**README.md**

This readme will explain usage of our scripts to detect missing relationships from the stix-elevator tool.

**Structure:**

* main.py and rel\_utils.py

**Usage:**

* python main.py  
  The *rel\_utils.py* file contains the functions used in the main script. These can be modified based on the user's interest.

**Input:**

* STIX 1.x (xml) and STIX 2.x (json) files and specified them in the main.py code.
* Relationship.csv: All supported relationship types are in “relationship.csv”. More types can be added in the future if necessary.

**Output:**

* Missing\_objects.txt: missing objects in the json file
* ref\_diff\_list.txt: the IDs and IDrefs of the missing relationships
* missing\_relationships.txt: missing relationships in the json file
* allInfoJson.txt: missing objects and relationships in the json file
* undefined\_relationship.txt: all the ID and Idrefs whose object mappings do not in the stix relationship.csv
* source\_objects.txt: source objects
* target\_objects.txt: target objects

The missing objects and relationships need to be manually inserted into the .json file.

**Code description:**

The key steps of the main.py file are

1. Finds all the ID and IDref of the original xml file and their objects in pair
2. Finds all the source\_ref and target\_ref of the converted json file in pair
3. Determines the ID and IDrefs that are in the xml file but not in the json file in pair.
4. Loads the relationship type “relationship.csv”.

The key function create\_*relationship* creates the missing relationships and outputs the undefined relationships (i.e., their relationship type between two objects is not defined in “relationship.csv”). Each module has comments that detail inputs, purpose and outputs.

Printing “Target\_types not in Objects” includes the target\_type that is not in the object dictionary. One can manually add those missing properties to the object dictionary and the csv file if they are available.

The current object table and the csv are built based on <https://stix2-elevator.readthedocs.io/en/latest/stix-mappings.html#relationships>.

####06/04/2022#####

Updated Object creation

For each type of objects, only its ID and type are real, and other information is made up as follows:

Indicator: pattern = ‘ipvr4:000’,

pattern\_type = ‘stix2.1’

Malware: is\_family = False

Campaign: name = ‘nameCampaign’

AttackPattern: name = ‘nameAttack’

Infrastructure: name = ‘nameIns’

ThreatActor: name = ‘nameThreat’

ObservedData: first\_observed='2022-06-03T17:12:55.594Z',

last\_observed='2022-06-04T17:12:55.594Z',

object\_refs = 'markings--027a4291-d4a9-490b-9107-800abaea4c8c',

number\_observed = 1,

allow\_custom=True

Requirements:  
1). stix  
2). stix2  
3). numpy  
4). nested\_lookup

5). Pandas

Installation:

1). pip install stix

2). pip install stix2

3). pip install numpy(usually comes already installed)

4). pip install nested\_lookup

**Fixing JSON files for Python-stix2 to read**

Often Python-stix2 may have issues reading the JSON files and some of the common problems and the solution to fix them are listed here:

1. InvalidValueError: Invalid value for WindowsPEBinaryExt 'time\_date\_stamp': must be a datetime object, date object, or timestamp string in a recognizable format.

A fix can be for

"time\_date\_stamp": "2018-03-28 07:14:10-04:00"

Is to change it to:

"time\_date\_stamp": "2018-03-28T07:14:10.000Z"

1. InvalidValueError: Invalid value for File 'extensions': No values for required properties for WindowsPEBinaryExt: (pe\_type).

A fix for

**"windows-pebinary-ext": {**

**…**

**…**

**"size\_of\_optional\_header": "4096",**

**"time\_date\_stamp": "2018-03-28T07:14:10.000Z"**

**}**

Is to add “pe\_type”: “Generic”, i.e.,

**"windows-pebinary-ext": {**

**…**

**…**

**"size\_of\_optional\_header": "4096",**

**“pe\_type”: “Generic”,**

**"time\_date\_stamp": "2018-03-28T07:14:10.000Z"**

**}**

1. InvalidValueError: Invalid value for EmailAddress 'belongs\_to\_ref': not a valid STIX identifier, must match <object-type>--<UUID>: NCCIC:WinExecutableFile-799d8919-9e90-487c-819b-e7303d95d982

This be because NCCIC:WinExecutableFile-799d8919-9e90-487c-819b-e7303d95d982 does not follow convention object-type-UUID[[1]](#footnote-1).

E.g., an easy fix is to remove the following from the JSON file:

**{**

**"belongs\_to\_ref": "NCCIC:WinExecutableFile-799d8919-9e90-487c-819b-e7303d95d982",**

**"id": "email-addr--98e8f5e8-9372-5cd0-9179-487a296e1d26",**

**"type": "email-addr",**

**"value": "konedieyp[@]airmail.cc"**

**},**

1. AtLeastOnePropertyError: At least one of the (dst\_ref, src\_ref) properties for NetworkTraffic must be populated (dearcry.json, hoplight\_fixed).

E.g.,

**{**

**"dst\_port": 88,**

**"id": "network-traffic--df70d186-889d-5c19-a548-d3c2554dbceb",**

**"protocols": [**

**"tcp"**

**],**

**"type": "network-traffic"**

**},**

Can be changed to:

**{**

**"dst\_port": 88,**

**"id": "network-traffic--df70d186-889d-5c19-a548-d3c2554dbceb",**

**"src\_ref": "ipv4-addr--a98d10a5-d98b-52ca-b54f-31069ab2880e",**

**"dst\_ref": "ipv4-addr--a98d10a5-d98b-52ca-b54f-31069ab2880e",**

**"protocols": [**

**"tcp"**

**],**

**"type": "network-traffic"**

**},**

1. InvalidValueError: Invalid value for ObservedData 'objects': The (payload\_bin, url) properties for Artifact are mutually exclusive (copperhedge.json, electricfish\_fixed.json, fastcash\_fixed.json, pebbledash\_fixed.json, sorefang\_fixed.json)
2. File "C:\Users\yuemeng\Anaconda3\lib\json\decoder.py", line 355, in raw\_decode

raise JSONDecodeError("Expecting value", s, err.value) from None

JSONDecodeError: Expecting value (blindingcan\_fixed.json, sunburst\_fixed.json)

1. InvalidValueError: Invalid value for Bundle 'objects': The (object\_refs, objects) properties for ObservedData are mutually exclusive. (hoplight\_fixed.json)
2. InvalidValueError: Invalid value for Bundle 'objects': No values for required properties for NetworkTraffic: (protocols). (slothfulmedia\_fixed.json)
3. InvalidValueError: Invalid value for ObservedData 'objects': At least one of the (dst\_ref, src\_ref) properties for NetworkTraffic must be populated. (sunshuttle\_fixed.json)

Fixed files:

Dearcry\_fixed.json, eccentricbandwagon\_fixed.json, artfulpie\_fixed.json, federalAgency\_fixed.json, solarWinds\_fixed

1. If we change it “NCCIC-WinExecutableFile--799d8919-9e90-487c-819b-e7303d95d982” then Python-stix2 still does not recognize it. [↑](#footnote-ref-1)