#118 - Data Engineering (with Gal Shpantzer)

[00:00:00]

[00:00:09] **G Mark Hardy:** Hello, and welcome to another episode of CISO Tradecraft, the podcast that provides you with the information, knowledge, and wisdom to be a more effective cybersecurity leader. My name is G Mark Hardy, and today I've got a special guest who's been a friend of mine for well over a decade, and wants to talk to us about what you might think is very useful. And it's logging, and I didn't really think about logging a whole lot, but Gal Shpantzer is our guest today, and he's going to tell you, why you as a CISO or as a security leader, need to be thinking about what happens to all this information that you collect. It may be megabytes or gigabytes or terabytes or petabytes.

It's just this fantastic amount of information. But what do we do with it? Where does it go? How do we make it useful in things such as that? So as always, if you listen to our podcast, please make sure you subscribe on LinkedIn. Follow us there because we've [00:01:00] got a lot more information we put out other than podcasts.

But for now, welcome Gal to the show.

[00:01:06] **Gal Shpantzer:** Thanks for having me G Mark.

[00:01:08] **G Mark Hardy:** So, Gal, as I said, I've known you for a long time. You've been working for over 20 years in security, a virtual CISO. We've worked together on a bunch of stuff and things such as that. And I think one of the things you like to claim is that you've never really had a real job, but yet you're one of the top influencers in the twitterverse of cybersecurity. People can follow you @shpantzersec. Tell me a little bit more about yourself.

[00:01:29] **Gal Shpantzer:** All right. So I was doing some work in physical security. And a friend of mine that I met hilariously at a modeling shoot, not kidding, told me that he could read my 15 minute line by line. Where am I? And who am I protecting and what are we doing? So I inquired within and he told me to check out this thing called 192.168.1.1, which meant nothing to me at the time.

And I said, I don't know [00:02:00] what this is. It looks like it needs a password. And he said, try admin, admin. And I said, and I quote, that will never work. Because I thought what negligent person from the telecom company would give us something that had that guessable username password on something so important.

And of course, it did work. And that was the beginning of the end of that career and the end of the beginning of my information security career. And I started doing a lot of training and got involved with SANS and met all kinds of cool people. Who helped me and mentored me. And that was in the late nineties and now I'm here doing all kinds of cool stuff for over two decades now.

I've done work with Series A startups all the way to what I affectionately call megaglobalcorp in the kind of global 100. They measure their revenue in billions a day, as well as a lot of nonprofits and kind of highly targeted NGOs.

[00:02:54] **G Mark Hardy:** So a lot, a lot of background, a lot of exposure. You'd said giant, Fortune N where n is a small [00:03:00] integer all the way down to startups and things like that. And so, you've seen a lot across all this and across a lot of the platforms, but over the last couple years, I know that you got this huge project that was involved in working with collecting data and more importantly than that, how do you architect all that and things such as that? How do you take, not a terabyte a day, but maybe a dozen terabytes or creating a petabyte lake. I mean, this whole thing is to me, outside of my scope of experience, but this is something that you've already figured out and a lot of organizations may have to take a look at it.

And if you haven't done it now, may have to do so in the future. So what kind of got you into that area of security relative to everything else you'd done, which have been security assessments and looking at the business and other technologies as well?

[00:03:47] **Gal Shpantzer:** Sure. So in 2013, I started looking into ransomware, and this is documented on Twitter with some of my word play shenanigans, where I mentioned an old [00:04:00] police song. And did a little spoof, a play on words. I said, I will turn your files to alabaster when you find your powner is your master. And I think I hashtag it as encrypted extortion wear or something.

because we weren't calling it ransomware quite at that time, or at least I hadn't heard of it that way. But I figured out that this was something that we should be taking a look at. And by 2015, 16, and obviously 17 when not pitch and rolled, , we saw all these big ransom worms or destructive worms.

And before that obviously there was the Saudi Aramco kind of nukeware, pavedware so I thought we need to understand what it looks like inside the LAN when something bad is happening. And that would require a certain amount of telemetry that looked like a lot of turn-in file systems, a lot of lateral movement at some scale. And so that's the kind of thing that I thought would be interesting. I started looking into it, what, what are people doing around that? I started looking into some of the web scale technologies that came out of the old older tech companies. [00:05:00] LinkedIn. Open sourced Kafka. I was very new at the time.

Netflix was certainly doing cool stuff. Google and Yahoo were doing amazing things, literally downloading the entire web every night and that, that's where Hadoop came from. So I thought, this is really interesting, but more so than the data science and analytics aspect of it, which I thought was almost a solved problem per se.

It dawn me that I think a lot of us were really kind of stuck in the before LAN, where we need to analyze stuff and I theory, we know how to analyze it, but we really need to know how to get a discipline around gathering, turning on logs, intercepting them, moving them around, storing them, and then creating use cases for very fast analytics as well as very long term analytics of things that were already in some sort of frozen storage for a while, and I'll give you a silly example that I think everybody here can relate to. Have you ever been on your phone and you're interacting with some application or a browser, and it [00:06:00] looks like your finger and the glass on your phone are not talking to one another, and you end up kind of jabbing at the glass on the phone.

Kind of poking it and kind of poke, poke, poke, poke. That is a frustration signal, or in some people call, a subset of frustration signals. A rage click, whether it's a mouse on an old school computer or it's a tap, tap, tap, poke, pop, poke on the glass of a phone. And that frustration signal is something that people who do this for a living and instrument applications and move things around a little bit.

They can tell when things are not going right and they can tell things almost immediately in near real time because they've instrumented these applications with logging APIs and observability APIs that then show them in their war room, Hey, this is what's happening with the user base, which ranks in, hundreds of millions, in some cases, literally billions of users.

So this is where these web scale technologies are used mostly in marketing instant fraud detection and stuff like that. So I thought, this is really interesting

because why can [00:07:00] someone understand when I'm rage clicking on whatever silly app I have on my phone? But people who are protecting critical infrastructure are having to deal with a lot more latency than that.

I thought that's not right. And so I looked into it and one of the things that I was looking at again for ransomware, and then in 20 16, I did some work for a very large insurance company. They needed to be able to understand the providence or where did this come from and how do we tag it and run it through the stream of data so we can pick it up later?

When GDPR started coming around looking from privacy tech engineering, they needed to understand what do we have on you? Where is it? How do we claw it back and can we delete it based on a user request because they have the right to essentially be forgotten. So that interesting use case that came from Europe started coming over the Atlantic and I thought, okay, this isn't an actual use case for this as well. And so I helped a large insurance company build a architecture for a data pipeline and I did [00:08:00] not come up with any of this. This is all, again, these big webscale technology companies like LinkedIn, like Google, like Yahoo and certainly the other big ones in the world out there that are doing a lot of social media, etc..

So, Netflix being one of them. If you go to Netflix engineering blog and you punch in Netflix, Keystone Pipeline, not the gas Keystone Pipeline, but the logging Keystone Pipeline. They open sourced that architecture and talked about it on YouTube and on their blog very clearly in 2015. And I thought, this is really where we need as security people to understand how, what is even possible.

and that's where you get into amazing things like enabling the ingestion of a log source from a lot of different log sources or a log source one time, and then distribute it in many places while potentially if necessary, analyzing the log flow in real time inside of an intelligent pipeline. With products like Kafka and more recently Flink, and these are [00:09:00] all open source tools you can get for free.

And in some cases there might make sense to get a, some consulting or support from the kind of vendors who support open source tooling like this. But this is entirely built on open source and it's just amazing what's out there. So this is a niche called data engineering, which in my mind is a critical first set of steps to create data operations that allow you to grab the data, put it where it needs to be, and then do the analysis thereof. And so that is where I think the mindset shift is, or the skillset shift is you need to understand data engineering as a part of

your SIEM and your SOC operations. And interestingly, for many CIOs and CTOs, once they see this project coming out of the CISO shop in many cases, they all be very interested because they have their own massive data sources that they need to put in several different [00:10:00] places instead of just one place.

So an example of this is kind of anti pattern is where you are contracted with either a managed SOC or a managed SIEM or even a SIEM that you own, and you allow them to organically and mesh you into their data pipeline. And what I mean by that is their proprietary tooling will be found in agents and various other aggregators on your network and on your endpoints and servers.

And then you're stuck. And the answer to the question, what should we be logging in order to answer which questions on the data science and data analytics side? Meaning which queries for detection, engineering should we be running, which imply that we have those logs in the first place. Those are good questions. The actual question that you're being forced to ask then is, can we afford to run those logs that we want [00:11:00] to answer those questions? We want through this water meter that the SIEM people set up, because it's in a gigs per day licensing model. Now you're stuck because you allowed them to put their proprietary agents and their proprietary aggregators and they built your pipeline for free but the cost is lack of modularity, lack of flexibility, and again, you are stuck answering the question, what do I want to log instead? You're answer, you're asking the question, what can I afford to log through this other pipeline that was built for me without really un answering my needs? Which brings us to the data engineering Iron Triangle.

I didn't create this either. I think the person who created the Google Drive created this. I heard this on one of his webinars a couple years ago. When one of the big N G A V companies bought their outfit because they're all getting into this game as well. So what is a data engineering iron triangle?

Well, in real estate we have location, location, location. In project management, we have good, fast, cheap pick two if you're lucky. [00:12:00] And in iron triangle of data engineering, we have cost, latency, and, scale. Right? So if you think of the iron triangle of data engineering cost is how much money can I afford to spend on whatever SIEM you have or whatever storage fabric you have.

Ideally, you have both. But when you're enmeshed in a kind of proprietary built by someone else for their financial purposes, instead of your engineering purposes, you end up having to route all of your data through their pipeline and

into their water meter, and then they'll tell you, well, you can just export the stuff you want to long-term storage, which is a silly thing to do.

Why? Because you're not using RAM and CPU power that should be spent on indexing and answering questions and queries, but now you have to grab all of the logs that came in that day or that. Grab them, zip 'em up, and then tail them out over the network. So that's a silly thing to do. You should be doing that before it hits the water meter shaping [00:13:00] that bandwidth and exporting it in parallel to potentially say a data lake like Amazon s3.

Right. So if you look at cost latency and scale, the higher the scale, obviously the larger the cost because of that water meter. And in many cases, the longer your queries will take, and in some cases they'll even time out because even if you have good query discipline, just a sheer amount of volume, which in many cases because you were not properly shaping the bandwidth you're spamming yourself and you're DDoSing yourself in terms of financially and in terms of operational use.

Now, some of these pipelines do have the ability to trim and shape bandwidth before the index or water meter hits. But what that means is, Logs are lost forever because you trim them at the source or inside the pipeline before distributing it to SIEM, but also not SIEM. Right? So you, you want to be able to answer a fundamental question When we have our logging pipeline, can we take these [00:14:00] logs and at will send all or none Or some to SIEM, but also not SIEM. Whereas when the SIEM builds your pipeline, again, I'm being repetitive here because it's a really critical aspect of this. Once you get enmeshed in their pipeline, you are theirs financially and operationally, and that's what we want to avoid. So cost, latency, and scale, no one tool can do all of those things really well without spending bazillions of dollars.

If I want to go and win a Formula One race, I'm not getting a Jeep or the other way around if I want to go off road in the. I'm not getting a Ferrari. So you really under have to understand the use case for this data. And the same data in some cases can be and should be stored, processed, analyzed, and queried in different ways by different tools for different reasons.

And that's just not possible when you're stuck in someone else's pipeline.[00:15:00]

[00:15:00] **G Mark Hardy:** That's rather a bit frightening to a certain extent. because I think to a large measure, we often outsource things. We let a vendor, a vendor who says, I'll take care of this for you. Okay, great. I'll make it happen

for you. Great. And it's a lot easier because as CISOs or security leaders, We've got all these things on our desk and we're just constantly go going around.

It's one less thing we need to worry about. But like anything else, as you said, if you entrust your data stream to somebody else, they're going to build it and they're probably going to optimize it for themselves, not for you. And anybody who's ever tried to change cloud providers understands exactly that.

It's a bit like Hotel California. You can check in any time you like, but it's really tough to leave.

[00:15:37] **Gal Shpantzer:** It is.

[00:15:38] **G Mark Hardy:** But when you look then at some of the usage, For this and is, is a case for data engineering, which I think most of us, and not even heard that term before. A lot of organizations, if you take a look at how they collect their information, all these logs, all these different sources trying to go ahead and run these scripts or have different servers, it's I think what you had mentioned is a spaghetti ball of pain in one of your talks [00:16:00] and this information just coming in from everywhere. We're trying to figure out how to do that as compared to actually making it into a reasonable pipeline. Whereas you had said whatever the producer is of these events, it's going to go into some frontline using Kafka, for example. I'm going to ask you to explain Kafka and Flank in a moment, just so if people aren't familiar with these tools, we're not going to give a big pitch on it, but at least 30 second overview. Now, that control plane is your own, it's your self-service, and you can decide what to do with it. Are you going to send it off to your SIEM or are you going to go out into an elastic search? Can you put it out there into AWS Glacier saying, Hey, I might never need this again.

But if I have a query that comes up where somebody says, Hey, we think we've had a low and slow entity in our environment for the last few months. Remember like SolarWinds.

How do you go ahead and do that? And this is a way that we can actually for the first time, essentially answer this question without handing over huge bags of money to the third party who said, well, thank you for giving us all this data.

We're [00:17:00] going to give you an answer, but it's going to cost you because you said they've been running the water meter. So this unpacked some of those questions there for our audience. So, quickly tell me a little bit about Kafka and Flank. What, what are those tools? What do they do for me?

[00:17:13] **Gal Shpantzer:** Sure. So if you think this may not be kind of the most technically correct answer, but it's very hard and it took me a while to understand that analytics can happen in the pipeline itself. And, and Kafka and Slink are not necessarily a pipeline tool, but they're a streaming or realtime analytics tool.

And so, kind of cognitively, if you think of the old school way of doing analytics, and I've talked to people. CIOs, CTOs or CISOs that have been doing analytics for many decades, two, three decades, very senior leaders, and it blows their mind to try to wrap around the idea of analytics inside a pipeline or real-time analytics because we are typically thinking of analytics as we have a place where the logs come from. We open a window, we let them in, we close a window, [00:18:00] we put 'em in a data warehouse or database, and we turn the crank on a SQL query for. Right in, in a database and then we get an answer. Those are what's called bounded data sets. And in Kafka, in Flank and NiFi, for example, all open source tools that do a simple event processing or up to including a fairly complex event processing and kind of sql.

Queries in line. These tools are doing stuff on what's called unbounded data sets. They're literally coming in in real time and they're getting analyzed in real time. And you can do almost anything you can do with a regular SQL query on a realtime stream. And so streaming analytics is something that is really critical to understand that is available to us and we need to expand our understanding of what's possible using these. So Kafka came out of LinkedIn and was open sourced, I believe in 2011. So it's been around for a while. There are companies that operate only Kafka. There are companies that operate Kafka plus some other things [00:19:00] for you as open source kind of support companies. And they do this for, big governments, big companies, and tech companies as well. Netflix for example open sourced their architecture of the Keystone Pipeline. And if you look at Kind of a technical infrastructure there, you'll see a fronting Kafka and then a more of a distributing Kafka afterwards. But there's Kafka and Flink can be kind of used together as a wonder twin powers activate where Kafka can be what's called a pub sub or published subscribe aspect of a pipeline where you have the endpoints or the consumers Topics or subscribing to topics from Kafka. So for example, some people want to put into the SIEM itself only alerts from, let, let's say their EDR or EPP and JAV tools, right?

And that's fine. So they can subscribe to the EDR alerts channel, if you will, from Kafka. Whereas s3 you can subscribe to the entire raw data stream of every binary and process that emerges from that binary. So you can see the [00:20:00] entire kind of EDR tool chain happening, and is filling up buckets in s3.

And like you said, in many cases it's not going to be something you can stick into a SIEM full-time. So those kinds of tools are allowing us to collapse the kind of combinatorics or, or like you said, I call the spaghetti ball of pain. We have x producers, if you will, on a, on a column on the left and y consumers on the right and in the middle there's kind of middleware of a bunch of scripts and a bunch of APIs and various pools and SFTPs, etc., that are sending logs and also receiving logs.

So it just looks like a spaghetti ball of pain. Do an effort to map out even what are we producing from where is it on the LAN? Is it in the cloud? Where is it going? How long should it be there? Which brings us to something called data contracts, which we'll talk about in a bit. But Kafka is a essentially a message bus where allows you to take data from a bunch of places and then publish it as a topic [00:21:00] that consumers can subscribe to.

So it's not all or nothing, and it gives you granularity where you essentially are enforcing. Kind of logging and ingestion policies via this middleware, and it's extremely powerful. What's interesting about doing some of these projects at Massive Scale at these American Global Corpse is you see even Kafka and any other tool in the IT space has some weird scale issues, right?

So Kafka is a very high throughput tool, but it's not necessarily designed for extreme high concurrency. So we started getting into 50,000, a hundred thousand, well over a hundred thousand kind of six. Independent producers in terms of agents and APIs and various other pieces of data producers that are sending logs into Kafka.

And you don't want to necessarily have that as your front end directly into Kafka, concurrently with six figures of sources. So what we did was we fronted it with NiFi. So we had a NiFi sandwich, had NiFi on the left, doing that first mile [00:22:00] from all of the various. Then that sent it into Kafka with a massive throughput, but much less concurrency.

There's only so many NiFi servers. And on the other side, we had either direct subscriptions from Kafka from the consumer side, or we had NiFi sending everything out to was Amazon s3. And after a certain amount of say serverless Lambdas does on S3 operating, we then would send into the SIEM.

Right? So we had the ability to send all or some or none of a given topic into the SIEM and allowed us to strip away things that we didn't want into the SIEM, but we did still want for the long-term threat hunting or for compliance that just didn't have a use case for a particular detection query in the SIEM itself, if ever.

Right. And so the question really around these types of tools, Are they help you enable the data analytics and the retention periods both instead of just one or the other, like we talked about earlier, where you're, you really shouldn't be blowing rare resources, whether on prem or in a SaaS cloud [00:23:00] service because those are not unlimited either.

They have to pay Amazon as well. You should not be blowing your hardware resources on those export once a day to be able to put 'em in long-term storage. It should go first into a data lake in my mind, at any scale. And then either in parallel for the alerts certainly couldn't go directly into the SIEM because you don't want that latency or complexity.

But the raw logs should definitely go through the pipeline and some storage fabric before it goes full on into the SIEM. So we talked about why we need to do this because we are tired of paying too much money for that water meter for not enough benefit. We want independence, modularity, scale, and flexibility of where do we want these logs to go.

Possibly to the SIEM forever, but also definitely somewhere else. And we need that flexibility. And then we talked about latency, cost and scale. You have to have, by definition some sort of compromise and you are able to do literally real time streaming analytics upstream of the [00:24:00] data warehouse or the SIEM itself.

I'll give you an example of upstream streaming analytics that is really stupid simple. So simple even a Shpantzer can do it. And that is my favorite use case for data science enabled by streaming analytics in this case. And that is sort by descending. I'm not smart enough to do. Super cool data science analytics techniques, random walks in the forest and all that cool stuff that I've seen from various places.

Good for them. They're really smart data analytics people. I'm not, so I thought, okay, how about we just sort by the top talkers on the network and we instrumented. Infoblox and this mega global corp had Infoblox and they had eight massive Infoblox servers that were just streaming a bunch of syslog.

We're talking about tens of thousands of events, a second of just Infoblox. DNS on the local LAN, right? It's a big company, but the LAN itself were producing tens of thousands of events per second, or eps into the pipeline after we had transitioned it from the kind of old school Syslog aggregators.[00:25:00]

And because NiFi, can impersonate a syslog aggregator, whereas I like call them aggravator, you can just send them to NiFi I instead and collapse that and then also get APIs and also get SFTP and so on and so on. So that fronting NiFi where we had the NiFi Kafka NiFi sandwich was ingesting tens of thousands of events a second in DNS and I thought, just show me the top talkers on the LAN. And we did a sort by as descending. We opened a window every 120 seconds or two minutes, we would close the window really quickly and just sort by descending. And ideally you want to be able to show me top servers talking and show me top desktops talking because they look a little bit different and they just do a little bit different things.

So I saw a desktop very fast many, many thousands of DNS queries coming out during that time. And I thought, okay, why would this desktop be doing this? And usually when you have a rate as a signal that is really, really high, there's either an IT fault. Some agent or some [00:26:00] process, instead of doing an exponential back off, it's doing exponential back on whatever that is.

And it's actually screaming louder and louder. And I thought, why would it be doing this? But in some cases it's potentially a red team or some bad people that have initial access and they're doing some sort of reconnaissance or lateral movement. And so top talkers in near real time will find that, and again, you don't have to be a massive, big Stanford PhD data scientist person to really do amazing work around detection engineering.

You just have to ask people, what do you want to see really, really fast before the SIEM even gets to the batch process? because they're inherently batch process and they're all competing. So they're in some cases stacked and they wait 60 minutes or 90 minutes or once a day. To run a particular important query because there's only so much, again, power under the SIEM itself.

So upstream sometimes we want to help do some pre-processing [00:27:00] for the SIEM itself and in some cases pop those out immediately. Similar to how the marketing people or the AB testing people are doing work around. What's hot, what's new? What's a frustration signal in terms of rage click? Is this fraud or not?

The moment they LAN on a registration page, for example, etc.. And we're getting better at this because you see some of, some of the folks doing this with various kind of graphs and somewhat, fairly fast analytics around, Hey, this person is not who they say they are. And they're trying to bang on a particular identity interface, for example. So we're getting better at doing this stuff, but, to large extent, streaming analytics is an exotic skillset or even a concept for most

IT people and security people. And I'm here to say it's been done, it is being done, and it should be explored as an option for some of your use cases as well using these tools and these architectures that are proven at webscale. So, These, these types of things were used to bust red teams in near real time where they came in and did lateral movement and [00:28:00] reconnaissance where you have one to end where you have a rate and a directionality. You can look at producer, consumer ratios, flipping, where things that are typically serving stuff all of a sudden are receiving stuff that could be where they're being abused to become a base for staging data before exfiltration.

All kinds of cool stuff. And so, I'm not inventing here or telling you anything new around you should do these cool detections, but rather, I'm telling you there are ways to implement. Upstream of these detections, the same detections very quickly and or make sure you are able to physically produce the logs that you have and send them to the places where they need to be in addition to potentially just the one SIEM that is in many cases taken over your pipeline.

And again, flexibility, modularity and financial independence, because we want the lodge to go possibly to the SIEM, but definitely to also not the SIEM.

[00:28:56] **G Mark Hardy:** So it's an interesting model that we look at here. And so now, [00:29:00] for example, the analogy that's coming up with in my mind is that if you're the National Transportation Safety Bureau, or NTSB, and you want to know what causes a plane to have an incident and things like that. Well, you could recover the black box, which is like going through your SIEM and going through the logs.

Or you could do real time monitoring, which would be your collision avoidance system that you have up in the air to make sure that, Hey I have information. You have information. Let's talk. Oops. Looks like we're on a collision course. Let's not keep going the same direction. And so, in a way you're doing that real time analytics, which is actually allowing you to make better decisions faster.

In this case, it's life safety, but in our enterprises it's more enterprise safety. But there's a couple other things. So in addition to doing the instrument analytics by getting rid of the spaghetti ball of pain, by having some sort of a realistic pipeline that can handle a fantastic amount of information we're trying to get out of the situation where a third party, mostly SIEM and we're not, we're not picking on SIEMs at all. They, they provide a very valuable service, but you don't want to outsource that whole pipeline to 'em and all your data. But the other thing also is that in addition to [00:30:00] doing that pre analytics, are

there ways that you can go ahead and trim those logs as something that's reasonable?

For example, I remember early on was doing blockchain research. I think like, well, wait a minute, this could be more efficient. Instead of making everything an ASCII, why don't we go ahead and compress it? We could go ahead and use a different character set ta, tata, and there's, and to a way, then there's a difference between what is the value that you're recording and then the whole string of information that has that value.

So, so any thoughts about that and how that is going to reduce my ingestion load, which is going to save me time and money and make my stuff work a little bit better.

[00:30:33] **Gal Shpantzer:** Yes, absolutely. So when you're looking at the idea of a financial water meter in front of a SIEM and really part of data engineering and, and enabling data analytics is explicitly thinking and documenting and surfacing and explaining and communicating over and over the explicit water meters that we know are.

But nobody is thinking of, everything has a water meter, [00:31:00] whether it's a rate per day that could be, in some cases, events per day. In some cases it's, it's gigs per day. It could be instead of, so some of the SIEM companies have felt the absolute rage from the community about, we're sick and tired of this terrible model that forces us to ask bad questions instead of what should we be analyzing?

What can we afford to run through the water meter? They're saying, okay, well, we'll give you quote unquote unlimited ingest. And so in some cases they're saying, well, we're not even talking about that stuff anymore. What ends up happening is they're limiting you on workload pricing, so they'll force you to do good query discipline, which in theory is easy, but in reality is not a trivial thing.

And also I'm hearing from some of my larger, kind of multi petabyte day net new clients is that sometimes these companies promise you, In writing or verbally or ideally both. That they'll give you unlimited ingest, but then they tell you, well, security logs are not really unlimited. Okay, well what is a [00:32:00] security log?

Is DNS a security log? In theory, it's just an it source. We love using DNS you and I have worked on some DNS log source incidents where we were able to

scope that stuff. But DNS in theory is not a security source, right? So we have to really just understand the financial model. the, the coin operated machinery in the sales and procurement process.

Document that and go all the way back, if you will, from the right where the SIEM is, where the storage is, where some of the analytics tools are, and I say tools in, in multiple, all the way to the left, where the sources are. The sources could be. Fluentd and a Kubernetes cluster on prem. It could be a Fluentd and a Kubernetes cluster in the cloud.

It could be an API from Salesforce. It could be an API from 365. It could be an old school syslog Infoblox server on prem. Could be all those things. So those are the sources and you have to look at the rate, and a rate is gigs per day and events per day. And then you look. Hey, what does weekends look like?

What does a [00:33:00] Wednesday morning look like? What does a Wednesday night look like? So you want to understand how to build out these types of products and services for use by the people who are doing the analytics, whether the CIO, CTO, and or CISO. And the pipeline is really agnostic. It doesn't care who or what is using these logs in terms of analytics.

It doesn't have to be a security source by any means. It could be dns, you use for performance, for all kinds of monitoring and analytics as well as for security, obviously, short and long term. So how do we trim the data? Well, first we capture it, but before that, we actually have to produce it.

So you go in, you say, what do I want to log? What can be logged out of this thing? And in many cases, people are stuck with the inability to even instrument at the source. To produce the correct logs or any logs at all because they don't necessarily own the code. If you have an application and you do own the code, you can instrument the code by using some drop-in open source tools or in some cases [00:34:00] closed source tools.

But an example of an open source tools would be a Fluentd or fluent bit, and you can instrument your application what it's dot net or some other Java or Ruby. and then you boom, you create a application, essentially logger API that then sends stuff over the wan or LAN through a web hook, for example, and, and into the intercept.

But in many cases similar to how streaming analytics is kind of a weird, exotic thing, a concept in many cases, we are still really generally good at instrumenting operating systems and. but we're generally really bad at

instrumenting things like identity services and other applications that are running custom code.

And why is that? Not sure why, but it's almost universally the case and it's not that hard to do if you follow the folks who are really doing the logging at the application layer in Fluentd fluent bit we're open sourced by a company called Treasure Data and they became dominant in the kind of Docker and Kubernetes container.

They're in all the big clouds. [00:35:00] Now, an example if you use Google Stack driver from gcp, they'll tell you to go download slow deed, stick it in your application. If you own the code, recompile it, and boom. With that secret that they gave you, the Google listener on the internet, on the perimeter, VCP is now accepting logs and now you have APM kind of product from Google Stack driver functioning.

So that's an example of, of open source enabling the instrumentation of all the way on the left of the logging itself, that not even first mile, but first meter, and then say, okay, well I'm going to start sending stuff out to the various aggregation tools.

[00:35:35] **G Mark Hardy:** So we've got a bunch of ideas here, and so I figured in the time we have remaining, let's think about what would a CISO or a leader do now that all of a sudden we've opened up our mind to say like, wow, I've thought all this stuff was just taking place automatically or as a fire and forget.

I've set it up. I have some AWS logs that I've been accumulating for years, and I go poking around there. I say, well, what am I actually recording? And what am I saying? And if I ever had to pull that back and is it in a format that I can [00:36:00] actually do queries that are useful? And this is outside of the SIEM, but if.

what would you recommend in his next steps if someone were to say, okay, go, I'll drink the Kool-Aid here, and I'll say, yeah, this, what we have is we got a spaghetti ball of pain. We've allowed third party vendors to basically own our data pipeline. There's a big water meter that's running out there, and we haven't optimized what's going in there.

So the meter runs faster and faster and faster. And of course, it runs both ways when you're trying to get the information out. But what are some good step-by-step ways to say, let me get control of this thing and move forward, and

therefore start to get to a point where it's a sustainable model where you can do real-time analytics in the stream.

You can answer questions and find bad guys in your environment very quickly by looking at anomalies like, Who's the biggest squawker for DNS as you had indicated before? And, and basically get a handle on this massive amount of information to say, Hey, we can call through it and do a lot more useful things.

Do it. And can we do that within the next five minutes of what we have left on the show?[00:37:00]

[00:37:00] **Gal Shpantzer:** Sure, sure. So, one of the things we want to do is we want to organize to operate a modern pipeline. So at layer eight, we want to understand who are the people who own these sources. Typically, it's not the CISO, even in some cases when the CISO has some sort of consultative ability to say, here are the firewall logging standards.

The firewalls are actually owned and operated in many ways by the network team. It reports up to the CIO, for example. So you want to get together with your CIO, SVP of engineering, who runs applications. Sometimes it's the SRE team who keep the lights on on the website and things like that. So you want to organize resources, also known as producers to, Hey, we're servers, databases, applications, network, NetSec, EDR, AV desktop, etc.

Then you want to say, okay, well what are my main consumers or syncs? So we have producers, consumers, sources, and syncs. The main SIEM, obviously, is one of them. What's going in there now? What do we want to put in there now, but we can't afford because of the enmeshment of the data pipeline by proprietary agents, etc..

What are other analytics [00:38:00] we want and how fast do we want them? What does it even mean to have DR / BCP and long term ability to move logs around and do threat hunting on them 3, 6, 9, 12 months later, and then understand that data engineering is an enabling momentum producing skill set and tool set. People who get the plumbing, the logging, different file formats, ETL or ELT, then you have the vendors rediscover with them.

Let's talk about water meters. Let's talk about pricing models. Let's talk about what, what's going on here? because we're growing. We have zero trust. We're going to get more digital transformation, more sources and more places giving us fatter, bigger, more expensive chattier logs from all kinds of continuous reassessment of who you are, what machine you're coming in from, where you

are, where you talk to your MSPs, talk to your cloud vendors and start explaining to yourself the spaghetti ball of pain. And then when you look at the use cases, you have to understand your own needs. Latency [00:39:00] sensitivity is a little bit of an advanced thing, but in many cases, really it's about size. What are my EPS versus gigs per day?

What are my rates and volumes? How big is my current capacity? And what is a projection? Even if I don't add any more tools or sources, where will I be in one or two years? because they're all growing. and then what do I have now? What do I know and who do I know inside this company that does analytics? because they've probably been, have had to do wrangling of the data itself.

So in many cases, your big data, analytical people, even though they don't want to, are being forced to do the work of data engineering. So they have some understanding of this stuff. Clearly cloud versus on-prem. And then migration plans from on-prem to cloud in some cases back re repatriation.

Those are things that are typically between CISO, CIO, and CTO steering committees around big movements of platforms and data sources and data operations and big business implementations that take months or years to understand the big currents and get involved. Is there Net new sources are a great [00:40:00] place to do a greenfield analysis where, hey, what does this look like?

What is possible in the order of operations? Identify the sources, log at the source, what does that first meter look like to instrument? Can we capture them in terms of aggregation on the LAN, or on Cloud, ingest them into something like NiFi, Kafka, Kinesis, Event Hub, stack Driver, etc.. In some cases do that transformation.

We're filtering, duping transforming from say, a CSV or a JSON into say a parquet, which is a part of hacking a water meter when you're doing a lot of massive amounts of storage for a long time, because that matrix of how much you ingest every day, let's say one petabyte. Times three years. Well, that's a lot of petabytes that you need to store.

So even understanding what it looks like to hack the water meter at the storage, just understanding that, just being able to say, put an s3, flip it with say Amazon Glue, and then it moves to Parquet and then compress it and then put it in tiering. So you have really three or four knobs and switches in the data [00:41:00] cockpit that you're able to implement as data engineering hacks that enable.

Resourceful and effective analytics. And so you can look at things like agent based. You can look at things like agentless, for example, like syslog or TCP out or web hook, things like that. And WEF/WEC in the Windows universe. It doesn't all have to be agent based. There are too many agents out there anyway.

And in many cases things like a miniFi or Fluentd fluid bit can do multiple things. Grabbing stuff from the endpoint. That you can actually collapse some of those first meter tools anyway. And so understand the sizing, understand the political big players that are moving things around at kind of corporate HQ level and get attached to those projects politically and in some cases technically, and it's amazing what you can accomplish there.

I'll give you a silly example of deduplication. Every day let's say you're a 10,000 person company. Every day someone is opening a tab on Google Chrome. And that, let's say, goes either to the internet or google.com and your DNS and maybe [00:42:00] your firewalls. And your proxies are capturing that times 10,000 times, say a hundred times a day, right?

You have a lot of information that's completely unnecessary because we don't need to know that they went to the internet. We want to know what they actually did after the query, but we don't need to know that first one. So you could do that as an example of a silly thing that everybody can understand. Despam the pipeline coming into your SIEM and to your storage environment.

So a lot of it's just looking at your biggest sources, understanding where the efficiencies are. Print out several logs, physically print them out and just look at them and then put 'em in a text file and start removing things and say, okay, how many percentages did I leave out because they're noisy and they're unnecessary.

So little things like that can save 5%, 10%, 15%. Moving stuff from JSON to Parquet. I've seen, depending on the key value ratio, save 15 to 87%. and that when, again, when you're looking at terabytes or even petabytes of storage and net new every day for X number of months or years, this stuff really [00:43:00] adds up.

So you have to understand your political layer, your big transformation capabilities, your big bottlenecks, which typically involve a water meter financially or technically those up on a big whiteboard, document them and attack one or two use cases that will allow the queries to go faster or to be more. And allow you to shape that bandwidth before it gets into the long-term storage and into the SIEM via just doing some basic stuff like glue on s3.

[00:43:29] **G Mark Hardy:** Well, that's a fantastic amount of information there in a short period of time, and I've almost wanted to go back and replay a lot of the advice you had kinda unpack that. But we're, we're at the end of the show and so I can't keep going on with this. But if folks want to get in touch with you, what's the best way to get a hold of you, Gal?

[00:43:44] **Gal Shpantzer:** Hit me up on info@securityoutliers.com terrible website, great consulting, but also you can interact with me on LinkedIn or on Twitter at Shpantzer.

[00:43:53] **G Mark Hardy:** That's @shpantzer; at S H P A N T Z E R. So Gal, thank you so very much for being on the show, for talking to us about [00:44:00] the concept of what a lot of us, in my opinion anyway, have taken for granted. But looking at the idea of data engineering, taking that pipeline, trimming it down at the beginning, making sure you're only grabbing the stuff you need, making sure that we can do inline analytics.

Getting it to the, not just the SIEM, but other locations we need to, being able to answer queries, either real time or go back and recover them. And of course, own our data pipeline because that's important because you don't want to be paying the toll on that water meter forever and ever and ever to a third party.

So thanks again for being part of the show. Thank you to the audience for listening to CISO Tradecraft. I hope you found this a useful and valuable episode. If so, give us a thumbs up on your favorite podcast channel or give us a rating and share with everybody else. Let them know where you're getting your good stuff.

Until next time, this is your host, G Mark Hardy, and stay safe out there.