forensic sciences

MDPI

Technical Note

Instrumenting OpenCTI with a Capability for Attack Attribution Support

Sami Ruohonen [1], Alexey Kirichenko [2], Dmitriy Komashinskiy [1,*] and Mariam Pogosova [1]

[1] WithSecure Corporation, Tammasaarenkatu 7, 00180 Helsinki, Finland; sami.ruohonen@withsecure.com (S.R.); mariam.pogosova@withsecure.com (M.P.)
[2] Faculty of Information Technology, University of Jyväskylä, Seminaarinkatu 15, 40014 Jyväskylä, Finland; alexey.l.kirichenko@jyu.fi
* Correspondence: dmitriy.komashinskiy@withsecure.com

Abstract: In addition to identifying and prosecuting cyber attackers, attack attribution activities can provide valuable information for guiding defenders' security procedures and supporting incident response and remediation. However, the technical analysis involved in cyberattack attribution requires skills, experience, access to up-to-date Cyber Threat Intelligence, and significant investigator effort. Attribution results are not always reliable, and skillful attackers often work hard to hide or remove the traces of their operations and to mislead or confuse investigators. In this article, we translate the technical attack attribution problem to the supervised machine learning domain and present a tool designed to support technical attack attribution, implemented as a machine learning model extending the OpenCTI platform. We also discuss the tool's performance in the investigation of recent cyberattacks, which shows its potential in increasing the effectiveness and efficiency of attribution operations.

Keywords: cyberattack; technical cyberattack attribution; digital forensics; machine learning; cyber threat intelligence

Citation: Ruohonen, S.; Kirichenko, A.; Komashinskiy, D.; Pogosova, M. Instrumenting OpenCTI with a Capability for Attack Attribution Support. *Forensic Sci.* **2024**, *4*, 12–23. https://doi.org/10.3390/forensicsci4010002

Academic Editor: Mary Aiken

Received: 23 August 2023
Revised: 21 December 2023
Accepted: 17 January 2024
Published: 23 January 2024

1. Introduction

Law Enforcement Agencies (LEAs), forensic institutes, National Cyber Security Centers (NCSCs), Computer Emergency Response Teams (CERTs), and companies providing cybersecurity services routinely have to investigate cyberattacks on organizations and citizens. In many cases, a key question in such investigations is who is responsible for conducting a given cyberattack (which is also a key question to answer for achieving accountability in cyberspace [1]). This identification of the source of a cyberattack—which can be a nation state, a crime syndicate, or other nefarious group or individual—is often referred to as "cyberattack attribution" and involves technical, legal, and political analysis [2]. In this article, the focus is on technical attack attribution, which is based on the analysis of technical attack traces and Cyber Threat Intelligence (CTI). Technical attribution activities rarely result in the names or locations of the people behind a cyberattack, instead providing threat actor monikers [3] (e.g., "APT 41" or "Black Basta") and similar information. This is one reason why it was pointed out in [4] that "... questions of responsibility are rarely decided solely through a single technological tool or form of evidence ..." [4] (p. 382) and "... a legal approach, rather than a technological one, can solve the attribution problem" [4] (p. 376). However, technical attribution is almost always the primary element of any attribution efforts, providing key facts and hypotheses.

Identifying the threat actor behind a cyberattack can be very important and valuable, though the attribution value and investigation priorities vary and depend significantly on the context. For internal cybersecurity teams, CERTs and commercial service providers, technical attribution efforts usually help understand the attacker's intentions, capabilities

and level of sophistication, modus operandi, and expected behavior, informing the defenders' security procedures from prevention to response and remediation and giving them greater confidence in their operations. For example, the understanding of an attacker's Tactics, Techniques, and Procedures (TTPs) guides the defenders in what additional attack traces and artefacts they should look for and what vulnerability patches they should prioritize in order to minimize the impact of an ongoing attack and the risk of future ones. In the context of cyberattacks driven by political, military, or industrial competition motives, the technical attribution value can include a reliable view of the impact of sensitive information loss and can even extend to driving foreign policy measures, if threat actors are associated with specific nation states. Also, importantly for LEAs, the insights provided by technical attribution efforts are almost always the first step in identifying and prosecuting attackers.

With all the potential value, technical analysis involved in cyberattack attribution requires a high skill level, experience, access to up-to-date CTI, and significant investigator effort. Furthermore, attribution results always contain elements of uncertainty, and skillful attackers often work hard to hide or remove the traces of their operations and to mislead or confuse investigators. Recognizing the challenges, the EU-funded CC-DRIVER [5] and CYBERSPACE [6] projects contributed to designing and developing a tool to support cyberattack attribution. This article presents the tool and discusses the results of its application in the context of recent cyberattacks, which show the tool's potential to increase the effectiveness and efficiency of attribution operations. Whilst our approach to translating the technical attack attribution problem to the supervised machine learning domain is similar to that of Noor et al. [7], we propose a way to overcome the lack of training data issue, use both low-level and high-level features to represent cyberattacks, and implement the tool as a contribution to a popular open-source CTI platform based on a de facto standard CTI language. This all significantly increases the chances of a high adoption and impact of the tool.

In this article, we first briefly review several noteworthy challenges of technical attack attribution, the data used in attack analysis, the connections between attribution and other key questions that arise in digital forensics and cyber incident response activities, and the earlier work on applying machine learning to the attack attribution problem. We then explain the technical approach, present the tool, which is based on a machine learning model and implemented as an extension of the OpenCTI platform [8], and show its performance in three cyberattack investigation cases (two of which were carried out by one of the CC-DRIVER and CYBERSPACE partners, WithSecure Corporation, and one by The DFIR Report group). The article concludes by discussing the challenges and directions for future work.

2. Technical Attack Attribution

When running analysis to identify the source of a cyberattack, investigators face multiple problems. Cybercriminals and other perpetrators often attempt to hide the origin of their attack network traffic by routing it via multiple links on the Internet, for instance, using proxy servers or onion-routing tools (such as Tor [9]) instead of directly connecting to the victim. They can utilize compromised or stolen devices to hide their identity, and they also rely increasingly often on tools commonly available on the victim's devices instead of using custom malware that can be fingerprinted and connected to their authors—this technique is known as "living-off-the-land" [10]. Attribution activities are further complicated by the growing popularity of the "Crime-as-as-a-service" mode of cybercriminal operations (malware-as-a-service, ransomware-as-a-service, DDoS-as-a-service, bulletproof hosting, etc.), the use of malicious code which is open-sourced, shared or stolen from other attackers (and sometimes even from state security agencies and security researchers [11,12]), and the use of malicious infrastructure (such as command-and-control servers) and TTPs previously attributed to other attackers. One should also note that CTI and other information crucial for attack attribution can be kept confidential by certain parties due to laws, contracts, and various—justified or unjustified—concerns.

Attack attribution is closely connected with several other questions typically asked by incident responders and investigators when trying to gain insights into the threat actor's operations in the victim's cyber estate. Good examples of such questions are:

- When did the threat actor breach the victim's systems and networks?
- What level of privilege does the threat actor have in the victim's systems and networks?
- What assets has the threat actor touched and potentially compromised?
- What is the impact of the breach?

So, essentially any data collected in an incident response operation can be useful for attribution-related analysis, while the data revealing the threat actor's capabilities, objectives, and behavior are of particular value. These data include:

- The Attacker's TTPs. The MITRE ATT&CK framework [13] is commonly used to structure and model this information.
- Indicators of Compromise (IOCs) and the attacker's infrastructure, such as the file hashes of malicious payloads and IP addresses which the attack traffic originates from or where the command-and-control (C2) servers are hosted.
- Malware analysis results (especially for victim-tailored malware with no public source code), which can provide high-value information. For instance, sometimes attackers make mistakes or leave traces in their malware code, and in other cases, they use evolving versions of the same malware for many years.
- Benign tools used by the attacker. These can be popular living-off-the-land binaries, such as Powershell and Windows Management Instrumentation (WMI), or other benign software found in the victim's estate that provides capabilities beneficial for the attacker.
- Exploited vulnerabilities, either previously unknown ones (zero-day vulnerabilities) or ones used earlier in other attacks. Exploitation techniques can be implemented in malware or by using appropriate benign tools, and we often see the same vulnerabilities used in multiple attacks conducted by the same threat actor.
- Attack metadata, such as the times when the attacker communicates with the victim's systems (which can hint at the attacker's geographical location) or information about the victim (as their operations, core business domains, location, etc., can reveal the attacker's objectives).

Given the nature of the attack attribution problem, an obvious approach is to look for similarities in data collected from attacks and about attackers. Identifying, ranking, and aggregating such similarities in large volumes of highly heterogeneous data is, however, time-consuming for investigators and requires expertise and experience. So, the growing number and sophistication of cyberattacks necessitate analysis automation, and machine learning techniques are a natural choice.

While the use of machine learning has recently been very popular in attack detection and malware analysis methods, it seems very few reports are available on its applications to cyberattack attribution.

Han et al. [14] implemented WHAP, a web-hacking profiling system that uses a simple similarity measure for hacking cases, which is based on heuristically assigned similarity weights for selected features (such as IP addresses and domain names) and Case-Based Reasoning. While the use of feature vectors for representing website hacking cases (which are just one type of cyberattack) and the defined similarity measure for those vectors are the only connections of the proposed approach to machine learning, conceptually, it can be extended to attack attribution methods by utilizing similarity search and clustering based on "learning from data". Even for website attacks only, this would, of course, require a major effort.

Noever et al. [15] presented a Random Forest classifier for attributing attack techniques (such as backdoor, man-in-the-middle, ransomware, and DoS) to the types of threat actors (organized crime, nation state, hacktivist, unknown). While this approach can be relevant, e.g., for policy discussions, it does not have the attribution of specific cyberat-

tacks to specific threat actors as its objective and would have little utility in real-world cyberattack investigations.

Noor et al. [7] present a framework for attributing unstructured (natural language) CTI reports and documents. Since "low-level Indicators of Compromise (IOCs) are rarely re-used and can be easily modified and disguised resulting in a deceptive and biased cyber threat attribution" [7] (p. 227), their work focuses on common high-level attack patterns (i.e., TTPs) for mapping a CTI report to a threat actor. With the labels for high-level attack patterns taken from the MITRE ATT&CK taxonomy, Latent Semantic Analysis (LSA) is used to index CTI reports with relevant labels. Then, a small set of CTI reports collected from publicly available datasets and marked with the threat actors behind the reported cyberattacks is used to train several machine learning models for attributing new reports. Although some of the models show a very good performance in the cross-validation tests, this is likely explained by the training and validation dataset's toy size (and no evidence is provided for evaluating the models on cyberattack data outside of the used dataset). More generally, we think that fully focusing on high-level attack patterns and ignoring low-level indicators will result in poor real-world performance because (i) many attackers use very similar TTPs (e.g., in ransomware attacks); (ii) high-level TTPs are easy to mimic in false flag operations; (iii) low-level indicators are actually reused (mainly due to attacker mistakes or time and cost pressures) and very useful in such cases. We will further comment on this high-level vs. low-level balance issue in the "Discussion and Future Work" section.

The use of pattern recognition and anomaly detection methods for TTP and IOC extraction from raw log data was also proposed by Landauer et al. in [16], illustrated by system log data analysis. These methods could be a supportive ingredient in the attack attribution process, providing additional features for CTI reports and attack descriptions to be attributed.

3. STIX-Based Attack Attribution Approach

A key objective defined in both CC-DRIVER and CYBERSPACE projects was to produce tools for following the threat landscape and actors (CTI management) and for investigating cyberattacks (digital forensics). Technical attack attribution capabilities fit naturally within this toolkit. As fully automated, reliable attack attribution is an infeasible goal, we chose to build an attack attribution recommender, based on the Structured Threat Information eXpression [17] (STIX 2) language, implemented as an OpenCTI [8] extension, aiming to guide incident investigators and significantly reduce their efforts.

To facilitate the process of identifying threat actors responsible for cyberattacks, the problem was framed as follows: Design and implement a machine learning model that takes a bundle of STIX 2 objects representing adversarial operations as input and predicts the most probable threat actors behind the operations.

Here, "Bundle" is a STIX 2 term that refers to a collection of STIX 2 objects. While, in principle, any STIX 2 entities can be included in a bundle, we started with the important special case in which a bundle is a set of "incidents" observed in a given (attacked) organization in a given timeframe. In STIX 2, such "incidents" represent information collected during attack investigation activities (usually conducted by LEAs, CERTs, or companies providing incident response services).

"Threat actors" refer to individuals, groups, or organizations which operate in cyberspace with a malicious intent. We, however, chose to build our recommender model to predict "intrusion sets" (which can subsequently be mapped to threat actors and then to identities, e.g., by LEAs) in order to provide greater flexibility. Cyberattacks are often leveraged by threat actors as part of a coordinated campaign against a specific target to achieve a specific objective. An entire attack package consisting of multiple campaigns sharing properties and behaviors and believed to be orchestrated by a single threat actor is represented in STIX 2 as an "intrusion set", and there are advantages in reasoning about attribution in terms of intrusion sets. For example, the threat actor behind a given attack may not be known, but their multiple operations can be grouped together in an intrusion

set, and then a new attack can be attributed—technically, without involving identities—to that intrusion set. This form of attribution almost always helps when responding to the attack and mitigating its impact, and it can also be a useful step in discovering the attackers' identities. We note that a threat actor can move from one intrusion set to another, changing their TTPs, or they can "utilize" multiple intrusion sets at the same time. The attribution relationships in STIX 2 are shown in Figure 1.

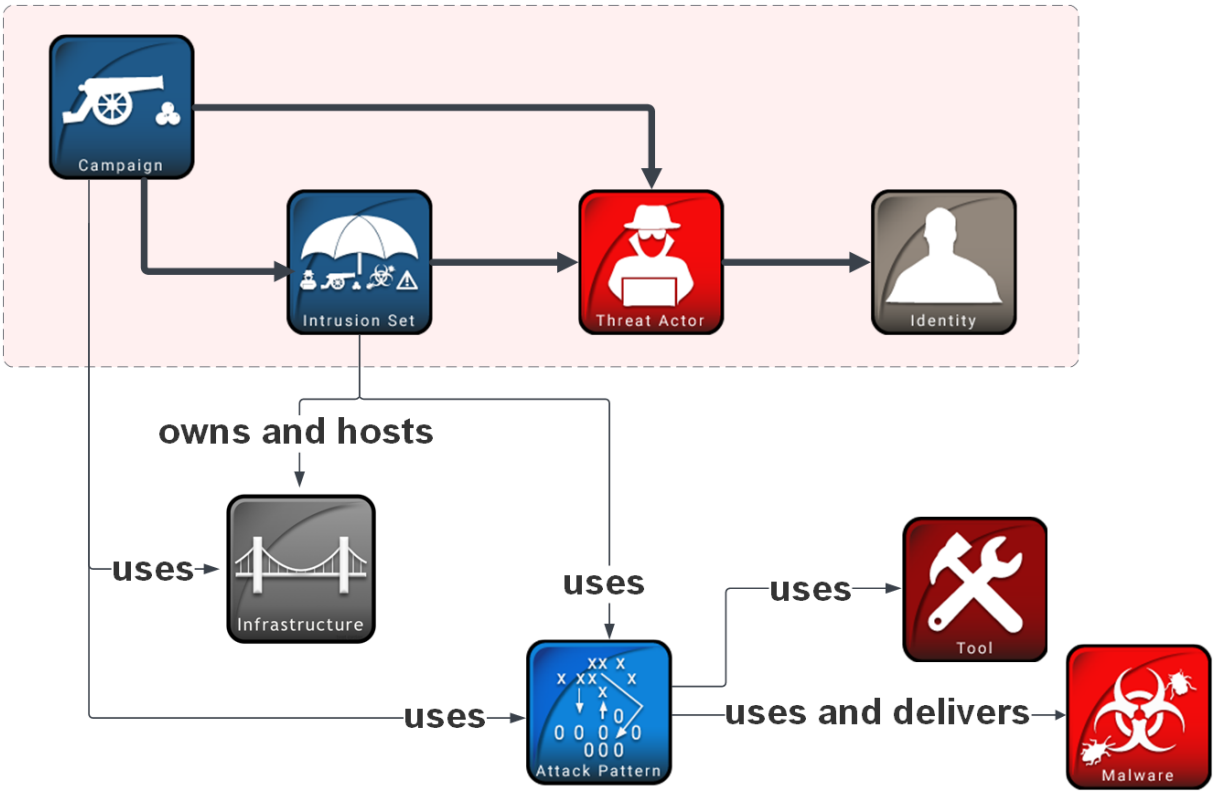

Figure 1. Attributed-to relationships in STIX 2 are shown with the arrows in bold (the icons are taken from https://github.com/MISP/intelligence-icons, accessed on 21 January 2024).

Our preliminary investigations confirmed that obtaining sufficiently large incident datasets to train a good attribution support model would be challenging, particularly because such datasets are often considered highly confidential. So, in parallel to extending the data collection, we simplified our problem to predicting an intrusion set for a single incident (instead of a bundle), as shown in Figure 2.

Figure 2. The simplified version of the attribution problem (the icons are taken from https://github.com/MISP/intelligence-icons, accessed on 21 January 2024).

For intrusion sets, a collection of over 350 entities (identified by their names, such as "APT28" or "Lazarus") was obtained from MITRE [18], AlienVault [19], Malpedia [20], and WithSecure. To compensate for the shortage of incident data, we chose to rely on the data augmentation approach, generating synthetic incidents based on the data from available intrusion sets. In producing incidents, specific rules designed together with cybersecurity experts, guided by their experience and observations, were followed:

- Incident data reuse the elements (properties or related objects) present in a specific intrusion set.
- The number of elements should be between 10 and 50 per incident, following a beta-binomial distribution with the median value around 15.
- From the elements present in an intrusion set, the following STIX 2 objects are reused: TTPs (up to 50%), tools (up to 20%), malware (up to 20%) and others (up to 10%). "Others" here include indicators, locations, and so on (all the entities that can be found in the intrusion set). The numbers in brackets indicate the upper bounds on the share of reused elements of a given type. However, if an intrusion set does not have, for example, "tool" elements at all, we will end up having zero tools added to synthetic incidents. With the chosen upper bounds, the actual numbers of attributes of a given type are selected uniformly at random.
- To keep the synthetic dataset balanced, each non-empty intrusion set is used to generate the same number of incidents.

Using this approach, hundreds of thousands of synthetic incidents can be produced from the available intrusion sets, and those form the main body of a labeled dataset for supervised learning. It is split into training and testing sets, where the testing (validation) set has 20% of the data, with the remainder used for training a model. At the time of writing this article, the incident dataset consisted of approximately 270 thousand entries.

The incident data are preprocessed for training and validation by applying CountVectorizer from the scikit-learn library [21] as a one-hot encoder, mapping incidents to sparse binary vectors: the entity IDs and names observed in at least one of the incidents are used as our features, and for a given incident the value of a specific feature is 1 if the corresponding entity ID or name is present in the incident and 0 otherwise. The properties and objects, from which we pick entity IDs and names, were informally introduced in the discussion of the data collected in incident response operations in Section 2. The complete list is:

- Attack Patterns (https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.html#_axjijf603msy, accessed on 21 January 2024)
- Malware (https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.html#_s5l7katgbp09, accessed on 21 January 2024)
- Tools (https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.html#_z4voa9ndw8v, accessed on 21 January 2024)
- Identities (including Sector entities) (https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.html#_wh296fiwpklp, accessed on 21 January 2024)
- Locations (including Country and Region entities) (https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.html#_th8nitr8jb4k, accessed on 21 January 2024)
- Vulnerabilities (https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.html#_q5ytzmajn6re, accessed on 21 January 2024)
- Indicators (including File, IPv4 address, Domain Name, and Process entities) (https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.html#_muftrcpnf89v, accessed on 21 January 2024)

An example of the IDs used as features is "T1059", which is an attack pattern specified in the MITRE ATT&CK knowledge base (https://attack.mitre.org/techniques/T1059/, accessed on 21 January 2024). Examples of the names used as features are "MagicRAT" (a malware described at https://blog.talosintelligence.com/lazarus-magicrat/, accessed on 21 January 2024) and "WinRAR" (a benign tool, malicious use of which is described at https://attack.mitre.org/techniques/T1560/001/, accessed on 21 January 2024). At the

time of writing this article, the incident dataset entries were represented by approximately 10 thousand (binary) features. We ran several rounds of experiments with three shallow multi-class classification models (with the same collection of intrusion sets but with the synthetic incidents generating algorithm that evolved from round to round): naive Bayes (the Bernoulli version was used, as incidents are represented as binary vectors), logistic regression, and random forest. In the final round, with the incident generation algorithm considered optimal by the experts (presented earlier in this section), all the models showed similar results, with a classification accuracy of around 0.97. The Bernoulli naive Bayes classifier was selected as the attribution recommender model primarily because (i) it has a significantly shorter training time than the other two classifiers (important for frequent retraining, which is a likely scenario), and (ii) its predictions are easy to interpret for incident investigators. Of course, the good observed performance of the models can be due to the synthetic nature of the data, so we are collecting more real-world incident data and planning further extensive modeling and validation experiments.

4. The Tool Design and Processing Flow

The attack attribution tool runs as an OpenCTI extension. The OpenCTI platform, with its growing user community and convenient framework for extending the platform's capabilities, has become a popular choice for storing, analyzing, and sharing both CTI and digital forensics data collected in cyber incident investigations. STIX 2, the underlying OpenCTI data format, allows for a rich representation of incidents as collections of observables and associated entities (such as TTPs, malware, and command-and-control infrastructure), combining high-level, abstracted views of attacks with relevant technical details.

The attribution tool is implemented as a connector [22] of OpenCTI and is available as an open-source contribution to the OpenCTI Platform project [23]. It is written in Python and uses the pycti Python library [24] to call the OpenCTI API. The tool is packaged as a Docker container, which is typical for OpenCTI connectors.

At runtime, the attribution tool runs as a standalone process and subscribes to the OpenCTI platform as an internal enrichment connector with a callback method for message processing. When a user wants to attribute an incident object and selects the attribution button in the platform GUI, an attribution request message is sent to the message queue, and subsequently the callback method is executed. The tool then uses the OpenCTI platform API to collect the relevant information of the incident object under analysis and of the objects in direct relationship with it, which is then passed to the attribution recommender model. To return the intrusion sets predicted by the model to the user, the tool creates a "note object" and attaches it to the incident object. Such a note object contains the top three intrusion set names with the corresponding model's confidence values and the links to the intrusion set objects in the OpenCTI database. There is also a configuration setting in the attribution tool for automatically creating a relationship between an incident object and the top predicted intrusion set if the model's confidence value for the latter is above a chosen threshold.

The connector periodically checks whether new data in the OpenCTI database should be used for retraining the attribution recommender model. For example, a new intrusion set is an obvious reason for retraining.

The tool processing flow is shown in Figure 3, and further details can be found at [23].

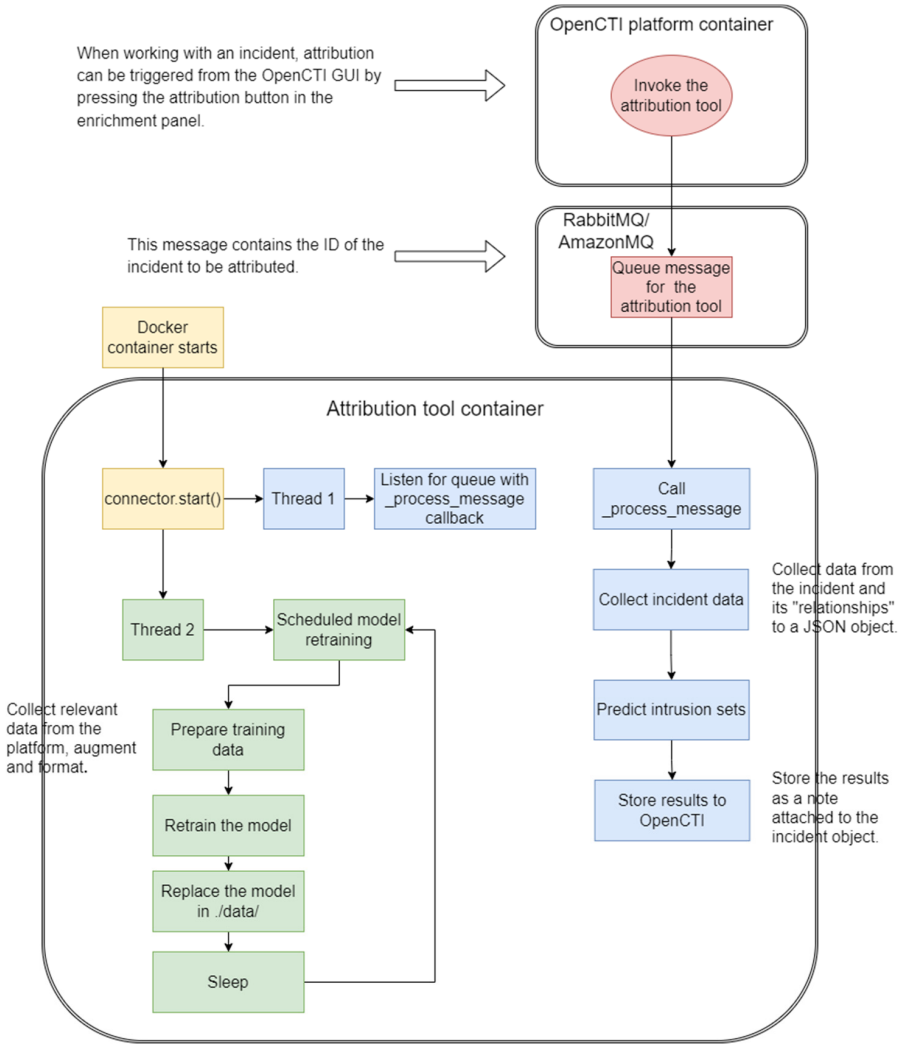

Figure 3. The attribution tool as an OpenCTI platform extension.

5. Attribution Results for Real-World Incidents

The data expression power of STIX 2 and the growing popularity of OpenCTI explain why an increasing number of incident response operations by WithSecure, a major European provider of cybersecurity services and solutions, rely on OpenCTI for data management and analysis. This recently gave us an opportunity to validate the attack attribution tool as part of two real-world attack investigation engagements.

5.1. The "No Pineapple!" Incident

The first attack, codenamed "No Pineapple!" by the WithSecure's Threat Intelligence team, transpired to be part of a sophisticated campaign targeting public and private sector research organizations, the medical research and energy sectors, as well as their supply chains. WithSecure's engagement started when a threat hunt in a customer estate identified beaconing [25] to a Cobalt Strike C2 server. Since the C2 server IP address was previously listed as an IOC for the BianLian ransomware group and a few other details also pointed to their involvement, the initial (low-confidence) assessment of the WithSecure's experts was that they were dealing with a potential ransomware incident. However, as more attacker tools, techniques, and actions were collected from the customer environment, it became evident that the main objective of the attack was espionage, and a North Korean state-sponsored threat actor was behind it. Notably, the attacker made a concerted effort at hiding their traces, clearing logs and deleting files, tools, and other indicators of their presence [26].

The collected digital forensics data were added to OpenCTI as an incident object representing the details of a single attack against a single organization. The object has quite a rich set of relationships, as can be seen in Figure 4.

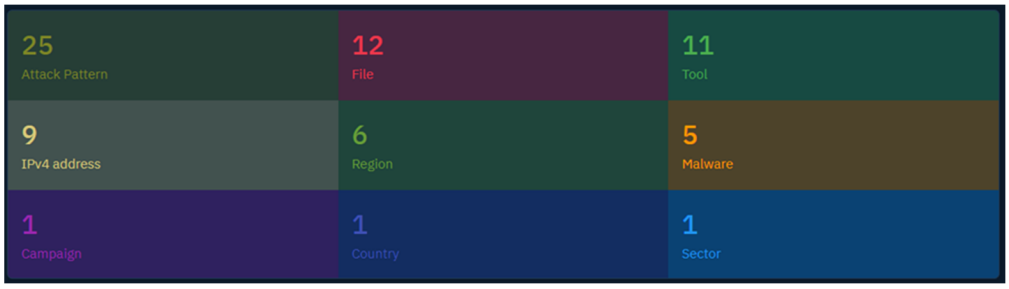

Figure 4. Relationships of the "No Pineapple!" incident as seen in OpenCTI.

We then applied the attack attribution tool to the "No Pineapple!" data and received the "Lazarus" intrusion set associated with a North Korean state-sponsored threat actor on the top of the list. It should be noted that at the time of the tool validation experiment, the Lazarus intrusion set was not updated with the "No Pineapple!" investigation data but represented the state of knowledge prior to the investigation.

The top three results reported by the tool (with the respective model confidence values) were:

1. Lazarus Group: 0.996186486423268.
2. Elephant Beetle: 0.003794891776652858.
3. APT29: 0.00001862067805979746.

So, the Lazarus group was suggested by the model as the most probable intrusion set for "No Pineapple!" with an overwhelming confidence, and this was subsequently confirmed by the WithSecure's experts. Elephant Beetle, which is a financially motivated cybercrime group, was the second model's pick. While the model confidence for the Elephant Beetle intrusion set is low, we note that it shares a set of common attack techniques with Lazarus, including blending in with the environment; deploying JSP web shells (JSP file browser, in particular); and operating from temporary directories. It also exploits known vulnerabilities in public-facing devices to gain initial access, although we are not aware of any specific vulnerabilities exploited by both Lazarus and Elephant Beetle. That is where the similarities end. Elephant Beetle is known to target different geographies, their operations have been financially motivated, and they often target web services and their components.

5.2. The "Black Basta" Incident

Another recent intrusion investigation by the WithSecure's Incident Response team also brought an OpenCTI incident object with a large number of relationships: 4 Attack Patterns, 5 Malware entities, 2 Tools, 19 IPv4 addresses, 9 Files, 5 Domain Names, 5 Processes, 3 Regions, and 1 Country. The top three results reported by the tool in this case (with the respective model confidence values) were:

1. Black Basta: 0.9919987932223466.
2. GCMAN: 0.0005211614823012417.
3. Saaiwc Group: 0.0005211614823012417.

The Black Basta intrusion set suggested by the tool was confirmed by the experts. In the OpenCTI instance that was used for the incident attribution, there were 21 Attack Patterns, 3 Malware entities, 6 Regions, and 2 Countries associated with this intrusion set.

Connecting the incident to a specific threat actor is more challenging. The Black Basta group is a ransomware operator and is believed to be a Ransomware-as-a-Service (RaaS) criminal enterprise [27], selling their ransomware and accompanying infrastructure to other cybercriminals. The "service package" can also include playbooks and other tools for

conducting attacks. If their "customers" fully rely on such tools and strictly follow such playbooks, their operations will be practically indistinguishable (in particular, difficult to distinguish from the core Black Basta group operations). So, the high confidence value returned by the attribution tool for the Black Basta intrusion set did not help much with identifying the threat actor behind the incident. At the same time, this high value helped the Incident Response team in carrying out the remediation operations.

5.3. Externally Attributed Case

In addition, we tested the tool on a publicly available incident report from The DFIR Report collection [28]. Based on this report, a new incident object and its relationships were added to the OpenCTI instance: 19 Attack Patterns, 4 Tools, 3 Vulnerabilities, 2 Domain Names, 2 IPv4 addresses, and 6 other Indicators. The top three results reported by the tool for this incident were:

1. Magic Hound: 0.9833991991845277.
2. Blue Mockingbird: 0.0010974177998640602.
3. FIN10: 0.0010974177998640602.

The Magic Hound intrusion set is commonly associated with the threat actor named APT35 by Mandiant or Phosphorus by Microsoft (currently tracked by Microsoft as Mint Sandstrom), and The DFIR Report analysts manually attributed the incident to Phosphorus, based primarily on the observed attack patterns. In the OpenCTI instance used for testing the attribution tool, the Magic Hound intrusion set contained the following relationships: 131 Attack Patterns, 15 Malware entities, 13 Tools, 6 Vulnerabilities, 6 Regions, 2 Countries, and 3 Sectors.

6. Discussion and Future Work

The results obtained so far indicate that the approach of building machine learning models for attributing STIX 2 incidents to intrusion sets is promising and can bring significant value to incident investigators and responders. The reliability of such models, especially when attackers actively work to counter attribution efforts using false flags and other techniques, critically depends on the availability of sufficiently rich incident data and on a suitable balance between high-level attack patterns and attributes and low-level indicators in the incident data representation. While STIX 2 is good for expressing TTPs, malware, tools, exploited vulnerabilities, targeted geography and sectors at a certain level, more subtle details—such as malware code similarities, custom passwords, developer host information, the attacker's email language, geopolitical objectives, malicious domain registrar and registrant information—are not supported yet.

We see several ideas to explore for improving the attack attribution tool:

- Acquiring more real-world incident data, preferably with attribution labels (but even unlabeled incidents can be useful), instead of heavily relying on synthetically generated incidents. (We showed, nevertheless, that even with synthetic incident datasets generated in a suitable manner, one can build practically useful attribution models.)
- If many organizations agree to combine their incident data, a high-quality attribution model can likely be trained, but incident data are often highly sensitive. One way to address the data confidentiality issue is to train a model in a federated learning manner [29] on data of multiple organizations. In particular, joint efforts with the FATE project [30] working on collaborative confidentiality-preserving learning on CTI data can be considered.
- Use of inherited STIX 2 relationships (through the OpenCTI rule engine). Currently, only the data of direct neighbors, i.e., first-level relationships, are used in the model for both incidents and intrusion sets. For example, a file associated with an incident may have another relationship with a custom directory in which this file was located. If the same directory is associated with other files, this information may be valuable for attribution but is currently ignored.

- STIX 2 supports timestamps which can be used for building a timeline of the attacker's actions. Because most of the incident data in our model training sets is produced synthetically from the intrusion sets, timestamps are currently ignored. Collecting timestamps whenever possible and including them in modeling should be explored for utilizing attack timelines in attribution.
- Controlling the weights of features in the incident representation. Currently, the influence of specific features is learned implicitly when a model is trained. Combining the data-driven approach with expert-defined rules could be explored to mitigate the impact of biases and other imperfections in the training sets.

In conclusion, we would like to emphasize that even when large and clean training datasets are available, attack attribution models will make mistakes and can be deceived by skillful and determined attackers. Therefore, such models should primarily be used in the recommendation mode, with human experts verifying their output.

Author Contributions: Conceptualization, S.R. and A.K.; methodology, D.K.; software, D.K., M.P. and S.R.; validation, S.R.; investigation, S.R., D.K. and M.P.; data curation, D.K.; writing—original draft preparation, A.K., S.R. and M.P.; writing—review and editing, A.K.; visualization, S.R.; project administration, A.K. All authors have read and agreed to the published version of the manuscript.

Funding: Parts of this research were supported by the CC-DRIVER project funding received from the European Union's Horizon 2020 research and innovation program under grant agreement No 883543 and by the CYBERSPACE project funding received from the European Union's Internal Security Fund—Police (ISFP) program under grant agreement No 101038738. The APC was funded by CYBERSPACE.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The STIX 2 data used for producing the recommender model were aggregated from the following sources: MITRE database. The data are stored in GitHub [18] and publicly available. Third-party restrictions apply to the use of the data (used under license for the presented research). AlienVault. The data are publicly available via the AlienVault service [19]. Third-party restrictions apply to the use of the data (used under license for the presented research). A free account is required to access the data. Malpedia. The data are publicly available on the Malpedia website [20]. Third-party restrictions apply to the use of the data (used under license for the presented research). A free account is required to access the data. All these data sources have open-source connectors for OpenCTI to facilitate data ingestion. The WithSecure data used for the presented research are not publicly available due to legal restrictions but constitute only a small portion of the overall dataset.

Acknowledgments: The authors would like to thank their WithSecure colleagues from the Threat Intelligence and Incident Response teams for the help and valuable discussions. We also express our gratitude to the CC-DRIVER and CYBERSPACE project partners for their support. Special thanks go to Timothy West for his helpful comments.

Conflicts of Interest: The authors declare no conflict of interests. The funders had no role in the design of the presented research; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Promoting Accountability in Cyberspace. Available online: https://www.rand.org/nsrd/projects/cyberspace-accountability.html (accessed on 15 December 2023).
2. Kastelic, A. Non-Escalatory Attribution of International Cyber Incidents: Facts, International Law and Politics. Available online: https://unidir.org/publication/non-escalatory-attribution-of-international-cyber-incidents-facts-international-law-and-politics/ (accessed on 2 December 2023).
3. Poireault, K. What's in a Name? Understanding Threat Actor Naming Conventions. Available online: https://www.infosecurityeurope.com/en-gb/blog/threat-vectors/understanding-threat-actor-naming-conventions.html (accessed on 2 December 2023).

4. Tran, D. The law of attribution: Rules for attributing the source of a cyber-attack. *Yale JL Tech.* **2018**, *20*, 376–441. Available online: https://yjolt.org/sites/default/files/20_yale_j._l._tech._376.pdf (accessed on 5 June 2023).
5. The EU-Funded CC-DRIVER Project. Available online: https://www.ccdriver-h2020.com/ (accessed on 11 June 2023).
6. The EU-Funded CYBERSPACE Project. Available online: https://cyberspaceproject.eu/ (accessed on 11 June 2023).
7. Noor, U.; Anwar, Z.; Amjad, T.; Choo, K.R. A machine learning-based FinTech cyber threat attribution framework using high-level indicators of compromise. *Future Gener. Comput. Syst.* **2019**, *96*, 227–242. [CrossRef]
8. Open Cyber Threat Intelligence Platform—An Open Source Platform for Managing Cyber Threat Intelligence Knowledge and Observables. Available online: https://www.filigran.io/en/solutions/products/opencti/ (accessed on 11 June 2023).
9. Tor Project. Available online: https://www.torproject.org/ (accessed on 10 June 2023).
10. Living off the Land: Attackers Leverage Legitimate Tools for Malicious Ends. Available online: https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/living-land-legitimate-tools-malicious (accessed on 8 June 2023).
11. Goodin, D. Stolen NSA Hacking Tools Were Used in the Wild 14 Months before Shadow Brokers Leak. Available online: https://arstechnica.com/information-technology/2019/05/stolen-nsa-hacking-tools-were-used-in-the-wild-14-months-before-shadow-brokers-leak/ (accessed on 8 June 2023).
12. Cobalt Strike, a Penetration Testing Tool Abused by Criminals. Available online: https://www.malwarebytes.com/blog/news/2021/06/cobalt-strike-a-penetration-testing-tool-popular-among-criminals (accessed on 8 June 2023).
13. MITRE ATT&CK Framework. Available online: https://attack.mitre.org/ (accessed on 10 June 2023).
14. Han, M.L.; Han, H.C.; Kang, A.R.; Kwak, B.I.; Mohaisen, A.; Kim, H.K. WHAP: Web-hacking profiling using case-based reasoning. In Proceedings of the 2016 IEEE Conference on Communications and Network Security (CNS), Philadelphia, PA, USA, 17–19 October 2016. [CrossRef]
15. Noever, D.; Kinnaird, D. Identifying the Perpetrator: Attribution of Cyber-attacks based on the Integrated Crisis Early Warning System and the VERIS Community Database. In Proceedings of the 2016 International Conference on Social Computing, Behavioral-Cultural Modeling & Prediction and Behavior Representation in Modeling and Simulation, Washington, DC, USA, 28 June–1 July 2016; Available online: http://sbp-brims.org/2016/proceedings/CP_136.pdf (accessed on 5 June 2023).
16. Landauer, M.; Skopik, F.; Wurzenberger, M.; Hotwagner, W.; Rauber, A. A framework for cyber threat intelligence extraction from raw log data. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019. [CrossRef]
17. STIX Version 2.1 OASIS Standard. Available online: https://docs.oasis-open.org/cti/stix/v2.1/stix-v2.1.html (accessed on 10 June 2023).
18. MITRE Data. Available online: https://raw.githubusercontent.com/mitre-attack/attack-stix-data/master/enterprise-attack/enterprise-attack.json (accessed on 15 June 2023).
19. AlienVault, The World's First Truly Open Threat Intelligence Community. Available online: https://otx.alienvault.com/ (accessed on 15 June 2023).
20. Malpedia. Available online: https://malpedia.caad.fkie.fraunhofer.de/ (accessed on 15 June 2023).
21. The Scikit-Learn Library. Available online: https://scikit-learn.org/stable/ (accessed on 2 December 2023).
22. Hassine, S. OpenCTI Ecosystem Snapshot. Available online: https://blog.filigran.io/opencti-ecosystem-snapshot-c5fcce96bf5b (accessed on 2 December 2023).
23. OpenCTI Attribution Tools Connector. Available online: https://github.com/OpenCTI-Platform/connectors/tree/master/internal-enrichment/attribution-tools (accessed on 2 December 2023).
24. Python API Client for OpenCTI. Available online: https://pypi.org/project/pycti/ (accessed on 2 December 2023).
25. Purple Team: About Beacons. Available online: https://www.criticalinsight.com/resources/news/article/purple-team-about-beacons (accessed on 12 June 2023).
26. Ruohonen, S.; Robinson, S. No Pineapple!—DPRK Targeting of Medical Research and Technology Sector. Available online: https://labs.withsecure.com/publications/no-pineapple-dprk-targeting-of-medical-research-and-technology-sector (accessed on 12 June 2023).
27. Threat Profile: Black Basta. Available online: https://www.hhs.gov/sites/default/files/black-basta-threat-profile.pdf (accessed on 18 December 2023).
28. The DFIR Report: PHOSPHORUS Automates Initial Access Using ProxyShell. Available online: https://thedfirreport.com/2022/03/21/phosphorus-automates-initial-access-using-proxyshell/ (accessed on 2 December 2023).
29. McMahan, B.; Ramage, D. Federated Learning: Collaborative Machine Learning without Centralized Training Data. Available online: https://ai.googleblog.com/2017/04/federated-learning-collaborative.html (accessed on 12 June 2023).
30. FATE (Federated AI Technology Enabler) Project. Available online: https://github.com/FederatedAI/FATE (accessed on 12 June 2023).