Heather Terry (00:12):

Welcome to the Hurricane Labs Podcast. I'm Heather. And today we're going to be talking about vulnerability management policies. Here with me today, I have Roxy our Vulnerability Management Specialist and Bill Mathews, the Chief Technical Officer and owner of Hurricane Labs. Hey, thanks for joining me today.

Bill Mathews (00:31):

You're welcome.

Roxy (00:32):

Thank you for having us.

Heather Terry (00:34):

For sure. Well, first things first. What is a vulnerability management policy and why is it important that businesses have one?

Roxy (00:47):

Well, a vulnerability management policy will provide details and guidelines on how you're going to go about scanning, remediation, any sort of exception handling, or anything really that has to do with how your business is going to run their vulnerability management program. The reason you need a vulnerability management policy is first of all, it's the requirement for some certifications like PCI and SOC. And if you don't have these things written down, as it is with any policy, if you don't have them written down people aren't going to know what to expect and what to do. So it's, it's very important to have policies, not only for vulnerability management, but for other aspects as well.

Bill Mathews (01:39):

Yeah. We use them, we use the policies. Yeah. We definitely were motivated by getting our SOC II and PCI stuff to really get our policy management in order. But we use them more as a guiding philosophies for what we're doing. So, for example, like Roxy was saying, you know, it tells you what to do, but it also tells you when to do it, which I think is an important point because a lot of places will have a policy and then just never do the thing they're supposed to be doing. So, you know, our policy is to scan our systems at least once a month. We actually ended up doing it more real-time because we use an agent, but as long as it's being done once a month, it's not in violation of the policy, but that allows us to do a couple of things–it allows us to know what we're doing when we're doing it, and then gives us sort of a way to audit what we're doing. So we'll know, you know, 'Hey, system X hasn't been scanned in 30 days, that's policy violation', but also it could signify a problem with system X, right? So those are, I think are really important points when you're building this kind of policy is trying to decide, what are you going to do with this data? Don't build policies just to build them you to have some sort of philosophy around it.

Heather Terry (03:01):

So when you're creating a vulnerability management policy, what sort of goals or objectives should you try to keep in mind?

Roxy (03:10):

So the reason that I have to write a policy for Hurricane Labs and not just get it off of the internet and copy and paste it is because Hurricane Labs has different goals and objectives than every organization. We also have different capabilities because we are a smaller company we're not a large company with thousands of users. We have the ability to remediate very quickly so we can get rid of critical vulnerabilities way faster than a Fortune 500 company with, I don't know, tens of thousands of employees. So our objective would be to remediate quickly and that's something that we can actually do. So I think it's important before you even begin, think about what your goal is. Are there certain consequences that you would like to happen? Like, do you have a whole bunch of systems that aren't even being used? If that's the case like you could put in there then in order to prevent vulnerabilities from piling up on these systems, if somebody doesn't log in for X number of days, then we decommission the server. So that's something to consider. What does your organization want to accomplish with the vulnerability management policy?

Bill Mathews (04:40):

And I think that's a really important thing to consider because in a lot of companies, if it's not written down, it doesn't have a lot of sort of cachet, I guess, because people are like, well, I didn't know that. Well, yes, you did. It's in the policy. So for instance, you know, write down your desired time to remediation. You may not hit it. It may be too impractical for you to hit, but at least it's there and it's something for you to sort of strive for. So then when you're reviewing your policies, you can think, well, what can we speed up? What can we make more efficient? How can we get these things into, you know, more into compliance with our policy because the policy can say whatever, right? So if you say, I want my time to remediation to be a year, which I think is excessive for any organization, but I'm an idealist that way. You could say that and your auditor will probably be question you on it, but you can absolutely put that in your policy and sayyou know, I want my time to remediation to be no more than a year. Okay. It's no more than a year. So now, you know, when you scan those machines, when a vendor, for instance, has some Java dependencies you can then push them and say, look, we got to remediate this before the year's out. That should be long enough for anyone to update their software or do whatever needs to be done. Because again, the vulnerability management, you're not necessarily going to ever get rid of all the vulnerabilities, but you need to include a way to remediate that vulnerability.

Heather Terry (06:16):

What all goes into creating the vulnerability management policy, like who should be on the team, what other considerations should be addressed?

Bill Mathews (06:27):

We actually have a small little security policy committee. We call it. And those are the people sort of writing all the policies and they come from every group that we have. So we've got, you know, people from SOC, our admin team, don't think we have any of our Splunk people on it, but, but they don't have a lot of policies. Their policies are driven by customer change management, mostly. So we just have a blend of people. Then obviously we have Roxy and myself.

Roxy (06:55):

And one of the things that you might want to do, even if you don't have someone from every single group on your team, or especially if you don't is speak to asset owners and speak to people that are doing the remediation so that you can get an understanding of what their job is like, because you don't

want to create a policy that doesn't work well with them, or you don't want to create a policy that makes their job even more difficult unless their job to be more difficult.

Bill Mathews (07:26):

Unless you don't like them.

Roxy (07:29):

But I mean, the goal of writing policies in general is not really to get people in trouble or to make their lives harder or any of that. It's really to make things easier for everybody and create structure and guidelines to help everyone. You can't make everyone happy, but at least you can talk to the people that the policy affects and get an understanding because one of the worst things you can do is to just write a policy based on what your ideas are of what's acceptable and not taking other people into consideration. You're going to affect more people that way in a negative way.

Bill Mathews (08:09):

That is a super important point because, because your policy will just be a miserable failure if you don't get buy in and you don't consider what other people have to do

Heather Terry (08:21):

When it comes to say like you're creating your policy brand new, what sort of timeframe should you really be thinking about setting aside as far as, you know, dedicating to this? Like how long would it take to create one from scratch?

Bill Mathews (08:38):

Well, we actually, yeah, we just did that. So go ahead. Sorry.

Roxy (08:42):

How severe your ADD is? I suppose I can write a policy in less than a day, but I oftentimes what I do is I read about what the requirements are. I read NIST documents and I read other resources like auditing companies will have, you know, blog posts on what the vulnerability management policy or whichever policy I'm writing what it requires and what it should have. And honestly, I will read for two whole days and think about it and dread writing it because I think it's going to be wrong. And I surprise myself by writing it within like an hour. And so I do recommend that you spend some time researching and trying to figure out what the requirements are before you even write the policy.

Bill Mathews (09:35):

Yeah. And I think that, at least in my opinion, because I don't have to write them Roxy does, writing them is a lot easier than implementing them because implementing them requires you to get people, process and technology kind of aligned together. And that's always, you know, a lot of fun because you will get some resistance to the policy, no matter how careful you plan it out, no matter how many people you involve, you'll get some resistance to it. Where you're building this kind of policy you really have to consider what kind of cultural changes have to happen. So we have a guy here, I won't name his name. His name is Tom. And he hates updating. He hates patching. Is doesn't like any of it and always complains because, Hey, why did they move this button over here and do this other thing? So he would be an example of a cultural thing. We'd have to change right when we're talking about patching.

Because he doesn't like it. He's completely against it. And I love talking about him when he can't yell at me about talking about it.

Roxy (10:34):

So this is where Tom finds out on a podcast that he no longer has a job. Is that what you're saying?

Bill Mathews (10:42):

No, no. Tom knows better than that. Tom is very good at his job. Fortunately. Again, again though, that's important to point out, Tom's job is not patching anything because he hates patches. So I wouldn't put him in charge of patching. That would be crazy if I said, "Hey, you know, Tom, let's put you in charge of your vulnerability management." Tom would just flip out.

Roxy (11:03):

That would be setting him up for failure.

Bill Mathews (11:06):

Exactly. You'd want to avoid that. Yeah. And you want to avoid that with any of these policies. You don't want to set your people up to fail because then they're going to start hating any policy you come up with.

Heather Terry (11:18):

Once it's all written down and approved, how do you go about actually implementing it?

Bill Mathews (11:24):

Oh, that's a whole other podcast.

Heather Terry (11:26):

Haha. Okay.

Bill Mathews (11:27):

At a top level, you got to have executive buy-in and again, we're small, so it's easy. I just yell at people and it gets done. You can't do that in a bigger organization. So you got to have some project managers on board. You gotta have some people that are a little more involved than you know the operational side of things, because again, a project manager, some sort of executive they're likely not going to know that your main servers require an outdated version of Java, right? So they, they just may not know that. She's just got to make sure you get all the right stakeholders involved.

Roxy (12:06):

But when it comes to implementing the policy, again, it really depends on what your objectives and your goals are. So have those in mind and try to reach those. Don't worry so much about little details. Sometimes what I'll do is even though I have something more detailed in my mind, I'll write something a little vague in the policy. Don't worry so much about the little details, worry about what your goals and what your objectives are first. Because if you try to do everything all at the same time, it's going to be overwhelming and you might actually miss your goals.

Heather Terry (12:45):

I used to go to art classes with my mother and they would always say, "Draw the form first, draw the fingernails last." You won't want to start drawing with the fingernails and get so caught up in the details that you miss drawing the rest of the form.

Bill Mathews (13:01):

Oh is that what I've been doing wrong?

Roxy (13:08):

You know, it just depends. Maybe, maybe your goal is to draw really good fingernails. You want to put detail into that.

Bill Mathews (13:19):

But, but to your point, Heather, I think that, I think that's an apt metaphor for what we're for what we're doing. Cause you don't want to start your policy. And we used to actually make this mistake all the time. We were building policies, you know, you'd start building and then somebody would say, "Well what about this one little tiny thing that might happen 10 years from now?" You know, you don't, you don't want to start there. You want to start very general and then have a process to carve out exceptions and specifics because everybody's had that one system that is just a pain, but you need it to run,

Roxy (13:53):

Right. You definitely can't have unrealistic expectations. You cannot expect everything to be a hundred percent remediated all the time. If that were true, then you could just do your job for a month or two and quit.

Bill Mathews (14:09):

Excuse me for a second while I write this down in Roxy's reviewOkay, there we go. Got it.

Roxy (14:19):

So the fact that vulnerabilities keep showing up actually keeps you, keeps you having a job. So it's not going to be a hundred percent.

Bill Mathews (14:29):

No, no. This is always going to be a, a process and a thing, right?

Heather Terry (14:33):

So now you've got this policy created. How often should they be updated? Are there like particular events that would trigger a review or update of the existing policy?

Bill Mathews (14:43):

At least annually, according to most of the compliancy things. We try to you know, update as soon as we discover something wrong or something that we needed to change, we will go back and update it. Otherwise we're not going to remember frankly. So we try to update it in as real-time a manner as possible.

Roxy (15:05):

So something that I do whenever I'm reviewing a policy, or whenever I'm looking at a policy to see what I'm supposed to do or what to expect, I always look at the revision date because if it's too long ago, like if it's more than a year ago, I'm going to want to check and make sure that is this still valid? Is this still what we're doing? So if you ever looking at policies and you notice the date is over a year or over what your information security policy defines as when you should be reviewing policies. And it's a good idea to bring it up and just review it in that moment when you discover that. But you should be reviewing it at least annually, if you're getting certifications and such.

Heather Terry (15:47):

Alright, so you have this shiny new policy everything's laid out, everything's defined and something happens where the policy isn't followed. What happens next?

Bill Mathews (16:00):

What a trick question. So you can't, you can lay that out in the policy itself. The most critical thing that's going to happen is your auditor's going to go, "Hey, you were supposed to do this and you didn't. So, you know, here's an admonishment that you better do this next time, or you failed to do this two years in a row." And now there's a, you know, some consequence that each compliance, regulation, whatever lays out for you. With vulnerability management, it's a little different because that's a critical thing, right? So it's not like you forgot to tell your users to power off their computers every night. It's more like, you know, I miss this huge glaring exploit. So a lot of these high profile breaches that you're seeing that you see or are seeing those are all from imminently discoverable things. They're not zero days, they're things that have been out for months or even years in some cases, and that a scanner would return back to you. So if you're not taking those results and opening tickets with the asset owners and saying, "Hey, we need to patch this, whatever we need to do to get this remediated," then you're not only not following your policy, but you're leaving giant holes in your network for bad guys to go or good guys in some cases to go, you know, "Hey, this is an exploit and I can do things with it." You should fix that. So there's, there's the practical sort of considerations, which is, Hey, I could get breached. And then there's sort of the business implications where you could actually lose one of your credentials. You know. So for PCI, for example, they're pretty not lenient about certain things. You know, if you aren't following your own policies about things, they can, they can not give you that certain renewal or whatever.

Roxy (17:49):

Now, there could be reasons why, if, if you're noticing a lot of policy violations, there could be reasons that have to do with your policy. So you can review your policy and make sure that your, for example, your exception, approving process, maybe there's something wrong there and that's why people aren't submitting exceptions or they, they aren't, the exception approval process is not working the way it's supposed to because there are legitimate reasons why somebody would not update a system, but that has to be approved by management or whoever is responsible for approving risk in your organization.

Bill Mathews (18:34):

Right.

Roxy (18:37):

So I was going to say, like, for example, I worked at a company that their entire virtual infrastructure was managed and maintained by a vendor whose product would not run on the latest version of Java. And that's a very common vulnerability finding is you know.

Bill Mathews (18:58):

No, no Roxy. You said it right. The first time it's a very common vulnerability

Roxy (19:03):

It's a very common vulnerability is you're not running the latest version of Java. So we had to submit exceptions every three months because the business could not run without this vendor. This vendor was the entire virtual infrastructure. So there are legitimate reasons. You just have to have the right, the right exception process and people have to actually be using it. They may not know that they need to use the process. So training could be a consequence as well.

Bill Mathews (19:36):

Roxy is a million percent right. That exception policy is super important and then on the other end on the actual operations end you actually have to do the exceptions either in your scanner, up to your auditor, whoever is actually doing it. And don't use software that can't run on the latest versions of Java. You know what I cracked that don't use software that needs Java.

Roxy (20:01):

That that would have been great. If we could have gone back in time a few years, when that company was first starting up and said, "Hey, how about you don't create your entire infrastructure with a vendor that requires Java?" But we couldn't do that. So it was like a year and a half or so of just exceptions constantly being submitted and approved.

Bill Mathews (20:28):

Fun times.

Heather Terry (20:30):

Alright. Well, thank you very much for joining me and partaking.

Bill Mathews (20:36):

As Roxy said, thank you for having us.

Roxy (20:38):

Yes. Thank you. This was fun.

Heather Terry (20:41):

Have a good one.

Roxy (20:41):

Bye!

Heather Terry (20:44):

And that's all for today. In our links below, we put together a few resources for planning out your policy, and that includes a checklist of six key tips for writing an effective vulnerability management policy, courtesy of Roxy. So be sure to check them out. Stay safe and we'll catch you next time. Bye.