CTI-TC Weekly Working Sessions

Meeting Date: Thursday, June 9 2016
Time: UTC 14:00:00
Purpose: CTI-TC CybOX Subcommittee Working Session

Attendees:

Ivan Kirillov - Moderator  John-Mark Gurney  Christian Hunt
Trey Darley  Alex Tweed  Jon Baker
Ron Williams  Nicole Gong  Jane Ginn - Recorder
Rich Piazza  Richard Austin  Other Guests

Agenda:

- CybOX Relationships - metadata
- CybOX Actions versus Patterning
  - Do Actions need timestamps?
- We made a bunch of edits! Please review.
- Extension sub-classing - do we need it for MVP?
- Possibility of a call next Wednesday focused on Network Connection and User Account

Meeting Notes:

**Updates on Objects Under Development**

Ivan

*Discussed how to show Relationships as metadata in the CybOX Objects*
*Gave example on the screen*
- Gave another example of how to handle this for other Objects – eMail example
- Asked for comments – Noted discussion on Slack Channel

Trey

Talked about how this kind of data would be in other Objects besides those in
Said that he preferred the Flat option

John-Mark

Remember that one of the design principals is that Flat is better than Embedded

Ivan

That’s a good point
Showed the Base File Object – Currently have 10 Relationships defined

John-Mark

I like idea – let’s only add to where it makes sense

Ivan

Gave example of Container Relationship – Metadata not needed
Where we have Type – We can restrict use of Metadata

Ron

The extent to which JSON provides a sparse data model – What effect does that have?
Ivan
For the most part, things are optional

Ron
I was probably thinking out loud

Ivan
If you have Sibling properties – Can’t specify metadata
We’ll have to specify in the Spec

Ron
You can define the dependent relationships

Ivan
Trey, what should we do? Shall we go with the Flattened approach?

Trey
Let’s do that – Work up the normative text and put out for review – avoid vote

Ivan
OK, less anyone else has comments, we’ll continue
Our next topic – **CybOX Actions**
Big component in the CybOX Core Spec

Trey
One design goal is One Way of Doing Things – Major pain points in previous versions
Ivan and I had intense immersion in the problem space – certain things came to mind
By rolling up the Patterning type… we thought we might be able to eliminate the idea
Of a CybOX Action

Ivan
Gave example to illustrate a CybOX Action – Showed how Action construct worked
Gave example of how it could work by using the Patterning approach
The Use Cases for the 2 are different
Use Case for CybOX Action came about in the Sandbox Use Case
Patterning came from a different type of Use Case

Trey
*Asked a question about how a particular MAEC Use Case might work*

Ivan
It is those two… plus, it can be used for Analytics
Gave example – Knowing that can tell you differences about how certain malware work
Having the data can let a malware analyst perform analytics

John-Mark
Is this talking about the Patterning that Jason and I worked on?

Ivan
Yes

John-Mark
One of the things that I believe is that a CybOX Object should not contain a Pattern
*Gave example*

Ivan
Right a Pattern is more abstract

John-Mark
I think we need to have a guide of what expected Actions are – Not normative.
Just a Guide – If we don’t have something like that, we have a problem
With Interoperability
So, you just made the statement that CybOX should not include Patterns
Except under certain circumstances
Showed the ambiguity in the use of both types
My personal preference would be to enhance the CybOX language
We should clarify both

Because Pattern is a specific string – It cannot contain an Object Ref
They are very different things – Pattern is matching, whereas an Action is a reference to other Objects

Used a metaphor to represent the difference between Instance vs. Action
There is a knowledge life cycle, is that the right characterization?

Yes, I think the idea that John-Mark had to write a Guide is a good idea
Ok, let’s talk about Associated Objects
So my question is – Can we use the Meta Data approach here?
What we could do is set up a Dictionary.
Let’s put together a few more examples and show how it would work

Ivan – I give you kudos for putting together those examples
Guys – we need to get the Core Types right to reduce the risk of having a major release.
There were 2 or 3 things that we need to discuss.
One is the ArgleBargle – That is the “Container” concept
This is the point where STIX & CybOX interface
In STIX it is an Observation

Do we have anyone from the DFAX world on the line
Explained how creating the container with an ID was a compromise
It is to help people that use DFAX & MAEC
At the Container level – Ivan and I thought we should have a STIX ID
It should be a globally resolvable UUID
Gave example of a vendor product of pure CybOX coming off device
If there is no Timestamp on the Container

I would agree that it would make sense to have a Timestamp

I thought that one of the that one of the things that we discussed was that the tool would not be emitting a Timestamp.
The Object would have the Timestamp
Ron

*Asked a question about an observation that occurs at a point in time*

John

*Gave example of a DFAX Use Case* – Date created vs. Date analyzed

What does the Timestamp say?

Trey

The time that the Facts were written down

Ivan

That would be a Created Time

John

We already have that on the STIX side

Ron

It will be interesting to see how this works in products

John

*Gave example of how times could be different*

John-Mark

This will throw a wrench in the Patterning – Now there will be 2 dates

We need to make sure that this is clear in the normative text

Ivan

That will be easy to clarify

Right now we don’t define Timestamp precision

John

We don’t have on the STIX – not for Created Time

For Observation Time we do

Ivan

STIX is going to extend things… I’m not sure if it should be a String

John

I don’t think we need an ID

Trey

If you take that away it will undo the DFAX compromise from F2F

John-Mark

I support that we add an ID... not a STIX ID

One question I will raise – CybOX will have an additional Object Type

John

Will you call it ID or CybOX ID?

John-Mark

But will not be a TLO

John-Mark

The Container is related to an Observation

John

*Clarified the discussion we had at the F2F*

John-Mark

I’m trying to remember the discussions we had back then –

Ivan

I don’t really like the idea of copying the ID...

John-Mark

We should review the notes from the Baltimore F2F
Ivan: I wonder if it makes sense to have the STIX ID.

Trey: Everywhere we can make it consistent... then, it is easier for implementers.

Ivan: Do we really want the Container to be globally unique?

John: The idea of that takes us away from the use of CybOX in this way.

Trey: If we have an ID field, it should be required – Showed inherent conflict.

John: We can write normative language to clarify.

Ivan: It has no context – It has to be used with another framework. That is why GUIDs don’t make sense in CybOX.

John: Then the other languages decide how CybOX is used. Then it will be globally referenceable as we discussed at the F2F.

Trey: Gave example of being short-order cooks.

John: Another option would be for you to create and the other languages can Reference (or not).

*Trey & John discussed how the ID should be used*

Ron: To earlier point, if the CybOX identifier is the same as STIX, what would be The implication?

Trey: Asked Trey to show example – We would copy the normative text. How many people who are using STIX are not going to use CybOX? Unless there is a clear reason, the normative text should be the same.

John: I don’t think the CybOX ID should be a UUID.

Trey: We have a couple of other points:
- Sub-classing – Send to the list
- We should talk about scheduling
- John since we are in Seoul – we won’t be able to make deadline.

Ivan: Can we reschedule the next call to Wednesday? Trey cannot make.

John-Mark: Yes, I can make that one.

Jane: I won’t be able to make that one.

Ivan: There are some really good things we really need to tackle.
I sent an email out to the list for comments
- Size fields
- Negative integers
Please review these things
If no comment, we’ll push these out to Draft
We’ll send another list of Objects out for review later today
With that, we’ll talk to you all later

Trey

Thanks all for your time!

************************************************************************

Meeting Terminated