Controls for Monitoring the Security of Cloud Services
Transcript

Part 1: Rapid Elasticity of Resources

Julia Allen: Welcome to CERT's Podcast Series: Security for Business Leaders. The CERT program is part of 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 I'm pleased to welcome Jonathan Spring. Jono is a member of CERT's Network Situational Awareness Team.

And today he and I will be discussing how cloud computing users, given the growth in cloud services, how users can better ensure that their sensitive information is more protected and secure. And for our listeners' information, we do have an earlier podcast on the Upside and Downside of Security in the Cloud that we posted in June of 2009, that you might want to listen in to.

So, with no further ado, welcome Jono, glad to have you with us today.

Jono Spring: Thanks, Julia. I'm glad to be here.

Julia Allen: Okay, so just for a little stage set, or a little recap, could you tell our listeners the definition of cloud computing, the short version, and the three most typical service models that characterize cloud services?

Jono Spring: Sure. There are, of course, several definitions of cloud computing. I like to work with NIST's, both because it is pretty good and also it's a legally binding, for the US government sphere.

I'll do the short version, so we don't put anybody to sleep. And, at the risk of being too brief, there are five essential characteristics of cloud computing: on demand self-service, broad network access, pooling of resources, rapid elasticity of that provisioning, and service or resource monitoring by the provider.

There are three service models that people usually talk about: software as a service, platform as a service, or infrastructure as a service.

Julia Allen: Okay, so when you talk about rapid elasticity of provisioning resources -- that doesn't quite roll off the tongue -- can you just give a little example of what that might mean?

Jono Spring: Sure. So if you are operating an application in the cloud, and you want to change how many CPU cycles you're using from, let's say a billion to 10 billion, so you can do that in a second or two.

The cloud provider has those extra resources available and so the resources are elastic. They can grow and shrink very much and they can do so quickly, on demand, in an order of machine time, in seconds.

Julia Allen: So obviously that has a huge upside if you're using somebody else's environment and services as opposed to trying to do that in your own environment, correct?

Jono Spring: Yes. Since many people usually share these resources, even though one person might not be using them all at once, someone will be using them. So you have a good economy of scale in that the provider can buy a bunch of machines and someone will be using them but you don't have to know it.

So you don't have to buy those machines and so there's a much lower cost to get into the game of having a large pool of machines.

Julia Allen: Excellent, thank you for that example.

So as I was preparing for our conversation, you gave me two great recent articles that you've written for *IEEE Security and Privacy* magazine, and I'll be using that as a basis for a lot of our conversation today.

And you talk about controls, things that both service providers and users can do at each of seven layers of cloud deployment, and we'll get into what those seven layers are in a minute.

But before we launch into that level of detail, do you have some overall considerations or things that both providers and customers need to think about before jumping into a cloud service, some considerations?

Jono Spring: Sure. There are some things that will apply regardless of what service model you're going to be employing your cloud service as. So with all clouds, there's a departure from traditional situations in which the information owner is responsible for monitoring and auditing the hardware and the software that they're using.

This isn't possible in a cloud, a public cloud at least, where the customer is purchasing services from a external service provider. Because, at a minimum, the customer is outsourcing the operation and ownership of that hardware.

Jono Spring: So given this federation of control, the customer and the provider are both at somewhat of a security disadvantage because there's an interface between where the provider has control and the customer has control over something, and this produces of visibility gap. Where this gap exists, changes based on the service model but it must exist in cloud computing at some level.

And therefore the customer cannot technologically guarantee that there is no malicious process mediating this hardware access for their software. So whatever recommendations I have presented will help to minimize the impact of this problem, this gap in the visibility, but I don't believe that it can be eliminated in a public cloud.

Julia Allen: Okay, so it sounds like -- obviously the customer has to really vet their provider very diligently, because it sounds like, based on this gap, or this shared responsibility, shared visibility, they really have to trust one another, right?

Jono Spring: Right. Cloud computing comes down to a matter of trust. So if you can't trust your cloud provider to make sure that their hardware is not only run appropriately but also that it is secured from outside and insider maliciousness, then you'll be in trouble.

You would use, basically, contracts and service level agreements -- or SLAs you would colloquially say -- so SLAs and contracts are your primary instruments to ensure this adequate amount of trust, and the customer can enact an adequate amount of auditing and monitoring powers to ensure that trust from the provider.

Part 2: Controls for Facility, Network, Hardware, Virtualization, and Middleware Layers

Julia Allen: Great, great. So let's turn our attention to the seven layers, and the controls within those, which is the meat of our conversation today. So just briefly, can you tick off the seven layers and then we'll spend a little time on each of them?

Jono Spring: Sure. So the basic first three are (1) the facility, (2) the network, and (3) the hardware. Then you've got, going up the stack, (4) virtualization infrastructure and the operating system as sort of one unit, (5) the platform architecture, which includes middleware as well as APIs, utilities, libraries. And then furthermore, you've got (6) the application layer and then (7) the user layer.

Julia Allen: Okay, so we talked a little bit about the three service models. And I know from your articles and other reading I've done that the service model really dictates which controls the provider implements and which controls the customer implements. But independent of service model considerations, let's just walk through -- and maybe if you can give some examples of a couple of the controls at each layer. So as you said, we can break this up.

So for facility, network and hardware layer, could you give us a few examples of some of what you've seen are the most useful or highly recommended controls?

Jono Spring: Sure, and all three of these layers are things that the provider is going to control. So these are things that the customer has to ensure in their SLA, So things like physical security of the facility, robust policies for surveillance. Make sure that there's no access by unauthorized personnel and that the personnel that are authorized, are vetted to minimize insider threats.

It's also important that the facility has a continuity of operations plan. FEMA does have some recommendations for this but there are other organizations that also have such things. But you do want to make sure that there's a documented plan for what happens in the case of an environmental disaster or something like that.

At the network layer, the provider is going to need to implement sufficient firewalls, intrusion detection and prevention systems at the network layer border, active IP and domain name black list policies to keep known bad things out of the cloud as a whole because that's one of the main things it can do to help keep the data inside safer. It has to monitor the network also for anomalies. In the case of an intrusion or something, be able to do retrospective analysis by maintaining some network flow data.

The hardware layer: it needs to be able to monitor hardware in order to do the elastic rapid provisioning. So those things should be monitored basically already. Also verify that the hardware is free from tampering occasionally and make sure that no one has gone in there and switch some wires around.

Julia Allen: So you mention that these are all provider responsibilities, typically, and is it the contract or the SLA where the customer can either call for or provide requirements for these controls? Would that be the best mechanism?

Jono Spring: That's where it would have to happen. I think that the current state of affairs is that the providers sort of run the market and so may or may not provide those things. SLAs need to be read carefully and if you have particular concerns, I think that that is the place to enforce them but you may have to band together with some of the other customers to get the provider to listen.

Julia Allen: And just to take that thought one step further, is there typically an ability for a customer to do an assessment or an audit or run some kind of automated tests or methods to see if some of these controls are in place? Or because, as you say, the service providers are really leading the charge, that's not practical?

Jono Spring: I don't think that, in general, the cloud providers supply that sort of information. However, large enough customers, I'm sure, can throw their weight around and get that information.

Julia Allen: Okay, okay good. So let's go on to moving ourselves up the stack: the next two layers, which are the virtualization infrastructure and operating system. You have those combined together. And then the middleware and platform architecture. Can you highlight a few controls for each of those?

Jono Spring: Sure. So for the virtualization infrastructure, this is providing basically the services on a traditional PC that your operating system would. So there might be several operating systems running on these several boxes but they're all controlled by some sort of central authority. This authority needs to be secure and controlled; non-essential functionality removed in those individual operating systems which are federated. The virtual machine control has been demonstrated to be vulnerable to hostile virtual machines and so there needs to be ways to revert compromised instances back to a known good state and to notice when that happens.

If that process is not documented by the cloud service provider, that should be a red flag that they may not have thought of that, which may turn into a problem down the road since that had been demonstrated to be possible.

For the middleware: middleware is a hard term to pin down. People almost don't like to use it anymore because it's starting to mean too many things. But basically this is virtualization management tools, data format conversion, enforcing access controls. Since this does enforce access controls and some security functions, this can be a significant weak point. Because, like in any system, the security functions are a target because they can provide a lot of value to the attacker.

So ensuring that these things are well coded, securely coded, fuzz tested (which CERT has a podcast on) -- those are all important things. Middleware is also a good resource since it mediates these things, it can ensure a secure communication by encryption, say, in an API. And as a cloud customer, should be careful of where the middleware is actually coming from. Just because the provider has middleware doesn't mean they're wrote it. It could have been outsourced. And so if you're concerned about the code quality and you trust the cloud provider, make sure that they are the one that's actually providing the code so that you can check all of your sources of the things that you're trusting.

Julia Allen: Okay, now in -- I know we weren't going to go with which of these controls apply by service model but I'm curious. If the customer is just purchasing infrastructure as a service (see if I've got this right), would they then be responsible for the controls in the two layers that you just described?

Jono Spring: In general, yes. They'd be responsible for the operating system and the middleware that they're running as well as the application. So the customer would then be in charge and responsible for deploying those things that I’ve recommended.

Part 3: Controls for Application and User Layers

Julia Allen: Okay, and there are two left, actually two left. So let's talk a little bit about controls for applications and then the ubiquitous users who are almost impossible to control but I know you have a couple of recommendations there as well.

Jono Spring: Yes. So applications, like middleware, need to be securely coded, fuzz tested. (We also have a CERT podcast on secure coding for anyone that wants to look back at that.) Applications are often web-based and so those must assume that the user is hostile because we've seen enough maliciousness out in the wild that that's a reasonable assumption to make. You want to use SSL, DNSSEC (which there’s also a CERT podcast on) for any web communications because those are best practices in general and the cloud is not exempt from those things.

Applications in the cloud are multi-tenant and memory persistent most of the time. This lets them serve the multiple users that you get the big economy of scale benefit for. This also means that if an application is compromised in memory, unlike on a regular host PC, it may persist for, essentially indefinitely.

So these also need to be monitored regularly and probably just refreshed regularly from a known, good, read-only copy. If you can white list application monitoring, as the provider, that would be ideal. If in infrastructure or platform as a service where the customer is providing the applications, that might not be as feasible. And also, these things should be documented so that the customer can see what is going on.

Julia Allen: Excuse me, Jono. What do you mean by white list application monitoring?

Jono Spring: Okay. So for example, an antivirus program is a blacklist application monitoring. It has a list of things that are known to be bad, which are black. And it checks running applications against that list to see if they’re bad.

Another approach to that, would be to use a white list, so things that are known to be good. And anything that is not on that list is not permitted to run. White lists are more secure because you can't trivially change the program to dodge the list but it is also more restrictive because it’s hard to change the applications that you are running in this environment.

Julia Allen: Great, thank you, thank you. Anything else about applications before moving on to users?

Jono Spring: Oh, there's so much more but I think that in the interests of time, we should move on.

Julia Allen: Okay, and so controls at the user layer.

Jono Spring: So user education will always be something that a security person can harp on. But what you can do as far as actually computer monitoring of users is you can monitor access patterns, if you're offering a web service where the user logs in from, and you can notify them of suspicious behavior. You can lock accounts out that are suspicious behavior.

This is similar to what credit cards are doing now. If your credit card is used from six different countries in a matter of a day and you didn't tell them you are on vacation, they'll shut your card down. You can do similar things for that. And you can make sure that your users know what uses are acceptable for non-cloud services and things, where they can use certain services if there's anything that's sensitive information that might be hosted.

Julia Allen: Okay, and then again going back to the service model, applications and users would be under control of the cloud service provider if you've contracted for software as a service, correct?

Jono Spring: Right. And of course user education is basically going to be under the purview of the customer all of the time.

Julia Allen: This was certainly the 30,000 foot fire hose treatment of controls by seven layers. But is there anything that you'd like to go back and highlight or anything that we missed at this level of review, that you'd like to mention?

Jono Spring: I just don't think you can't emphasize enough that the SLA is your keystone here for having a reasonable interaction with your service provider. Since everything is based on trust, that agreement has to establish that trust clearly and completely and if that's overlooked, then I think that you will have a lot of problems.

Julia Allen: So Jono, you've covered a wealth of material here. And I know that there's lots of other resources that listeners' can go to, to dig into this topic more thoroughly. Could you mention a few of your favorites?

Jono Spring: Certainly, so we've got the IEEE articles that you mentioned earlier in the show. The Cloud Security Alliance, CSA, has some guidelines. NIST, as I mentioned, has some guidelines not only on the definition but security and privacy in public clouds in general.

The European Network and Information Security Agency, ENISA, has a really good document on a cloud computing risk assessment, and Google actually documents in a security white paper, its basic cloud security tenets, which I think is also very useful.

Julia Allen: Well I cannot thank you enough for all your time, expertise, and preparation for our conversation today. I sure appreciate it and I think our listeners will enjoy this conversation. So thanks very much.

Jono Spring: Well thank you, Julia. I appreciate the opportunity as well.