Annex B. Tools

TL;DR Tools

- OSINT tools are useful for disinformation analysis
- Work out where and how to store your datasets

Disinformation Tools

The tools you need depend on the size of the response you're planning to run, the number of people involved, and things like whether they already have access to their own specialised tools for things like tracking disinformation narratives. A good basic set of tools will include:

- Ticket tracking. Tickets help you keep track of incidents, including actions taken on them and where to find more information.
- Analysis. Extract information from artifacts and other social media data.
- Data storage. Somewhere to store and access large and diverse datasets.
- Incident notes. Share notes and incident summaries whilst a team is working on them.
- Alert sharing. Share incident information quickly with other teams.

Response Tracking

If you have a large team with people joining and leaving responses, response management becomes essential. A ticket tracking app helps a lot, but we've also run incident responses armed with nothing more than a Slack group and a Google folder.

Team Communications

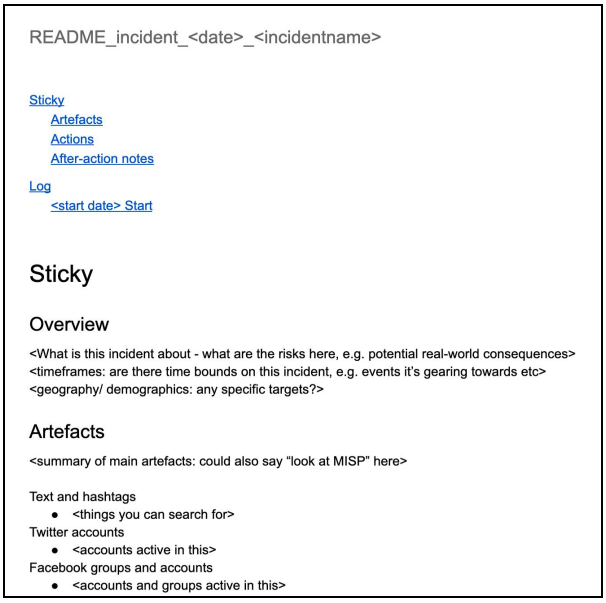
Use whatever works for your team. We tend to use Slack.

Ticket Tracking

Incident tracking tools range from a shared spreadsheet (e.g. googlesheets and airtables) to ticketing systems (The League uses D3PO), and case management systems like TheHive.

The CTI League disinfo team tried using Hive to manage its list of incidents, and links from them to the other objects and data connected to incident responses. Check Hive and search for the incident name. All incidents will have the tag “disinformation” and word “Incident” in the title, which should help with searching.

Tracking with a Googledrive


README_incident_<date>_<incidentname>

Sticky
Artefacts
Actions
After-action notes

Log
<start date> Start

Sticky

Overview

<What is this incident about - what are the risks here, e.g. potential real-world consequences>
<timeframes: are there time bounds on this incident, e.g. events it's gearing towards etc>
<geography/ demographics: any specific targets?>

Artefacts

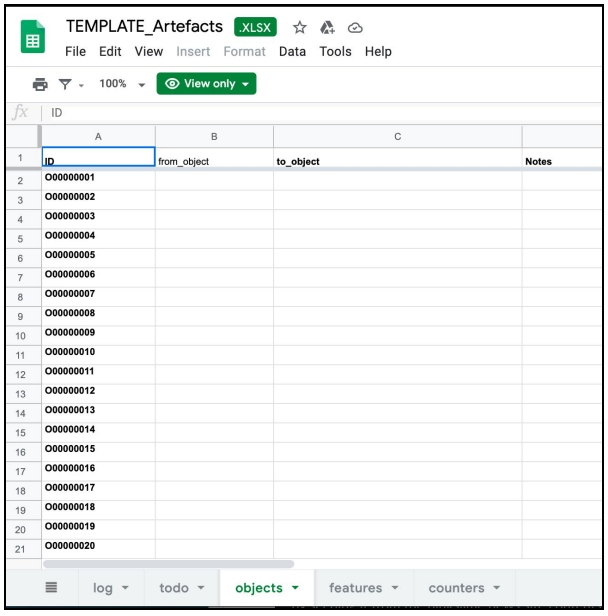
<summary of main artefacts: could also say “look at MISP” here>

Text and hashtags
- <things you can search for>

Twitter accounts
- <accounts active in this>

Facebook groups and accounts
- <accounts and groups active in this>

Googledoc-based reporting



Googlesheet-based reporting

With this variant, you still need to track which incidents you're responding to. We used a googlesheet with a row for each incident, giving it a name, a status (live, watching, closed), a start date, an end date (when we closed the response, not when the incident closed), and links to any slack channels and googlefolders we used to collect and store artefacts, and write up reports in.

One iteration of these processes used a shared googledrive for each incident, with a folder for images and other artefacts that couldn't be easily copied into a document (videos etc). Within the googledrive was a README file (template above), with sections for artifacts found (because pasting the same image repeatedly is less helpful than giving it a reference number), actions taken, a log of what was found and done on each day of the incident response, and after-action notes when the incident is closed. This is loosely based on some of the shared documents used during crisismapping.

One flaw of the document-per-incident approach was that it became difficult to share and check incident artifacts. We solved this by creating a googlesheet template where we could add artifact URLs, then run code to upload them into other systems for automated analysis.

Analysis

Social media analysis: At a minimum, you'll need network and text analysis tools. Some of our teams bring their own; otherwise sourcing or creating open-source analysis tools is a good thing.

Artifact analysis: we're often starting investigations from single artefacts: text, images, video, domains, groups. We borrow heavily from OSINT toolkits to analyse each of these.

Analysis Tools

Some basic tools:

- Data gathering:
    - Reaper https://github.com/ScriptSmith/reaper https://github.com/ScriptSmith/socialreaper https://reaper.social/ - scrapes Facebook, Twitter, Reddit, Youtube, Pinterest, Tumblr APIs
- Network analysis and visualisation: there are many tools for this.
    - Gephi is a good standalone tool https://gephi.org/users/install/
    - Networkx is a useful python library
- URL analysis
    - Builtwith.com
- Image analysis
    - Reverse image search: tineye.com, Bellingcat guide
    - Image search: bing.com, yandex.com
    - Image text extraction: bing.com, yandex.com

Disinformation-specific tools:

- Indiana University has a set of tools at https://osome.iuni.iu.edu/tools/
    - Botometer: check bot score for a twitter account and friends https://botometer.iuni.iu.edu/#!/
    - Hoaxy: check rumour spread (uses Gephi) https://botometer.iuni.iu.edu/#!/
    - Botslayer https://osome.iuni.iu.edu/tools/botslayer/
- Bellingcat made a list of useful tools

Bellingcat's really big tools list - worth reading if you need a specific OSINT tool

Gephi

Gephi is useful for viewing and analysing networks. This is a manual process for creating twitter network diagrams, with instructions created from Andy Patel's video:

- Get Gephi from https://gephi.org/users/download/ - install it.
- Start Gephi.
- Click on the top menu, then file, then "import spreadsheet". Grab User_user_graph.csv - use all defaults
- Top menu: Go to data laboratory, "copy data to another column", click 'id', click okay.
- Go to overview. RHS: Run modularity algorithm, using defaults
- RHS: Run average weighted degree algorithm
- LHS: Click color icon, then partition, modularity class. Open palette, generate, unclick "limit number of colors", preset=intense, generate, okay
- LHS: Select "tt", ranking, weighted degree, set minsize=0.2, choose 3rd spline, apply
- LHS: Layout: OpenOrd, run. Then forceatlas2, run. Try stronger gravity, and scaling=200
- Top menu: Preview - select "black background", click "refresh". Click "Reset zoom"

Gephi has an API - these tasks could be automated.

Python

You'll probably use Python scripts and Jupyter notebooks. Useful resources include:

- Python distributions and libraries
    - https://www.anaconda.com/distribution/ Anaconda. Comes with Python, Jupyter notebooks and many useful libraries including Pandas and BeautifulSoup

- Written material and videos:
    - The Best Way to Learn Python Good for non-programmers.
    - Learn Python the Hard Way Comprehensive learning
    - The Python Tutorial Good for seasoned coders
    - NewBoston's Python videos
    - Online book: Think Python
- Courses:
    - http://www.pyschools.com/ A course that grades you as you go.
    - Codecadamy's Python course. Type your code into a box and test it online.
    - MIT's Intro to Computer Science. Free online MIT course
    - Python's list of python tutorials
    - 5 Best Websites to Learn Python
- Getting help:
    - When you start python, type help(xx) to get information about what you can do with the variable called xx.
    - If you get stuck, Stack Overflow probably has answers to your problem.

Data Storage

We've tried DKAN storage for json, CSV and image files, with sql for other objects of interest, and are investigating other storage methods.

- Data storage / Threat Intelligence tools
    - https://getdkan.org/ DKAN is a data warehouse tool - it's where we store large datasets and their descriptions,for analysts to use.
    - MISP https://www.misp-project.org/

Incident Notes

Shared notebooks (e.g. Googledoc templates) work for this, and some tracking systems (e.g. TheHive) also include shared notes.

Incident technique, artefact and narrative sharing. Techniques, artefacts and narratives are objects of specific importance to an incident: they're the objects that you want to share with responders, like hashtags, groups, and superspreader account ids. Each incident is built on techniques, artefacts and narratives: collecting, annotating, and sharing these is an important part of the teams' work. We've tried a range of tools, from shared spreadsheets (googlesheet templates) to MISP and DKAN for this.

Alert Sharing

One group can only do so much on its own. Most of our communications to date have been through individual connections and the cross-team tracking system inside the League, but MISP allows for both setting an event to public share, and for emailing event summaries out to a subscriber list. Other possibilities include a public list of non-sensitive incidents, an incidents mailing list etc.

For incident sharing, we've worked with the MITRE ATT&CK toolset, MISP, and OpenCTI.

The important thing is that the data we share tells a story. The AMITT framework summarises behaviour; the fancy tags (DFRlab dichotomies etc) help describe the event and provide context on what we're seeing, and MISP objects help us represent the relationships between things: Who posted a blog post, who was mentioned in a news articles, who is the registered owner of a domain etc.

Ultimately these are the things we're aiming to build and share. Not a flat list of indicators, but a model of how the adversary operated.

AMITT STIX

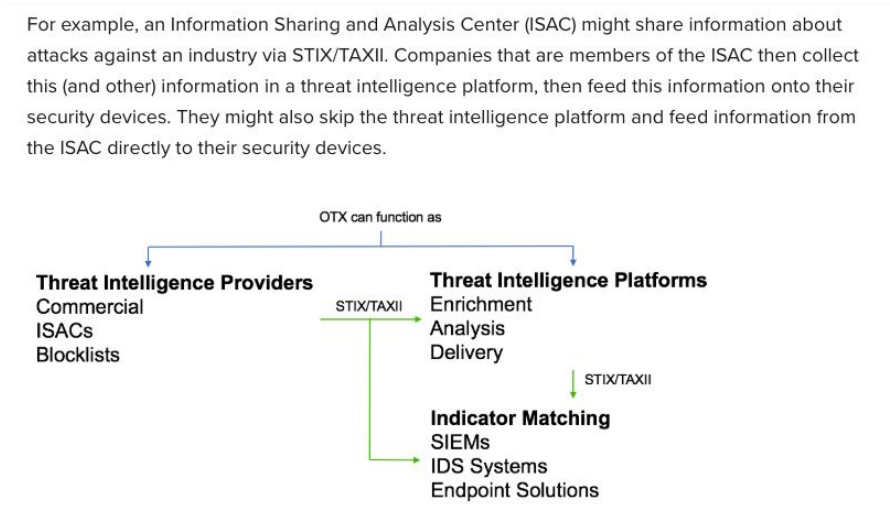


Image from https://stixproject.github.io/about/STIX_Whitepaper_v1.1.pdf

AMITT is now available as a STIX 2.0 bundle; STIX 2.1 will include an incident object.

MISP with AMITT

To use AMITT, list and share the components you see in your incident: AMITT is now built into the MISP tool, making this easy to do. Compiling and reporting incidents is an important aspect of both responding and developing the tools needed to do so. To be effective, those reports should include as much information as possible about the stages and techniques at play in those incidents.

MITRE ATT&CK 2.0 is in the works and it refactors high level capabilities into implementations. It's a direction we'd like to see with AM!TT and something our group will continue to work toward.