>> And then last, but very much not least, we have the folks who are actually trying to do this, the healthcare proof of concept.

And just as one point, we've heard from one stakeholder who is perhaps a little skeptical of the multi-stakeholder meeting model, he said a proof of concept is worth a hundred stakeholder meetings, maybe even a thousand. So we're really happy that folks are rolling up their sleeves and getting ready to do this.

And Mike from New York Presby is going to give us an update, possibly with help from Jim.

>> Wait a second while I enter the password.

Do you want to give a history? Jim, do you want to sort of a very brief history while I do this?

>> Sure. We have been working in this group to -- from the very start, we said when we met together back in July of last year, we said okay, these are a lot of good things that need to be decided, a lot of work that needs to be done, but let's actually show that we can do it, and let's get some facts on the ground.

And so that's why we started the healthcare proof of concept. It was originally the idea of Jennings Aski from New York Presbyterian. And we joined and said yes, let's get this done, let's show that there is real value in this whole process. So please, Michael.

>> Thank you. My name is Mike Dittamo. I am the Information Security Risk Manager at New York Presbyterian. And I'm certainly presenting on behalf of Jennings Aski who could not be here today.

So, quick objective statement. So this proof of concept is a collaborative effort across both HDOs and MDMs. And a lot of the decisions and determinations that went into this came out of all of the great work that was done by the other working groups.

Our objective is clear, to essentially prove out the successful feasibility, usage and value of this SBOM across cross-sector using a standardized format and process. And I'll go into detail.

So we have three primary project phases. Definition, execution, and conclusion. And all of these have subphases as well.

We are right about at the end of the definition phase now. And that's where we define the use cases that we are going to incorporate into this proof of concept. And also identify the data formats that would be used. As you've heard, there are a couple of different data formats, the SWID and SPDX.

As we approach the execution phase, we will actually prove out those specific use cases. And then you'll see the arrow going back. We can also certainly go back and refine some of the determinations that were made in the definition phase.

In the execution phase, we have identified the participants and observers, recorders. One of the deliverables we have is a questionnaire that will be filled out by the individuals partaking in the proof of concept.

And that's going to enable us to capture a lot of the important information that will go back into that feedback loop.

And then finally, the conclusion phase, which is going to be the preparation of a report and publicly available information that will be shared.

I will just skip ahead a couple of these. So the use cases. These are very high level, as you can see. The first one is procurement. So as part of the medical device

procurement and certainly as part of the risk analysis and assessment piece, this information is going to be proven to be very valuable.

You know, you cannot really assess what you don't know. To look down into those dependency trees is going to give a great picture and a great process as to how to truly assess the risk. Potentially identifying the how's, the where's, not just that surface level risk assessment.

Additionally, there is an asset management component. One of the things we're looking to do with this is to consume the SBOM information into CMDB type inventories and to be able to query that and have that visibility at all times which certainly plays into the risk management, vendor management, and overall vulnerability management of that particular device and all of its dependencies.

As I stated, these are fairly high level. Within each of these the group has also defined some more granular use cases, personas and things of that nature.

So who are the participants. I will read these off. The HDOs here are New York Presbyterian, which is where I work, Cedars-Sinai, Christiana Health, Mayo Clinic, Mass General. And then the MDMs are Abbott, Bayer, Philips, and Siemens. And I believe we have representation from a lot of those folks today.

This is a bit small so I apologize for that. But determining and defining the scope of the POC. We had two categories. So it was a binary in and out depending upon the specific item.

I will go through each of these briefly. The first one was comparing the CBOM, the Cyber Security Bill of Materials which includes hardware components versus the SBOM, which is strictly the software components.

We determined that the SBOM was going to be the focus of this POC, not the CBOM. Primarily because we wanted the proof of concept to be attainable, to be successful. We didn't want to lose track of the primary issue of the POC. There were some complexities with the CBOM as well.

Another thing was around the determination of the canonization of an acceptable format. While we may lean on one of the two formats identified, any work that comes out of the POC would not be an endorsement of one of those particular data formats. They would both be considered feasible. We are not comparing them against each other. We may just rely on one for purposes of the concept work.

The one that we wanted to focus on was the value of conforming to a standard. There is a value to have a standard across all device manufacturers, and all consumers as well. To obviously spiral out and have a number of standards is not going to be sustainable. It's not going to be feasible. It's not going to be digestible and the value is lost.

The inclusion of vulnerability information. So this is an interesting one. We talked about this quite a bit. The determination was made to actually keep this out. At least out of this 1.0 version. The information in the SBOM can certainly be determined to have the vulnerability information by the consumer, but to actually have it alongside those particular attributes was determined to be left off for this.

The dependencies, this was briefly mentioned in a previous earlier discussion as well. We give it a best efforts optional type tag here.

You know, I think it was said that yeah, I know what that first component has but I don't know what's beyond that. Obviously, it would be ideal to have everything, that

entire supply chain from back to front there. Not always going to be possible. I'll actually talk about a little bit about that latter as well when we go into software of unknown providence.

The global unique and component identifiers. So this is another one that we did not include in the version 1.0 of the POC because it produced some problematic issues. Ones that were certainly needed were vendor name, version down to build number. And then another one that we talked about was the context. And I like that comment, yeah, it's in here, but don't worry about it.

So this is where the compensating control or the non-exploitable potentially vulnerable type gray area comes in.

I don't know if you can actually see it on here, but we actually had an X there, that was struck through. Initially it was one of the things we wanted to talk to and to capture to say yes, this has this particularly and possibly vulnerable component, but you don't need to worry about it because of this. Because we're not making these calls. Because there is a compensating control in place.

We did change that out for this particular 1.0, once again, due to complexity and to keep the POC moving forward.

The last two, or I should say the next two was the delivery method. So we do want to have this to be something that could be pulled, whether it be an outbound web call to be accessible.

The device sits on the network. It does not have an SBOM, that information can be pulled in. Maybe there is an API that can also can be used for that data access as well. It is likely going to be fairly static for the most part. There could be potentially changes as well. When those changes are made, obviously subsequent calls would be made.

>> Michael, can I jump in on that point?

>> Absolutely.

>> This is Jim Jacobson from Siemens Health.

We decided originally to have this be a very smooth process where you could just make this call, retrieve the SBOM through some API that would be exposed.

What we ultimately decided was that that wasn't valuable to the proof of concept. That our goal was to see how the data are produced and consumed rather than to show that we have automated all mechanisms in place.

>> Thanks for bringing that up. For the proof of concept, we may manually grab that file. Obviously, ideal if it's automated, but for the proof of concept we're just going to, manually grab those files.

And lastly here, I know it was mentioned, the previous one seems like kind of a no-brainer, but machine readable format.

I know it was mentioned that it could be in PDF or something like that. We have the machine readable formats available and that's certainly a requirement for this to be a success.

So some recent activity, and some plays into what I will go into on the next slide as well. Because we will be transferring or are transferring a level of data between the HDOs and the MDMs, we wanted to ensure that there is nondisclosure in place, that all parties are comfortable with the level, types, and volume of data that are being transferred.

So we are in a bit of a, I don't want to call it speed bump, but let's say a bit of a crossroads to kind of get this confirmed with our respective legal counsels or approved by our respective legal counsels. I think it's something we can probably discuss later today on status and timelines.

We also -- the initial draft of the observation collections, that is the questionnaire I referred to earlier in the presentation. That is a fairly robust questionnaire that was an effort of collaboration across both the HDOs and the MDMs to identify all of the upfront questions that we're likely going to have as we go through this proof of concept, a data gathering exercise.

That's dynamic. It can change as the POC continues. But it's something that we wanted to get up front in front of everyone, everyone's buy-in to ensure that the objectives and questions were identified up front.

The MDMs are developing the SBOM draft with a couple of completion dates here. We currently have end of April with the Health Delivery Organization studies to follow shortly after, at the end of May. So if you go back to one of my earlier slides there, that's in that execution phase. We're just kind of in that phase right now.

>> The way I describe that is we are approaching the climax if not the dynamo.

>> Yeah, this is the exciting part.

And then lastly here, to finalize the data format selection and work with a product list now with the MDMs to develop where the SBOMs will be available.

So the potential concerns. And I know I've mentioned this briefly before, but the intent is not to necessarily choose winners. We're not looking for canonization. We're not looking to pit this against this or say we have an endorsement for this.

What we are going to do is to just prove out the value that a standard should be considered or needs to be considered and that there is value behind that on both the MDM and the HDO side.

The second one plays into that NDA comment I had earlier where there is a degree of confidentiality that is assumed by both parties. We want to make sure that everything is -- is kosher, so to say, prior to creating any type of public report. So there's going to be a lot of talk about that as well prior to doing so.

I'm just going to skip over the next one. This was really about a determination of roles and participants. We feel as though we've addressed that appropriately.

And the last one here, because we are going across a number of Health Delivery Organizations and Medical Device Manufacturers, we wanted to ensure there was no conflict of interest or any type of business relationship issues that would arise as part of this. This is truly a working group, and the business relationships exist outside of that.

And the last two things, these are two to open items that we did want to highlight after a recent discussion. The software of unknown provenance. So this is one that plays into the dependency tree and was brought up in a recent call.

There are issues where -- or there are situations, rather, where we don't know or the vendors do not know. There is a software component, and there is no way to get that information. It could be due to bankruptcy, could be just due to a legacy component that does not have that information any more.

We discussed on how to handle that. Is there a place holder? Is there as much as you know, is there an inference? We're still kind of discussing that to determine the best way to handle that, but it's certainly something that came up as an open item.

And the last one is to investigate the better ways to avoid inconsistencies in software component IDs in the SBOM document.

Jim, I'm not sure if you had anything else to add on that?

>> Just a couple of points. In case it hasn't been clear, one of the main focuses of our proof of concept is to shake out problems in both production and consumption of SBOMs so that we can inform the rest of the groups and help them understand what still needs to be accomplished. That's one point.

Another point is what we have discovered in producing the SBOMs is that there is no tool automatically magically creates what we need to create in order to satisfy the use cases that we're talking about.

That it is -- in some cases, tooling helps, but a lot of it is done manually. That part of it can be automated internally at some point or even better automated by tooling which is available externally to everyone. But right now it does still require manual manipulation.

>> Could you clarify that a little bit? Is it there is no tool or tooling, or there is no language universal one? Because I have built tooling in Java and Jenkins that spits it out.

>> Right, right. What I'm saying is it is up to the MDM, the manufacturer to determine what's the best way to get through this proof of concept. Do we take advantage of something that already exists? Do we need to add some special sauce at the end to meet the use cases that we've determined?

But there's nothing universally available certainly. And it just depends upon how much we want to automate it in order to get the proof of concept.

And, finally, the HDOs that are executing the proof of concept from the consumption point of view, most of them are actually going to be integrating this information into their systems or to attempt to integrate the information into their systems. Although in one case it's going to be a desktop exercise to walk through. But I'm really excited that we'll get the opportunity to see the consumption side in practice as we execute.

>> Just to quickly kind of expand on that. I think I may have glossed over it a bit.

There's consumption in the CMDB as well, but there's also a vulnerability management component as well. So just to have the visibility of the components is the one piece. To be able to actually have downstream systems that identify those vulnerabilities outside of just an IT asset inventory sense is also valuable. We'll be doing that as well.

>> Sorry, Art Manion at CERT. The discussion about confidentiality and NDA, I would like to maybe flag for this afternoon. But specifically, could I ask is this some, clearly a concern.

>> Yep.

>> Is it the concern sort of for the pilot proof of concept, or is this perhaps a longer term concern of that sector?

>> It is for the proof of concept.

>> Yes.

>> The reason is we're producing data which are not official in any sense, right?

>> And also you have to remember this is an industry where there are very clear rules about what data can and can't be public. And so there's a -- the lawyers needed to be assured that no, this isn't about patient data.

I do want to point out that in terms of progress that this working group is making they have been working really quickly and doing amazing work. And then they said let's write an NDA, and then the lawyers just take a really long time to get things done.

>> Yeah. Such is life.

>> But very good work.

>> Any other questions? Oh, I think we have one more.

>> This doesn't come up too, too often in just because of the Cisco question earlier about existing tools and tooling.

One of the really nice things we talked to a lot of hospitals about is they can't even do a normal vulnerability scan without breaking or really hurting a lot of these fragile devices. Such a short list. Maybe they can do a targeted interrogation of a handful of devices as opposed to all.

So it's really more about the Brownfield risk of traditional tooling maybe containing the blast radius and scope, some of that harm.

>> So this is Elliott from Cisco again. You very carefully partitioned the space there between Greenfield and Brownfield. I won't comment too much on the Brownfield. We see that -- well, I'll say we see that problem across sectors especially in industrial actually that you hit the -- you do a scan and screw up the timing of a device and somebody can really be hurt.

>> Um-h'm.

>> I Know that Siemens is worried about this across the board. I worked with Siemens on this as well and a lot of the other manufacturers.

There is some new work that is going on on the Greenfield which is the remote attestation work that is going on in the IETF, (indiscernible) is the name. And the other is security update of IoT devices, SOUP. But that is something we can talk about later.

>> This is Jonathan. To echo what Josh brought up about devices and vulnerability management solutions that are run against medical devices could create some problems as well as other components within the industries.

As we are talking and bringing up the thought came to mind is as we start to develop SBOMS, is there any concern that once you transfer this level of knowledge to the consumer, is the manufacturer -- and me being a medical device manufacturer -- so the responsibility is not discontinued at that point.

But from an industry perspective, is the thought -- is there a thought or a concern that I'm now transferring the responsibility of maintaining a device and its hygiene as a manufacturer or a supplier. And I'm now transferring that over to a client consumer because I've now given them what they need to know to maintain the product. And this is kind of industry-wide topic or point of view.

>> That may be a bigger thing to explore this afternoon.

>> Okay, great.

>> It kind of plays into what we talked about this morning a little bit, yeah.

[OFF MIC]

[LAUGHTER]

>> I'll bite my tongue on that one.

>> Thanks, Mike. This is Steve Abrahamson, GE Healthcare.

In your proof of concept, are you looking at new devices, you know, devices currently on the market? Are you also going to be looking at the dreaded legacy devices?

Because we found as we're -- you know, we've created a library of software build material for our devices. And it's rather difficult for some really old devices to generate a software build material. That's where some of these automated tools might come into play.

One thing that Jim and others are alluding to is that for newer devices, let's say devices currently in your portfolio that you're selling, you would like to think that the engineering team for the manufacturer would know what software is going into it, or at least even if it is SOUP they know that they have put some SOUP in there. I don't like the term SOUP, but we use it.

So we like to think that the engineers actually know what they're putting in the product. And then furthermore, because we expect that the revised FDA guidance is going to require a software build material or CBOM to be submitted with the 510(k) documentation, we can't necessarily rely on a tool to develop a software build material unless we have validated that tool, which is something that we're not in a position to do.

So we have to kind of marry up the use cases for the automated tools where maybe we don't have the engineering knowledge still readily available to identify what's out there versus newer products where we would expect to have the correct engineering discipline to actually know what software is in our products.

You could go back -- maybe going back a few years to Microsoft where I would even make the case that we should know what the code does. You know, some of the older versions of Windows they had code but they didn't know which code did what basically, but that's another story.

So I think what we found is that we rely on the engineers to select the software components from a menu that drives a consistency. We don't want people making up their own terminology.

And then just a follow-on to that comment. One thing you indicated was the globally unique identifiers was a very hard problem. Were you getting to the terminology or the nomenclature for specific software components with that?

>> Correctly, exactly. Yeah, they could have a variation of names and things like that to have the specific unique identifier for that particular software component is something that could be quite difficult to tackle.

>> So comment on that as well. And we can maybe this afternoon get more into that.

But what we found in implementing a SBON system in GE Healthcare, we are a large manufacturer and we have some other large manufacturers in here, even within our organization if we don't have a defined software components that the engineers would select from in building up the SBOM, one will -- one is going to call it Win 7, somebody's going to call it Windows 7.

>> Exactly.

>> Especially like 16. I'm just making up that number, but there's many different flavors of Windows 7 that have to be specifically identified.

So even within a given manufacturer organization, I would strongly suggest that you rely on a common menu of software items, and I would be happy to share the way that we do it. And then when you're starting to aggregate SBOMS from multiple manufacturers, it becomes even more critical that they are all using a common library of

component nomenclature as much as possible, whatever level of uniqueness you want to use. A common library would make it much more useful. I yield the floor.

>> If I can comment on that. We have decided to do the best job we can. So I agree on this side, on the point of making sure that as a manufacturer that you're consistent across the board, right.

Then we came to the decision point should we make sure that between manufacturers we're using the same terminology. And what I think we'll see in the execution of the proof of concept is that consumers are going to experience variability in the same sense that will represent reality for some time to come.

>> So I'm going to -- this is an excellent segue for the last few minutes before we take our lunch break. So, first of all, I want to thank Michael.

>> Thank you.

>> Michael, who is fairly new to this initiative.

>> Yeah, a few weeks.

>> And really appreciate it.