Meeting Date: Thursday, April 7 2016
Time: UTC 14:00:00
Purpose: CTI-TC CybOX Subcommittee Working Session

Attendees:

Ivan Kirillov - Moderator          John Anderson          Gary Katz
Trey Darley                        Brad Butts             John Wunder
John Wunder                        Ali Khan               Laurie Thomson
Paul Patrick                       Jeffrey Mates          Will Urbanski
Rich Piazza                        Michael Pepin          Mark Davidson
Bret Jordan                        Dean Thompson          Jane Ginn - Recorder
                                     John Wunder
                                     Laurie Thomson
                                     Will Urbanski
                                     Mark Davidson
                                     Jane Ginn - Recorder
                                     Other Guests

Agenda:

- Update on State-of-the-Art
- Approaches to Resolve Outstanding Issues
- Go over the Feature Roadmap

Meeting Notes:

State-of-the-Art of CybOX Development

Ivan Kirillov
Rework of CybOX relative to STIX – Tear line with STIX
Question about whether we have consensus that CybOX does not just serve STIX

Trey Darley
Explained that there was general consensus that CybOX does not serve exclusively STIX
But, that said, CybOX probably has a higher obligation to STIX
Does CybOX serve STIX or other use cases?

Discussion of Issuing Ballots to Resolve Outstanding Issues

Ivan Kirillov
Suggest we consider a ballot that CybOX’s primary responsibility is to STIX
Ability to create/submit new CybOX Objects
Should we vote on use cases? Objects vs. Observables? Actions?

Ivan Kirillov
Request to have a ballot that suggests that the primary obligation of CybOX is to STIX

Gary Katz
It's fairly vague.
Ballot might be one sided, because the CTI TC is mostly STIX users
Might be better to identify what the breaking point is.
Would be good to design a process by which people can suggest objects
More specific ballots are better: is observation in, is action in, etc.

Ivan Kirillov
The question is also can CybOX be used on its own
-- Or is it always used by down steam standards?

Gary Katz
Ballot didn’t work
Needs to be a way for people outside of OASIS to contribute

Bret Jordan
We have to follow OASIS rules

Ivan Kirillov
Is it a stand-alone-language?
Or is it a set of Types that are used downline

Bret Jordan
I’ve always thought it should be pulled into a higher-level language

Trey Darley
We are trying to make it easier for people to contribute
Make your own Object-Kit

Gary Katz
Just want to have clarity in Balloting

Trey Darley
Can have illogical outcomes with some Ballots
Gary, can you lead a small working group to describe the tearlines

Gary Katz
I don’t mind taking partial responsibility

Bret Jordan
I’m all for CybOX being its own thing

Ivan Kirillov
Want to make the ballot clear
Use Cases Question
Stand Alone vs. Object-Level Language

Jeff Mates (CHAT PANEL)
My major concern is that both STIX and CybOX should be easily portable between each other.
Need common definition that all of the lowest level object types either implement or extend upon.
For example a CybOX and STIX relationship should be the same
Core structure for a FileObject should interact with Indicator should allow a relationship
Connect the two in a consistent manner.

Review of Feature Map

Ivan Kirillov
Considering doing a vote like they are doing in STIX on the MVP Matrix?

Trey Darley
Because there are votes going on in STIX

John Wunder
Can you give us an example?

Trey Darley
Field-level markings – Will you have in STIX – Can we borrow them

Ivan Kirillov
Discussion about Relationships and what they mean
In past – relationships defined between objects
-- Objects and Processes
Sometimes necessary... but may be embedded

John Wunder (CHAT PANEL)
On the Relationship question, there’s obviously overlap
STIX has a lot of features in TLOs that we would have to inherit into CybOX
My thought was that CybOX defines a minimal relationship model, STIX just extends it

Dean Thompson (CHAT PANEL)
Relationships are important, they are the ability to convey the true story

Jeff Mates
With Embedding - That is when the process really starts breaking down
It is much easier if we simply say no embedding

Dean Thompson (CHAT PANEL)
It is better to say that this file was created, it spawned a process, did the following registry key changes and did a process injection.
Some objects lend themselves more to relationships than others.

Ivan Kirillov
Can you send me that Use Case?
If you look at the Network Connection Objects

Jeff Mates
I’ve worked on that one too...
 Does make it less human-readable... but for machine-readable makes it faster

Ivan Kirillov
Is this something we need to do in the spec, or depend on tools?

John Wunder
It is a spectrum – we need to find the point on the spectrum

Jeff Mates
Gave example of Network Connection – Number of possible relationships

Ivan Kirillov
I guess that makes a lot of sense... Too much embedding, too many
We may have to decide on a case-by-case basis

Trey Darley
We are getting hung up a lot on the Network Connection Object
This is most complex by far
Maybe we should put together some drafts...
What it looks like as embedded vs. relationships
Define 5 or 6 examples
Then write JSON for each and run tests

John Wunder (CHAT PANEL)
+1 to sample JSON, plus parse code

Ivan Kirillov
We cannot take that approach for each object

Jeff Mates
We had actually looked at that before Owen sent out
If you put it as a string internally, build this... fill in as a string – then duplicating where you are storing
As opposed to only one way of doing things

Ivan Kirillov
That was a problem with STIX 1.x because there were two way of doing things

Trey Darley
We need an architectural plan

Jane Ginn
Terry McDonald presented a set of Design Principals – Maybe we should look at those again

Gray Katz
What about Extensions? Still on Roadmap?

Ivan Kirillov
Yes, it is
Trey Darley
Gave example of how different Extensions are mutually exclusive and not exclusive

Gary Katz
Happy to help – Want to volunteer Jeff

Jeff Mates
I adore Extensions. Using them makes everything easier to work with

Ivan Kirillov
Patterning is something we don’t have consensus on

Trey Darley
[LAUGHTER] Understatement of the Year

Ivan Kirillov
Still tied to Object Model in STIX… not as useful as before
Still a dependency there – Still some direct linkages from the Patterning Expression Language
Need to figure out whether anyone needs patterning beyond the CybOX level?
Is patterning MVP?
Documentation needs to be improved
Basic patterning vs. massively complex patterning
Need to get a work product out from the patterning working group

Trey Darley
All I’m seeing now is the use of Observable Composition
If people had option. Everyone is doing that at the CybOX level… Is there a lesson learned

Jeff Mates
Have seen… but, poor documentation side

Gary Katz
I would agree on the poor documentation

John Wunder
For what it is worth… puts it all to the patterning structure. All gone into a single patterning place

Trey Darley
First I’ve heard of that...

John Wunder
I believe it is in the Pre-Draft Spec

Bret Jordan (CHAT PANEL)
https://docs.google.com/document/d/1F1c05GgYaJFV1Z04B8c_T3vEE-LRQTPExF24LvQQAsk/edit#heading=h.8gcmzxS0n1ri

John Wunder
Timestamp example – What should be in STIX vs. matching in a database – John-Mark Example?

Ivan Kirillov
Does not have to be tied – Just an operator in the language
This is something that the Patterning Sub-group has to define

John Wunder
Noticed that it is marked TBD – Is it because you thought it should be in STIX

Ivan Kirillov
Yes

Trey Darley
I don’t feel comfortable pulling Patterning out of STIX… Gave example of how it could be confused

Ivan Kirillov
That would be my understanding of the question
If Patterning goes away… what would we use instead? Snort

Trey Darley
If we do the Patterning in the MVP
Jeff Mates (CHAT PANEL)
If we are bringing up the proposal for common again could we try to put patterning in common?

John Wunder (CHAT PANEL)
Wouldn’t that mean patterning would depend on CybOX?
CTI Common would depend on CybOX
Since patterning is tied directly to CybOX

Jeff Mates (CHAT PANEL)
Not really because if common has relationship and a few core object structures in it your patterning language might point to: core object with property X without knowing if a given property group is defined property group = extension sorry

John Wunder (CHAT PANEL)
Hmm, that’s not exactly how they’re defining it now I don’t think some of the operators are cyber-specific

Jeff Mates (CHAT PANEL)
True, I feel like there ought to be a function based workaround for that but not sure or if it would end up being too ugly even if it did work correctly

Ivan Kirillov
There were a few outstanding questions

Gary Katz
Should we add this to the list of Ballots?

Ivan Kirillov
[Chuckle] Yes... overload

Bret Jordan
We all agree we need Patterning...The question is How Much?

Trey Darley
If we agree with Jason Keirstead’s perspective... then it would be a very advanced type of patterning
How do we define for the short term vs. the long term?

Ivan Kirillov
We need to get some output from the Patterning Workgroup

Gary Katz
I think that makes sense

Trey Darley
Now we only have a few minutes
Do you all think we should do this again?

Gary Katz
I won’t be able to make it next week, but I think it is useful

Ivan Kirillov
Ok, we’ll have it next week – Not sure how many more but we will decide as we go

Trey Darley
What are the Action Items?

**Action Items**
- List of Ballots to Gary
- Relationship Object Discussions
- Need Resolution on Patterning

**Meeting Terminated**