Overview of STIX 2.0 Round1 Strawman Proposals

January 14, 2016

What are the STIX 2.0 Round 1 Strawman Proposals?

- Set of independent yet integrated concrete proposals for addressing numerous high-priority issues

- Contributions as experts not co-chair
  - Sean Barnum, Paul Patrick, Cory Casanave, Pat Maroney, Jerome Athias, Shawn Riley and input from a few others.

- The STIX 2.0 Round 1 Strawman proposals include 18 proposals covering >24 tracker issues including:
  - 8 Top Ten Roadmap issues (#306, #148, #291, #221, #201, #360, #231, #380)
  - >16 other issues

Motivation behind proposals?

- Move us forward quickly but correctly
  - Demonstrate these are not mutually exclusive
- Identify and close quick consensus issues
- Identify good issues for F2F agenda
- Provide well thought out fully modeled proposed solutions to issues to focus and frame conversation and move us forward faster
- Yield concrete model output for spec rather than just talk about the issues

Design Approach

- Is NOT pitching a “strawman”
- Is suggesting a set of individual proposals on specific issues
- Good to show that multiple proposals can work together but traceability is very important to ensure we know what the strawman actually does and does not provide

- Principles:
  - Recognize the value of the collaboration that has occurred over the last 4+ yrs
  - Recognize lots of issues have been identified to make things better
  - Focus on each issue and understand its factors (as-is, to-be, impact)
  - Consider local issue factors but also its effect on other issues/areas (modeling is key)
  - Integrate solutions into “strawman” and review previous issues/decisions
  - Be mindful of real-world use cases
  - Ensure practicality of JSON MTI serialization implementation
  - Maintain ability to support other potential serializations

Design Targets

- Simplify everywhere possible without losing key capabilities
- ”One way of doing things” unless absolutely justified
- Consistency
  - Across STIX and betweek STIX and CybOX
- Modularity

Are the two “strawmen” diametrically opposed?

- No

1. Mostly aligned on intent and resulting solutions
2. Differences on some specific sub-issues
3. Each may cover additional issues the other does not
   - These will require further discussion post-F2F

- F2F agenda strongly influenced by #1 & #2

Common results

- Consistent object derivation
- Significant reduction of object embedding
- Relationships as independent structures
- Abstraction of common high-level concepts (Sighting, Identity, Source, Victim, etc.) to separate objects
- Clarified semantics/refactoring of TTP and ExploitTarget
- Simplifying structures like controlled vocabularies and data markings
- Significantly reduced optionality
- Flattening unnecessary complexity
- Remove redundancy (e.g. Indicator_Expression)
- CTI Common
- Etc.

Issue coverage

- Extend core constructs from a single base class (#148)(RM #3)
- Make IDs required (#221)(RM #5)
- Add Alternative_IDs to all top level objects (#358, #187)
- Remove Short_Description (#194)
- Abstract Source to top level construct rather than embedded only within other constructs (#233)
- Remove the @id/@idref attribute from some constructs (#336)
- Make Observable structure align with other components (#160)
- Remove either embedded or referenced relationships (#201)(Rm #7)
- Abstract relationships as top level constructs rather than embedded within other constructs (#291)(RM #4)

Issue coverage continued

- Make field names consistent for usages of Information Source (#263)
- Abstract Sightings into an independent construct rather than embedded within Indicator (#306)(RM #1)
- Clarify semantics of different types of TTPs as expressed in the TTP construct (#360)(Rm #10)
- Refactor Kill Chain Types (#117, #191, #241, #190, #47)
- Flatten list layers in Package (#382)
- Remove abstract base types for “top level” objects (#386)
- Refactor Report object (#385)
- Clarify semantics of different types of Exploit Targets as expressed in the Exploit Target construct (#387)
- Abstract Victim to top level construct rather than embedded only within Incident and TTP (#149)

Where are they?

- Overview page and proposals are in the STIXProject/specifications wiki

- Full STIX 2.0 Round 1 Strawman draft UML model is available in a “MagicDraw Model” folder in the stix-2.0-Round-1-Strawman branch of the specifications repository on github. Diagrams folder contains diagrams if you cannot view the model directly.

Proposal format

- **Issue Summary** (prose description of the issue in question)
- **Proposed** (prose description of proposed changes)
- **Proposed Model** (diagrams from strawman model demonstrating proposed changes)
- **Examples**
  - **JSON Schema Serialization snippets** (hope to collaborate with TWIGS team to integrate)
  - **JSON Serialization example snippets**
- **Open Questions**

Results

- Dramatically improved consistency and simplification
  - > 100 types and properties removed
- Optionality significantly reduced
  - “One way to do things” applied universally with only a single exception (data markings references embedded)
- Now very much graph-aligned
  - Improved analytic support
- High assurance that changes are explicit, traceable and understandable
- Integrated model baseline to support STIX 2.0 spec evolution

Intersection with TWIGS

- Full alignment on 9 of the proposals
- Partial alignment (often close) on 9 of the proposals

- Hope to bring both perspectives (model-centric and serialization-centric) together to yield clear treatments of the issues and baseline content for language spec and MTI serialization spec

F2F STIX agenda

- STIX Block #2 (10:40-11:30am 1/14/16) will focus on the areas of alignment and apparent consensus

- STIX Blocks #3 (2:40-4:15am 1/14/16) & #4 (12:30-2:15am 1/15/16) will focus on areas of near-consensus from “strawmen” and list discussions
  - Relationships
  - Source reference for each construct embedded or via relationship
  - Define consistent Timestamp format
  - Sightings
  - ID_Format
  - Simplified CV structure

- **Backup**

Notes and Caveats

- All proposed changes exist in an integrated STIX 2.0 Round 1 Strawman model derived from 1.2.1 model
- Proposals, model fragments and JSON snippets will often include changes from other issues and proposals
- JSON snippets are illustrative not normative
  - Don’t get hung up on style or naming conventions (we can adjust that)
  - Snippets were kept as simple and focused as possible
  - There are likely errors here and there
- It is our intent to provide JSON Schema snippets but have not yet had time. Assistance would be appreciated.
- We’ll tackle these proposals on the list but please feel free to register thoughts and feedback within the relevant issues in the tracker