contexts in which an individual application may or may not be secure enough for the purpose to which it's intended.

MATHENY: Jack, we're going to end with this question today. Have you seen anyone who really approaches security in the way that you describe, or is this just unrealistic?

DANAHY: It's funny, as we get closer to sort of the nuts and bolts of it, as we get down to individuals writing code, many times we have a lot of differing commentary on what is an appropriate focus on security? How do I manage security appropriately? How do I make these kind of decisions.

There's a lot of conversation that goes on. And so, sometimes what is really an earnest conversation around the right thing to do, along any number of axes can be seen as a lack of commitment. Right? But I don't believe that it is.

And I will tell you that one of the things we see, which is a relatively high level of support for the style of interaction, is in regulation. Whether it's new cyber security regulation, whether it's the new stuff that comes out of the utility providers, or whether it's standards bodies and financial services...

The organizations which are going to regulate relationships with our providers and the regulations that organizations will be under to protect our private information or our financial information, those regulations don't call out a differentiation between, if you build it yourself or if someone builds it for you.

They're very, very strict. They describe the responsibility of the organizations which are going to take in my information or handle my functions for me, and it describes how they will be held responsible. So I think at the highest level, you know, up to the congressional offices here in the United States, those folks are taking it very, very seriously.

And what ends up happening is that sense of responsibility, that sense of understanding that it is ultimately the responsibility of the organization which captures the data or provides the service to ensure that it's done securely, is driving all this good thinking further down inside the organization...

...and will drive a more consistent approach to understanding whether applications and services are secure enough from outside providers, whether it be designed internally, externally, really without regard for where it's coming from, but mostly with regard to the security that it's intended to provide.

MATHENY: Jack, as always, thank you so much for sharing your time today to discuss this podcast, Trust and Transparency, Managing Secure Outsource Relationships. We really appreciate your time today.

DANAHY: My pleasure.

MATHENY: That was Rational's Jack Danahy, Worldwide Security Executive. To share this podcast with your colleagues, or if you're interested in more podcasts like this one, check out the Rational Talks To You Podcast Page at www.ibm.com/rational/podcasts.

We'll include a link to the white paper titled, Trust But

Verify, How to Manage Risk and Outsource Applications. So,
check it out today. This has been an IBM podcast. I'm
Angelique Matheny. Thanks for listening. Keep tuning in as
Rational Talks To You.

IBM Podcast

[MUSIC] [END OF SEGMENT]