IETF SACM WG Virtual Interim Meeting

June 15, 2016

Summary

This was our second virtual interim between IETF 95 and 96, and we are very much on track for the way forward. We are making progress on the information model, we have adopted a software identification draft, and we are close to completing work on the vulnerability draft. Our way forward between now and IETF 96 in bullets looks like this:

- Complete information model merge with a draft update no later than July 8, 2016
- Submit the requirements draft to IESG
- Continue driving the software identification work with a draft update no later than July 8, 2016
- Complete the vulnerability draft WGLC before IETF 96 with the intent of not progressing that draft to IESG immediately, but giving the draft a "hard review"

Thanks to everyone who participated in the meeting, and who participates on the list.

Notes as Submitted by Charles Schmidt and Danny Haynes

Agenda bashing

[Karen O'Donoghue]: Any agenda bashing?

[Danny Haynes]: We are going to try to split up the Information Model (IM) Update session into two sessions: (1) IM Update and (2) Selecting a Data Format for an Endpoint Information Data Model.

WG Status

[Karen O'Donoghue]: We are still making progress on the IM and a new draft was posted on June 8, 2016. We still need to get the SACM Requirements to the IESG for review. I haven't done that yet, but, I will try to do that this week. We also completed the call for adoption on the SWID M&A for PA-TNC I-D draft on June 9, 2016 and it was adopted. We have also made progress on the Vulnerability Assessment Scenario I-D. Overall, we are making good progress.

Vulnerability Assessment Scenario Update

[Danny Haynes]: Agenda.

[Danny Haynes]: Status. After adoption in April there were a handful of open issues and edits. That spurred list discussion. Since then we have worked many to completion and they were incorporated in the draft published on 6/8

[Danny Haynes]: Summary of resolved issues.

[Danny Haynes]: What to do with this I-D? We didn't really settle on the end state for this draft? We created this to help us focus by breaking our problem space into more manageable pieces. Will it stay a WG draft? Will it become an RFC? To consider: will other WG find this valuable (if so, maybe publish as an RFC)? Also, since terms change, will that impact us? Any thoughts?

[Ira McDonald]: I run contrary to the current IETF trend. Documents record rationale and through process. I am in favor of publishing as an informational RFC. Maybe add a note in the abstract that terminology is evolving.

[Adam Montville]: I agree – publish as informational?

[Jim Schaad]: Given that we would reference a Terminology draft, I wouldn't worry about the terminology.

[Ira McDonald]: Don't need to point to a specific version.

[Jessica Fitzgerald-McKay]: One open issue was moving some terminology from this paper and into the Terminology draft. Should we move but also keep in so one doesn't need to search for the right terminology?

[Ira McDonald]: I like that approach. Likely that this will be published before the terminology draft is published. I hate reading documents that require you to import other docs.

[Adam Montville]: One counterargument: which definition is in play? Would argue putting everything in terminology draft so there is an authoritative reference.

[Ira McDonald]: In favor of putting things in terminology.

[Jessica Fitzgerald-McKay]: As am I. Terminology draft has all our terminology. It will be the reference that is correct. But I don't want to have to read multiple docs to understand. If updates are needed, can do that later.

[Charles Schmidt]: Maybe park as a WG draft for now and promote to RFC later.

[Ira McDonald]: I like this. We want to show thought process. But parking for a while until processes are done.

[Adam Montville]: Are we at a point where we need to decide this?

[Charles Schmidt]: We are trying to pin down the final plan for the document. Sounds like that plan is to create an informational RFC, but that there is no need to rush to publish it.

[Danny Haynes]: Yeah. Don't need to decide now, although some terminology decisions impact. Once we make decisions, they will be stable, but at the moment we are still changing.

[Danny Haynes]: Seems to be consensus to bring to an information RFC at some point but not now.

[Danny Haynes]: Issue #1 – Dealing with terminology and whether we want to promote some to the Terminology ID. Sounds like we can keep them in our WG draft and also put in Terminology Draft without problem. Maybe that is the best approach? Don't have to pull them out. With that in mind, maybe we can go through some quick definition changes made to pull them up. People can decide if they like and can pull up to Terminology draft while keeping in the WD.

[Danny Haynes]: Supplemental Collection – Recently there were some new edits. I think this term could go in Terminology draft.

[Henk Birkholz]: This definition basically says supplemental collection is a specific version of collection. What is implied is the collection is a process that can reiterate. Supplemental implies that this is not the first collection. I see here that this is more like a virtual definition – if you are missing something, go collect it.

[Danny Haynes]: Yes.

[Henk Birkholz]: Maybe add "workflow" or "process cycle" and use the existing definition of collection.

[Charles Schmidt]: So you are suggesting just a general "collection" term, and then clarify the process flow to note that collection can occur at different times?

[Henk Birkholz]: Yes. Maybe not define a special collection and instead just define the workflow.

[Charles Schmidt]: As long as the concept of pre-assessment and subsequent assessment activities are understood as supported, I don't require special terms.

[Henk Birkholz]: Are you familiar with the Discovery task?

[Charles Schmidt]: No.

[Henk Birkholz]: It is similar.

[Danny Haynes]: Take that as an action.

[Danny Haynes]: We'll bring the rest to the list and bring them there. Summarize what we talked about and bring in main points. They should go pretty quickly.

[Danny Haynes]: Next steps: Close out remaining issues and focus on solutions such as SWID M&A.

Information Model Update

[Danny Haynes]: Agenda

[Danny Haynes]: Status. Most changes in the IM revolved around the work to merge the I-D and WG IMs. I included information in SACM statements and content elements to try to build discussion. All changes in the 6/8 draft.

[Danny Haynes]: Statement and content element. Statement is effectively an envelope to pass information between SACM elements. Contains content elements, which contain metadata about that element. Seemed like a nice way to organize information when exchanging between SACM components. One thing about the nesting structure – as this information passes from component to component, assuming the model is part of the payload, you can get some tracking of where information went over time as it goes through the lifecycle. At the last VIM there was a question as to whether there needed to be explicit statement and content elements or does it just add unnecessary complexity.

[Danny Haynes]: Mapping – To answer that question, looked at the content elements to see which were supported by SWID M&A and OVAL. For SWID M&A: for unique identifier, you need to rely on the NEA server, when it receives messages from the endpoint, it assigns an identifier before it passes it on. Data origin – that data is built into header information. Creation timestamp – SWID M&A there are a couple of options: when did the posture collector believe the change occurred vs. time the endpoint assembles the information. Publication timestamp – time the data is published. Type of content – we need to go back and revise. When people think "type of content" people are thinking type of content. For SWID M&A, that would be a SWID tag. Need to update definition. But this is supported by SWID M&A. Creation timestamp of content – is this creation by vendor, dropped by installation, generated by package database, last recorded event? Have to select something there. Data source of content – TLS certificate or SWID 2015 device identifier. Bottom line: SWID M&A could support the data in the content element, with some decisions to be made.

[Danny Haynes]: Mapping (OVAL) – Little trickier. No concept of a global unique identifier. On origin – you can get some general information, but nothing that tracks to a specific instance. Creation timestamp – Does have a timestamp in Generator element, but it might not be accurate. Publication timestamp – same deal. Creation timestamp of content – just at the document level. Doesn't break down by what is created. Data source – some info about what endpoint, but it is just a collection of attributes (IP, network interfaces). Would need some sort of matching algorithm. If an extension to PA-TNC was developed for OVAL, it would pick up many of the items discussed earlier.

[Danny Haynes]: Decisions to be made – Maybe don't worry about OVAL now; just think about SWID M&A. Comes down to: do we want solutions to have to carry this metadata in the payload of SWID M&A, or can we say, if you publish this system to the SACM ecosystem, you need to hit X requirements? That is the big decision we need to make to move forward. Thoughts?

[Ira McDonald]: I was wondering – there is a trend for saying "just meet these requirements", but that kind of weakens SACM because it makes it pretty obscure as to whether the elements are there since they have to be mapped. I kind of lean to explicit inclusion.

[Adam Montville]: Which question are you looking at now, Danny?

[Danny Haynes]: Bullet 2. Need to figure – we have these constructs: do we need constructs to be explicit in the solution, or is it something we let the solution deal with. SWID M&A – can get this information. It just needs to be done some way. Would need to be some documentation.

[Jim Schaad]: I don't have the same view of the information model. If these are pieces of information that someone is going to want to see, they need to be there. I'm less worried about how the data model represents them. I want to see an information model that I can think about writing SQL queries against. That is more important to me.

[Danny Haynes]: Do you mean the actual information model, or the data that gets represented.

[Jim Schaad]: Information model.

[Danny Haynes]: So you are saying, it might be nice to have these constructs, but the implementation doesn't need to break them out.

[Jim Schaad]: Yes.

[Adam Montville]: Agreed. The reason we wanted this model in the first place was to make sure we include specific things. If we use these structures, it requires data to be rewritten to be used. Not sure I like that.

[Ira McDonald]: They need to be in the IM. Once they get into some query-able server, the content elements don't need to be explicit.

[Jim Schaad]: That is a data model problem.

[Ira McDonald]: Yeah. I agree – we don't need extensions to NEA/PXGRID for metadata structures. Need to be able to derive the metadata.

[Adam Montville]: Agreed.

[Henk Birkholz]: This is the right way to go. Don't enforce existing DM to enforce hierarchies. But the solutions of NEA are involving always target endpoints. Closer to the connection task. Every communication after that will not use NEA. NEA collects from endpoints, not connecting general SACM components. There will still be a gap in common intercommunication of components. Since that is still to be defined, maybe use the content elements. But don't force on existing structures if the high-level mapping is possible.

[Danny Haynes]: So action item is that those constructs are conceptual.

[Jim Schaad]: I will have a lot easier understanding after I see IPFIX structure.

[Danny Haynes]: Understood.

[Danny Haynes]: With that, we have these two questions regarding creation timestamp. Is there any preference on whether this is the time that the Posture Collection believes the change occurred, or should it be when the message was published? (I.e., when received by server – transmission time.)

[Jim Schaad]: Also clock skew between device and network. I would expect to see both of those timestamps. If the device is capable of keeping track of time, it would have time of collection and the time it is put into SACM. So you can look at clock skew, order.

[Ira McDonald]: Also things collected at the device and deferred for forwarding.

[Jim Schaad]: Yes. And device could republish something even if no change because that is interesting.

[Ira McDonald]: Agreed. As it moves on through SACM structure, timestamps should occur as it moves.

[Jim Schaad]: Especially if someone transforms data.

[Ira McDonald]: Agreed.

[Danny Haynes]: I think the next step – many of these same questions apply to the content timestamps as well. I think we need to capture those in the information model.

[Charles Schmidt]: Would there be real operational differences that come from having all these timestamps.

[Henk Birkholz]: My perceptions: vendors that sell these products, tend to look at detection over time. So yes – if they correlate over time to detect anomalies over that. There is a high need for sophisticated features that are almost never produced. Because often you don't know what the timestamp means. Never been done. At least we give the option of doing that. Highlighting the different varieties highlights the need.

[Danny Haynes]: Maybe what it is: if we get these defined, maybe we only pick a few that are require for support, but allow others to be optional. So others can support.

[Adam Montville]: May go beyond vendors. Internet response capabilities – if it matters when dropped on endpoint or when published to a NEA server.

[Charles Schmidt]: Seems reasonable. I just was wondering why we needed 6 timestamps – would they be used?

[Danny Haynes]: Next steps. I think we were able to get some consensus. Will confirm on the list. Make clear that SACM statements and content elements do not need to be explicit as long as the data is there. For IPFIX – will try to get a proposal out soon. Also, Adam noted that a big chunk of the information model is in the appendix as well as sections 5-7. Need to clean that up. If people want, we can create a text file in the repository and put the information there. That will lighten up the document. Would like another updated version before the next IETF meeting. Big target for that – do some work with statement content elements, IPFIX, and Section 5-7. (Appendices nice-to-have.)

[Danny Haynes]: The discussion on a data format for an endpoint information data model will be turned into a mailing list discussion.

SWID M&A for PA-TNC Update

[Charles Schmidt]: For the agenda, I want to first discuss the milestones because it is always good to show progress. I would also like to discuss the data model for communicating software inventory information. This is the question Gunnar raised about how we are going to represent information we are collecting in the messages that get moved from the NEA Client to the NEA Server which is getting the information from the endpoints and making it available to the SACM ecosystem. Then, we will have some time to talk about next steps.

[Charles Schmidt]: The first milestone is that we submitted proposals for the technical changes to the mailing list and most of those changes are in limbo because they have dependencies on the data model although we had lots of good discussion. We also published a new draft (revision -01) that simply removed the references to the IF-IMC and IF-IMV specifications. The next milestone is that the I-D was adopted as a WG draft. I think having myself and Gunnar as editors will be good because I am the original author of SWID M&A for IF-M and Gunnar is not so it will be good to have both perspectives in this discussion.

[Charles Schmidt]: There is a lot of information on the upcoming slides, but, I want to emphasize that everything is up for discussion and nothing is final. Just let me know your thoughts and we will discuss them and revise the work as necessary.

[Charles Schmidt]: First, the data model for SWID M&A is the ISO SWID Specification. Specifically, 2009 and 2015. What is up for discussion is, do we keep this as the data model, do we use something else, do we extend it, do we develop a profile on it, or do we create a completely new data model. The data model is currently in ISO, but, we need to see where we want to be. There are many open issues and there seemed to be consensus around them such as the decision to record the application location, the decision to record the source of the tag, but, all of that boils down to a data model question, but, actually implementing all this means we have to actually have a data model so most of these things ended up getting folded into this bigger question.

[Jim Schaad]: Should we be talking IM or DM at this point?

[Charles Schmidt]: We should probably be talking IM at this point.

[Ira McDonald]: That was the comment I was going to make about jumping straight to data model.

[Charles Schmidt]: Here is the nominal data flow for what the SWID specification does and what I think we were talking about on the mailing list, but, I wanted to make sure we were all on the same page. On this endpoint, there are multiple sources of inventory information: one may be the tags present on a file system, another may be a package manager (rpm, yum, etc.), and another source might be a software inventory scanning tool. These sources go to our PC and are normalized to a common data model in compliance with the IM and then it gets sent to the PV on the NEA Server where the NEA Server makes it available to the SACM ecosystem as a Broker. This is where things like the unique identifier for a statement that Danny mentioned come into play. The NEA Server would assign that. Are we all on the same page up until this point?

[Henk Birkholz]: This implies two things that the PC is not a SACM Component and that the NEA Server is a Provider of information and not a Broker. Is that correct?

[Charles Schmidt]: I think that is correct. Can you elaborate more on the implications of this?

[Henk Birkholz]: The PC in this situation can never talk to other SACM Components and always has to go through the NEA Server first via the PA-TNC binding because there is nothing else. Then the NEA Server is the first SACM Component because it can provide this information to the SACM domain to be consumed and discovered by a Broker and such.

[Charles Schmidt]: The PC always talks with the PV and the definition of a NEA Server is something that hosts a PV. You could have multiple NEA Servers that were all SACM Components that the PVs were communicating with. I am still not sure why it is not a SACM Component.

[Henk Birkholz]: It is because every SACM Component has to be a provider of information or a consumer of information to the SACM domain and this is not. It is a collector, but, not a collector as defined by the SACM terminology. It is a specific collector that is only talking to NEA Servers.

[Danny Haynes]: Is your main point that because it cannot directly communicate with other SACM Components (it has to go through the NEA Server), it is not a SACM Component?

[Henk Birkholz]: Yes, I think so. A SACM Component is a provider or consumer of information from the SACM domain and this does not. The NEA Server does though. A PC is a software component running on the target endpoint itself so you planted it there probably because you own the device.

[Charles Schmidt]: Right.

[Henk Birkholz]: This internal collector is not a SACM Component. There is an open question as to whether it is and this is an example of it is not.

[Charles Schmidt]: Henk could you give me an example of a SACM Component that could not possibly have a PV on it?

[Henk Birkholz]: I think for example a SACM Component that would be tasked with provenance of data would not have to have a PV on it.

[Charles Schmidt]: It wouldn't have to. It could get the data indirectly, but, it could.

[Henk Birkholz]: But, it doesn't need the data, it just needs the references on the data. It will just store it lightly and only the references are important here.

[Danny Haynes]: I don't think there is anything preventing from a NEA Server having multiple SACM Components.

[Henk Birkholz]: I didn't mean that. Endpoints can have multiple SACM Components. I am just saying that if it cannot directly communicate to the SACM ecosystem then it is not a SACM Component. You always have that indirect communication through the NEA Server. Since it is not defined anywhere, it is not wrong, but, it is just something we need to address.

[Charles Schmidt]: What do you think the implication of a PC not being a SACM Component? A PV is a SACM Component correct?

[Henk Birkholz]: Yes, PV is a SACM Component. The implication for a PC is that it can't send raw data to the SACM ecosystem. If you want something else, you might need additional collectors on the endpoint. This will result in many collectors on the endpoint.

[Charles Schmidt]: Two good points out of this. (1) in your definition of a SACM Component it sounds like it is a grid (i.e. many-to-many). NEA is not a grid. It is a consolidation (i.e. many-to-one). Regarding many collectors, NEA supports many collectors of different types of information, but, all communicate through the same NEA channel to the NEA Server. So yes, in that regard, you will have multiple PCs, but, all are using the same protocol stack and using the same components. Also, all that PCs have to do individually is identify information that they want to collect and then package it into a message that can be sent over that channel. Other collectors will have to do that as well, but, I would say that is an advantage because all are using the same communication framework.

[Henk Birkholz]: This sounds good. I am fine with that because it is one way. It would be good if we could publish data as-is.

[Charles Schmidt]: The NEA Server can also do one better in that it can provide additional metadata.

[Ira McDonald]: To fit in the real world, this will always be the way it is done; a gateway to SNMP, NETCONF, and other protocols. I would suggest that the PV and NEA Server normalize the collected data into a normalized format for SACM. That is, have the NEA Server be the normalizing relay rather than an interpreter.

[Henk Birkholz]: Yes.

[Charles Schmidt]: Any other questions on data flow? <no>

[Charles Schmidt]: A few other assumptions on this data flow is that we are not constraining sources. We are not going to say you need to use only one source. Our specification should be able to take what is given and convey that to a NEA Server and convey it to the rest of the SACM Architecture. That is the first point. The second point is that, in this flow, each piece of data comes with a single source. Right now, the specification assumption is fairly loose and the source has to be consistent with this information and detect changes. As far as change detection is concerned, there are different ways and I don't think we need to decide this now, but, we have to decide where the normalization occurs. Is it part of the source? Or, is it part of the PC. I don't have an answer now, but, it is something that we should discuss.

[Ira McDonald]: By source, do you mean endpoint?

[Charles Schmidt]: Yeah.

[Ira McDonald]: I don't think source should have additional functionality. I think that's a good way to eliminate using sources.

[Charles Schmidt]: That is a very reasonable point. Okay, then the normalization would be done at the PC which makes sense because the PC would be required to have some channel between itself and the source and clearly has some coding that is source specific.

[Jim Schaad]: I think the normalization could also occur at the PV as well.

[Charles Schmidt]: This gets to the question that Gunnar raised on the mailing list that whether the communication of the PC, the messages transmitted, should have a common data model which would imply all normalization would occur on the endpoint whereas Gunnar and a few others on the list advocated for the normalization to occur as far out in the branches of the SACM ecosystem as possible.

[Henk Birkholz]: I think the discussion on the list may be about the SACM domain. As soon as it enters communication with other SACM Components, it has been normalized. I think the data model between the internal collector and the NEA Server, it is not relevant because no one else is going to have to look at that because you are only talking to the NEA Server. You can make up whatever you want there and NEA is there and it is concise and that's fine. So, having that, it is only important that you have the normalization at the very first point you are a Provider to the SACM domain.

[Charles Schmidt]: This aligns with my thinking as well. Gunnar was saying some sources are going to be endpoint specific and that you will need to know all the normalizations so why not put it in the same place (i.e. NEA Server).

[Henk Birkholz]: The PCs won't use guidance since it not part of SACM domain. There is no semantic difference between those two scenarios.

[Adam Montville]: First, I think I heard Henk say originally that the first point of provisioning coming from a SACM Provider is where the normalization would have to occur. So, however far out that Provider is, is where that normalization would take place. The second thing that I wanted to mention is that if we put everything central then you are going to run into implementation issues. I'd rather have the implementation of the normalization distributed among all the different vendors that we care about rather than having one vendor have to go out and do all that work separately. Or, a collection of vendors doing all that work separately. If that makes any sense.

[Charles Schmidt]: Adam, can you elaborate on that? I am not sure I was following you.

[Adam Montville]: If we look at the PV and NEA Server and if we say normalization is going to happen on that end. Then, there is one vendor who is going to implement this in the SACM domain and there are hundreds of different ways to normalize software identification data into a particular data model. It then becomes incumbent on that one vendor to support all those translations which could be burdensome rather than distributing that load. I'm not sure which approach gains better traction in the marketplace. I think it's harder to make one vendor do all that work.

[Charles Schmidt]: In general, not required though. In a lot of NEA/TNC implementations, there is a pairing. If you are an implementer, you write both a PC and PV.

[Adam Montville]: Yes. Maybe.

[Charles Schmidt]: Yeah, if that's the case.

[Adam Montville]: Right, okay, if that's the case. In an ideal world, the PCs are being written by whomever and the PVs are being written by whomever.

[Charles Schmidt]: Absolutely correct.

[Adam Montville]: Okay, keep going. Sorry.

[Charles Schmidt]: The other side is that if you have a situation with multiple vendors for different PCs (one vendor for Mac, one vendor for Windows, one vendor for Android, etc.). If they all go to a central PV, in the normalization procedure when coming from the PC, it may not be that all vendors normalize the same way whereas if it happens on the PV, it is a heavier lift for the PV, but, at least you get a more normalized normalization.

[Adam Montville]: I suppose. I don't know. From my perspective. Again, I am not talking as a chair. I should have mentioned that earlier, but, rather I am talking as a contributor. It seems to me like what I would really want is if I got multiple distinct entities implementing different components of the ecosystem, I want what goes between them to be well defined and deterministic so if that normalization, that we just talked about, is normalized one and that's the way it is normalized. Maybe that is an ideal that can't be reached, but, I want any of that data. This goes back to data-in-motion between components. We want that to be very specific. To do it in a very specific way and like what we talked about earlier about normalizing what does it mean to take a SWID tag, for example, or other information and "normalize" it into whatever data format we pick out. It is that mapping that is important.

[Jim Schaad]: I have two issues with saying it has to be done on the PC. The first is that I have some really lightweight devices that collecting a SWID tag and sending it off is something they can do, but, anything more than that they are going to have problems with.

[Adam Montville]: But, then, the collector isn't that. The collector will be something else from our perspective. I think.

[Charles Schmidt]: No.

[Jim Schaad]: Well, it's hard to say because it is sending data that can be validated

[Adam Montville]: It is sending data to another component that can do the normalization. At that point.

[Jim Schaad]: But, it's the PV that is doing that normalization.

[Charles Schmidt]: I think what Adam is suggesting is that when a component on an endpoint is extracting SWID tags and dumping them to another point which is then that becomes an external PC which does the normalization and communicates to the PV. Do I have that right?

[Adam Montville]: Yes, I believe so. In this diagram, it shows a PC that is actually on an endpoint, but, a PC need not be on an endpoint.

[Jim Schaad]: So, you are talking about a distributed PC at that point. That gets interesting.

[Adam Montville]: No, not necessarily. I am not sure I exactly understand what you mean by distributed PC.

[Jim Schaad]: If it can have part of the collector on the device and the other part off the device.

[Adam Montville]: Its collector may not necessarily be on the device. It can interact with the device to query it.

[Jim Schaad]: Right and it does that by talking to its component on the device.

[Adam Montville]: It doesn't have to be its component. We might be getting mixed up on terms there. If I SSH into something, do I have a component over there?

[Charles Schmidt]: I think there is some nuance here that will need to be examined. How do you deal with an endpoint that doesn't necessarily have a PC, but, is providing information that will eventually make it to the PV? I will try to make some references on that concept because you are absolutely right that this nominal data flow is not going to cover every type of endpoint.

[Charles Schmidt]: I put together some use cases that we want to support with our IM. We want to look at what is supported in the Vulnerability Assessment Scenario so we don't go into a bunch of hypotheticals, but, I think we also want to think of other use cases for inventory information so as not to paint ourselves into a corner. I had a few ideas, but, wanted to see if others had thoughts. This doesn't have to be a pop quiz. Feel free to bring ideas to the mailing list.

[Adam Montville]: This looks reasonable to me.

[Charles Schmidt]: in that case, treat this as a kick off to mailing list discussion. If people have other ideas for use cases let's get them out there.

[Charles Schmidt]: For the IM, what information, at a core level, is necessary for what we want? Clearly, we need to the software name, the version to whatever precision is necessary to make determinations, we will need a software publisher to disambiguate potentially overlapping names, and it sounds like the group wishes to have software location included. It sounds like the group has a desire to identify the source of the information itself. What else? Of course, these can go to the list too, but, if others have some ideas.

[Danny Haynes]: Would it be good to have a hash of the files?

[Charles Schmidt]: Yes, that is a very good question. It is actually two questions. There is the question of measuring the hash of discovered files and there is also using information to collect effectively golden measurements. In SWID tags, this would be the difference between payload and evidence fields. Payload is the golden measurements whereas evidence fields would be integrity measurements collected from the system about some set of files. Yes, that is certainly something to consider.

[Henk Birkholz]: I would strongly +1 to having hashes although they can't always be supplied as evidence or as payload. I would say it is vital.

[Charles Schmidt]: Any other thoughts or comments? Again, we should take this to the list to get more ideas.

[Ira McDonald]: Good list, but, there are probably two or three more. What is meant by software name? It has always struck me as very weakly defined in SWID tags.

[Charles Schmidt]: At some level, I think it means what we are given.

[Adam Montville]: Maybe human-readable marketing name?

[Charles Schmidt]: Most forms of evidence seem to have a name concept built into that, but, what they mean by it varies considerably.

[Danny Haynes]: Since we are talking about IM, we may need to leave it open to accommodate different data models.

[Ira McDonald]: Okay, so there is not really much precision to software name within publisher.

[Henk Birkholz]: A small addition to this. The IE in the current IM distinguishes software into operating system and application. Operating system names are much more complicated than application names because operating systems have updates and hotfixes where most applications are packages and more closely versioned. So maybe there is an attribute here to distinguish between operating system and application that is necessary? I am not sure.

[Charles Schmidt]: You do bring up good point that patches, especially those that do not change the version number of the software they update, which you could argue is bad practice, but, a practice. We need a way to identify those as well.

[Ira McDonald]: Yeah.

[Henk Birkholz]: Operating systems also have patches that do not change the operating system version. Also, operating systems are a very special software because it talks to hardware.

[Ira McDonald]: RFC2790 in the software install table, which is separate from the software running table, distinguishes between operating system, application, and driver. Why they distinguish between driver and the rest of the operating system because back then they didn't mean application-layer driver, they mean operating system driver. That has been morphed into many IETF and other models as a result because it has been ubiquitously implemented in every operating system bigger than a bread box.

[Charles Schmidt]: So, I think we are out of time. So, I just want to wrap up as far as next steps. We will continue these discussions on the list. I would like to get draft IM for IETF 96, but of course, that depends on the discussion.

[Jim Schaad]: Can you send me a draft of the NIST document with the SWID schema?

[Dan Romascanu]: You said IM. Do you mean a new one or data model?

[Danny Haynes]: I think he means put it in the existing IM.

[Charles Schmidt]: Yeah, we will work with existing work.

WG Way Forward

[Adam Montville]: Can we try to get the IM merger completed by July 8, 2016? Danny let me know if that is doable.

[Danny Haynes]: Yeah, the merger is almost complete. We have some changes out of this meeting and the rest of the work is around fixing the IPFIX syntax, and cleaning up Sections 5 through 7 and the appendices as I mentioned during the update. But, let's shoot for that date as I want to get this done.

[Adam Montville]: We still have to get the SACM Requirements to the IESG. I will work with Karen to get this done.

[Adam Montville]: I don't think there is any reason why we should be able to have an updated software identification draft by July 8, 2016.

[Adam Montville]: We should complete the WGLC for the Vulnerability Assessment Scenario I-D by IETF 96.

[Danny Haynes]: I thought we were going to hold off?

[Adam Montville]: It doesn't need to go to the IESG.

[Ira McDonald]: Do we really need to?

[Adam Montville]: Fair.

[Jim Schaad]: I would like to call it a last call. It means we do a hard review. It does not mean we need to advance the document.

[Ira McDonald]: Agree.

[Dan Romascanu]: Do we have to plan for IETF 96 in Berlin? Tomorrow, we should have the first agenda.

[Adam Montville]: We should be able to have our two meetings like normal.
[Adam Montville]: Thank you everyone for attending and participating in the meeting.