MSc Artificial Intelligence Thesis

Temporal and Ontological Threat Intelligence Analyses

Author:
Anniek Jansen

Supervisor:
Fabio BN Massacci

Second Reader:
Firstname Lastname

September 20, 2024

Abstract

The abstract should briefly summarize the contents of the paper.

Keywords: Uncertainty Representation, Ontology, NCSC, APT, Temporal Analysis, ...

Acknowledgment

The acknowledgment should briefly thank the people involved with the thesis.

Table of Contents

Abbreviations

AAO Avionics Analytics Ontology.

CASE Cyber-investigation Analysis Standard Expression.
CVE Common Vulnerabilities and Exposures.

NVD National Vulnerability Database.

UCO Unified Cyber Ontology.
URREF Uncertainty Representation and Reasoning Evaluation Framework.

NCSC Attribute Translations

Beschrijving Description.
Beveiligingsadviezen Security advisories.

Kans Likelihood.
Kansomschrijving Likelihood description.

Mogelijke oplossingen Possible solutions.

Nationaal Cyber Security Centrum (NCSC) National Cyber Security Center (NCSC).
NCSC inschaling NCSC classification.

Platformen Operating Systems.

Schade Impact.
Schadeomschrijving Impact description.

Toepassingen Product.

Uitgiftedatum Publication date.

Versie Version.
Versies Versions.

1 Introduction

Explain the main purpose of the investigation. .....

First, an explanation of the main topics, Threat Intelligence and Uncertainty Representation, is provided to set the foundation for understanding the subsequent sections. Sections 2 and 3 cover the literature review and the corresponding findings. A concise conclusion analysing areas of interest for further research is presented in Section 4. Following this, Section 5 details the exploration and processing of two national security datasets. Next, Section 6 explains the methodology for mapping the processed data into the UR-REF ontology. The results of the study are shown in Section 8. Lastly, this thesis concludes with an in-depth discussion and final conclusions, presented in Sections 9 and 11, respectively.

1.1 Threat Intelligence

1.2 Uncertainty Representation

2 Literature Review

This literature review focusses on investigating the state of knowledge on uncertainty representation in the field of Threat Intelligence. To achieve this goal, a keyword-based search was conducted to retrieve relevant sources on the topic. The collected sources were then screened based on their title, abstract, and a set of selection criteria. The final sources were clustered according to the selection criteria to ultimately analyse each cluster using a thematic analysis strategy to determine the general findings of different uncertainty representation techniques. This process is shown in Figure 1 and is similar to the methodology used in [15].

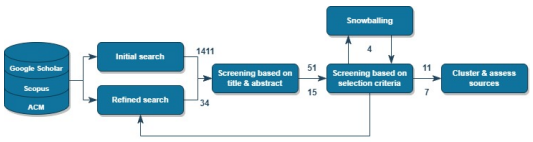


Fig. 1: Literature review process.

2.1 Keyword Search

To gather a comprehensive collection of relevant sources on the topic of uncertainty representation within the field of Threat Intelligence, the first step involved creating relevant keyword-based searches. Initially, three keyword-based queries were executed to retrieve sources from Google Scholar, Scopus, and the ACM Digital Library. The keyword-based searches are listed in Table 1, along with the total results and the number of relevant sources that were ultimately selected for the conclusion section of this review of the literature. Searches on Google Scholar produced an enormous volume of scholarly sources. For example, the first query generated a total of $3,940$ sources. Consequently, only the sources displayed on the first ten pages of each keyword-based query were included throughout this process. This selection was facilitated by the Google Scholar search result method, which prioritises relevance[1]. After collecting a total of $1,411$ sources from digital libraries using the first three keyword-based searches, the titles and abstracts were screened for source selection. The fourth keyword search was only implemented during the screening phase of the selection criteria, since more relevant sources addressing the subject of uncertainty representation itself were preferred. The sources retrieved from this refined search were screened similarly based on titles and abstracts.

2.2 Selection Criteria

The next step involved assessing sources using a predefined selection criteria to ensure that the articles are suitable for this literature review. The selection criteria are categorised into four main sections, namely technology, research method, study domain, and analysed measure. To be considered relevant, the source must contain all selection criteria. With the exception of technology, each selection criterion has three sub-criteria, as shown in Figure 2.

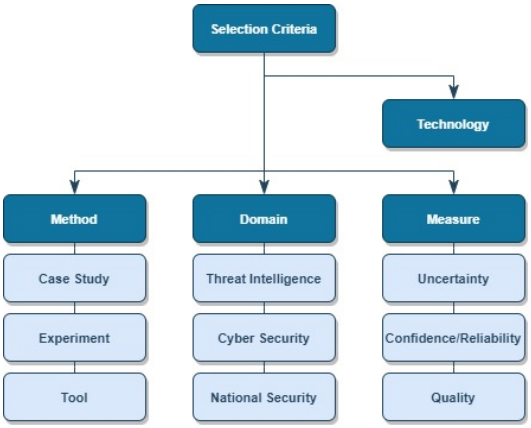


Fig. 2: The four selection criteria including the sub-criteria.

Technology. The first selection criterion is the technology used to represent the uncertainty. Unlike the other criteria, it does not entail strictly defined sub-criteria, as the purpose of this literature review is to investigate all the practical technologies already implemented in existing research. The techniques employed for metric representation are diverse, ranging from ontologies, Bayesian networks, and quantitative techniques to other examples such as vector representation, graph matching, explainable AI, and fuzzy description logic.

Method. The validity and utility of the conclusions drawn from a source are highly affected by the research method. The method provides a systematic approach to conducting the research and explaining the results. Given the objective of this study of discovering a working technique to represent uncertainty, methodologies are preferred that also focus on developing and testing a certain technique. Accordingly, the three sub-criteria of the

[1] Google Scholar Search Help: `https://scholar.google.com/intl/en/scholar/help.html`

Keyword Search	Google Scholar		Scopus		ACM DL	
[*threat intelligence* OR *security intelligence* OR *cyber intelligence*] AND [*AI* OR *ML**] AND [*representation* OR *encoding*] AND [***uncertainty* OR *bias* OR *error***]	3,940	2	237	2	97	1
[*threat intelligence* OR *security intelligence* OR *cyber intelligence*] AND [*AI* OR *ML**] AND [*representation* OR *encoding*] AND [***fusion* OR *aggregation* OR *merge***]	2,240	4	242	4	71	0
[*threat intelligence* OR *security intelligence* OR *cyber intelligence*] AND [*AI* OR *ML**] AND [*representation* OR *encoding*] AND [***predictive policing* OR *prediction***]	2,510	0	376	3	78	1
[*threat intelligence* OR *security intelligence* OR *cyber intelligence*] AND [***uncertainty representation* OR *representing uncertainty***]	38	7	6	4	1	0

Table 1: Total results found (left) and total relevant sources selected (right) per keyword-based search and per digital library, including duplicates. *The abbreviations “AI" and “ML", as well as the completely written terms “Artificial Intelligence" and “Machine Learning" were used.

methodology include instances of case studies, experiments, and tools. Methods such as surveys and interviews are excluded, as this study itself will also not implement these validation types.

Domain. The third selection criterion is the study domain of the source, specifically the field of interest the study aims at. Although the primary focus of this research is Threat Intelligence, it is also valuable to search for sources with closely associated domains that may also be relevant for this review. For example, while Threat Intelligence focusses on monitoring and countering security threats, cyber security aims to protect cyber systems from unauthorised access or cyber attacks in advance. According to a study on cyber security definitions, the most representative description would be as follows: “cyber security is the collection of tools, policies, security concepts, security safeguards, guidelines, risk management approaches, actions, training, best practices, assurance and technologies that can be used to protect the cyber environment and organisation and assets” [19]. Therefore, the technology methods used to represent uncertainty in the field of cyber security could also be beneficial for this literature review. In addition to Threat Intelligence and cyber security, a third sub-criterion for the study domain has been defined as national security. This domain contains sources that focus on counterinsurgency and military operations.

Measure. The goal of this research is to investigate uncertainty. However, uncertainty is a complex concept that is influenced by various factors. To address this ambiguity, we also collected sources that explore similar measures that also possess a degree of vagueness. Specifically, we obtain sources that study measures of confidence or reliability, as well as those that aim to describe data quality using metrics like accuracy, completeness, validity, uniqueness, and consistency.

2.3 Snowball Technique

During the screening process of the sources based on the selection criteria mentioned above, a direct snowball technique was implemented to identify additional sources relevant to the review of the literature. This was achieved by examining the in-text citations and utilising the reference list of the articles under consideration. The direct snowball approach resulted in the acquisition of four new sources.

2.4 Clustering

Including the articles collected using the direct snowball technique, a total of 18 sources were collected that are relevant to the topic of representing uncertainty in the area of Threat Intelligence. The initial searches yielded 11 interesting sources, while the refined search led to an additional 7 unique sources to complement this literature review. While great effort was made during this literature review process to find all relevant sources on the subject, it is possible that some sources were overlooked. Nevertheless, the 18 collected sources serve as a solid foundation for providing a comprehensive overview of the current state of knowledge on the representation of uncertainty in the domain of Threat Intelligence.

The findings of these 18 sources are explained by clustering based on the selection criteria. The sources are grouped together and eval-

Technology	Method			Domain			Measure		
	Case Study	Experiment	Tool	Threat Intelligence	Cyber Security	National Security	Uncertainty	Confidence / Reliability	Data Quality
Ontologies	[1][4][5][10][11][13]	-	[1][4][5][10][11][13]	[4][5][10][11][13]	[1]	-	[4][5][10][11][13]	[1]	-
Blockchain	-	[2][9]	[2][9]	[2][9]	-	-	-	[2][9]	-
Bayesian Networks	[4][5][10][11]	-	[4][5][10][11][21]	[4][5][10][11]	-	[21]	[4][5][10][11]	[21]	-
Vector Representation	-	-	[6][12]	[12]	[6]	-	-	-	[6][12]
Inexact Graph Matching	[7][8]	-	-	-	-	[7][8]	[7][8]	[7][8]	-
Explainable AI	-	[14]	[14]	[14]	-	-	-	-	[14]
Combining Data Functions	[16]	-	[16]	[16]	-	[16]	[16]	-	-
Expert-based Profiling	[17]	-	-	[17]	-	-	[17]	-	-
Weighted Evaluation	[18]	-	-	[18]	-	-	-	-	[18]
Description Logic	[20]	-	[20]	[20]	-	-	[20]	-	-
Total	**12**	**3**	**14**	**13**	**2**	**4**	**10**	**6**	**4**

Table 2: Sampled sources clustered per selection criteria for each technology.

uated according to common characteristics. An overview of the selection criteria per technology is shown in Table 2, including the total number of unique sources per selection criterion.

3 Literature Findings

3.1 Technology

Ontologies. Of the 18 sampled sources, 6 articles show that uncertainty representation can be carried out using ontologies. Most of the sources collected implement the Uncertainty Representation and Reasoning Evaluation Framework (URREF) ontology [4][5][10][11][13], which has been developed to assess and manage various aspects of uncertainty in information systems by implementing applicable criteria from the four classes of uncertainty evaluation form the URREF. Two of these studies incorporate the URREF ontology into a secondary ontology which is the Avionics Analytics Ontology (AAO) [10][11]. This integration enables the provision of semantic uncertainty for new data. Finally, one of the sources sampled in this cluster applies the Unified Cyber Ontology (UCO) [1]. The source introduces a specification language called Cyber-investigation Analysis Standard Expression (CASE), which not only aligns with UCO, but also expands upon it by incorporating confidence levels based on the Admirality Code Credibility Scale as an important property to represent uncertainty.

Key Finding: The URREF ontology is predominantly used for representing uncertainty. Integrating the URREF ontology with a secondary ontology can help with articulating the uncertainty associated with novel and more finely specified datasets.

Blockchain. Two of the 18 sources sampled introduce a blockchain-based system for Threat Intelligence that leverages multiple data feeds simultaneously to verify threat detection reliability [2][9]. In the first source a ranking system is created to categorize threats based on their reliability by accumulating multiple verifying files [2]. In the second source, each data input is analyzed by various Threat Detection Systems (TDS), resulting in the computation of an anomaly score based on the associated confidence levels for each threat [9]. Both the ranking system and the confidence computations contribute to evaluate the presence of uncertainty within the threat detection process.

Key Finding: A system built upon blockchain technology utilizes multiple data inputs for the purpose of assessing and ranking the reliability or confidence of threats.

Bayesian Networks. The utilization of a Bayesian Network (BN) as a technique for representing uncertainty in the context of Threat Intelligence has been researched in 5 of the 18 sampled sources. Noteworthy, four of these studies apply BNs in conjunction with the previously discussed URREF ontology [4][5][10][11]. The conditional probability tables of BNs can be used to represent the uncertainty of causal relations, while estimation metrics can be achieved by applying the URREF ontology. Different implementations of BNs are described in the sampled sources, among which the focus on estimations of expected probabilities [5] or entropy of BNs to quantify the uncertainty in the probabilities of an input variable [4]. The final source in the cluster does not integrate the BN with the URREF ontology. Instead, it presents a multi-level information framework based on Dynamic Bayesian Networks (DBN) to identify the Most Valuable Variables that could

reach the specific confidence threshold of a threat [21].

Key Finding: Bayesian Networks are a valueable tool for representing causal models when making decision under uncertainty. The majority of the examined sources implement BNs in conjunction with the URREF ontology, each source emphasizing distinct focus points.

Vector Representation. Of the sampled sources, two researches implement vector representation. The first source calculates distinct data quality metrics by selecting the most relevant vector values and aggregating them [6]. The other source computes an information quality (IQ) score that assesses accuracy, completeness, and uniqueness by utilizing the Common Vulnerability Scoring System (CVSS). Here, the system uses vectors to represent various attack parameters and their corresponding values [12].

Key Finding: Vectors have the ability to represent and aggregate the most relevant values associated with a threat.

Inexact Graph Matching. Two of the 18 samples sources research the feasibility to represent uncertainties through inexact graph matching. Both articles use a Truncated Search Tree algorithm that involves the computation of both probabilistic and possibilistic similarity scores for each node between a predefined template graph and the observed data graph [7][8]. The final aggregated similarity score indicates the confidence of the observation, thus providing a technique to represent uncertainty. The second article is more enhanced as both various crisp similarity measures are examined, as well as multiple fuzzy similarity measures [8].

Key Finding: Inexact graph matching can provide the confidence of an observation by calculating a similarity score for each node based on a predefined template graph and the observed data graph. Different similarity score methods exist.

Explainable Artificial Intelligence. The utlization of explainable AI is researched in one of the collected sources. The research proposes a framework known as FAIXID, which implements both Explainable Artificial Intelligence (XAI) and data cleaning techniques with the objective of enhancing the explainability and understandability of intrusion detection alerts [14]. The XAI component of the framework includes exploratory data analysis, dataset summarization, visualization methods, and a spectrum of explainability algorithms from the AIX360 toolkit. As a result, the framework enables Threat Intelligence analysts to make more informed decisions by gaining a profound comprehension of the underlying uncertainties inherent in the alerts [14].

Key Finding: Explainable Artificial Intelligence in combination with data cleaning techniques facilitates in illustrating the inherent uncertainties associated with threats.

Combining Data Functions. One of the 18 sampled sources, researches a Managing Uncertainty (MU) representation language that uses various features and combining functions to update information about threats in its knowledge base [16]. The combining data functions presented are the Bayesian theory, the Dempster-Shafer theory of evidence, fuzzy logic, and lastly the theory of endorsements. Using these combining functions, the MU ultimately represents uncertainty using the qualitative measures for each threat; “very-uncertain", “uncertain", “certain", and “very-certain" [16].

Key Finding: An expansive research approach involves combining diverse technologies. Each function contributes to an updated version of a threat’s knowledge base, thus its uncertainty.

Expert-based Profiling. One of the sampled sources examines the likelihood of a threat based on a threat profiling reference table designed for expert use. Both the history and motivation, as well as the final probability of a threat can be scored using the qualitative judgements “low", “low-moderate", “moderate-high" and “high" [17].

Key Finding: Experts provide qualitative assessments for both the historical context and underlying motivation associated with a threat, enabling the computation of the likelihood of a threat.

Weighted Evaluation Method. Among the 18 sources included in the sample, one source focuses on the implementation of a weighted evaluation method to calculate the weighted average of 10 different parameters of a cyber Threat Intelligence (CTI) source. Through this weighted sum

approach, the trust in the quality of the CTI source, and consequently the uncertainty, is continuously assessed and re-evaluated [18].

Key Finding: The weighted evaluation method continuously assesses and re-evaluates the trust and uncertainty of a cyber threat based on 10 distinct parameters.

Description Logic. One of the collected sources researches a formalism based on fuzzy description logic to effectively represent uncertain and vague cyber knowledge [20]. This formalism combines diverse subsets of the SROIQ description logic, enabling the formalism to represent probabilistic, possibilistic and fuzzy knowledge [20].

Key Finding: A formalism combining subsets of the SROIQ description logic can represent knowledge including uncertainty in cyber data.

3.2 Method

Case Study. On of the main research validation approaches is the use of case studies, where the authors extensively research the respective subjects using one or multiple examples. In this particular cluster, 12 of the sources are case studies, with 4 sources primarily focused on research related to air traffic control [4][5][10][11], 2 sources utilizing person identification examples [7][8], 1 source examining social media for crisis management [13], 2 sources investigating military missions [16][21], and the remaining 3 sources researching various cyber threat cases [17][18][20].

Key Finding: The sampled literature encompasses numerous researches centered on validation by case studies. It is noteworthy that the sources implementing blockchain, vector representation, and explainable AI technologies do not rely on case studies for validation.

Experiment. The least frequently employed research method is experimentation, in which multiple specific groups, models or variables are used to test a hypothesis. Among the 18 sampled sources, only three are validated by experiments. Specifically, two of these sources utilize the blockchain technology to experiment with the quantity of input feeds [2] or the aggregation of anomaly scores from different threat detection systems [9]. The third source evaluates both human analyst experiments, as well as a proxy evaluation experiment to test the efficiency of supporting explainable AI [14].

Key Finding: Experimentation is the least common research method of the sampled sources, with only three sources validated through experiments. Notably, no experiments are conducted for the prominent technologies mostly used to represent uncertainty, namely ontologies and Bayesian networks.

Tool. The predominant research validation approach in the sampled literature is the application of a tool. Of the 18 sampled sources, 14 sources utilize a specific tool, which includes the implementation of frameworks and architectures developed to provide researchers the components to execute models using programming interfaces. The different tools researched are the Cyber-investigation Analysis Standard Expression (CASE) [1], a blockchain-based cyber Threat Intelligence system architecture [2], the URREF ontology [4][5][10][11][13], a two-party privacy preserving protocol [6], the intrusion detection ORISHA platform [9], the Overt Vulnerability system OVANA [12], the Explainable Artificial Intelligence (XAI) and data cleaning framework named FAIXID [14], the Managing Uncertainty (MU) tool [16], a noval fuzzy description logic formalism [20], and a multi-level information fusion framework [21].

Key Finding: Tools serve as the primary validation method in the sampled literature. Notably, only three of the ten technologies lack research that incorporates a specific tool. Specifically, the inexact graph matching, expert-based profiling and the weighted evaluation method.

3.3 Domain

Threat Intelligence. The primary emphasis of this thesis is on the subject of Threat Intelligence. Consequently, it is not surprising that most of the sampled literature is applied in the Threat Intelligence domain. Of the 18 selected sources, 13 are oriented towards the process of collecting, analysing, and sharing information regarding potential cyber security threats and vulnerabilities. Some specific examples of Threat Intelligence discussed in the collected literature include intrusion detection [9], vulnerability assessment [12][14], and Threat Intelligence sharing [18].

Key Finding: The primary focus of the sampled literature is on Threat Intelligence. With the exception of the two sources employing the inexact graph matching technology, all the technologies include research focused around the identification of threats.

Cyber Security. The cyber security domain is closely related to the Threat Intelligence domain due to its common focus on cyber systems. However, within the context of this literature review, the sources that investigate cyber threats are clustered in the Threat Intelligence domain. In contrast, sources directed solely towards the handling process of any cyber data are classified according to the cyber security selection criterion. Therefore, only two of the sources sampled fall into the cyber security domain [1][6]. The first source discusses the CASE specification language, which provides a structure to represent and share details on how cyber information is handled, shared, analysed, and interpreted [1]. The second source investigates a protocol designed to assess the quality of cyber security data under certain privacy restrictions [6].

Key Finding: A strong connection between the cyber security and Threat Intelligence domains exists, leaving only two sources to the cyber security cluster, excluding those centered on threats. With the exception of ontologies and vector representations, all other technologies are not represented by articles applied in the cyber security domain.

National Security. Among the 18 sources sampled, 5 of the sources are applied in the national security domain. Here, the primary emphasis resides in the safeguarding of a nation's sovereignty and the protection of its citizens from various dangers, thus not limited to cyber threats. This means that it does not only include cyber threats. The sources in this cluster apply their research related to counterinsurgency, commonly referred to as COIN [7][8], and military missions [16][21].

Key Finding: Some of the sources in the sampled literature focus on national security, particularly on counterinsurgency and military missions. These researches exclusively apply Bayesian networks, inexact graph matching, and the combining data function technique, while the remaining technologies are not covered in the literature.

3.4 Measure

Uncertainty. The primary research emphasis is on representing uncertainty, and the majority of the 18 sources focus on this particular measure. Specifically, 10 of these sources employ uncertainty to convey the trustworthiness of data input and / or output. In addition to numerous sources researching the URREF ontology [4][5][10][11][13], two articles focus on representing uncertainty through confidence and similarity scores [7][8], while others apply combining functions to express qualitative measures [16], assess the likelihood of threats by expert-based profiling [17], or handle True/False description logic statements [20].

Key Finding: The aim of attention is on uncertainty representation, with most of the literature concentrating on this measure, particularly the research on ontologies and Bayesian networks. Only the sources implementing blockchain, vector representation, explainable AI, and the weighted evaluation method lack the focus on uncertainty.

Confidence / Reliability. Among the 18 sources, 6 are dedicated to the implementation of a confidence or reliability measure. Specifically, one source explores the Admiralty Code Credibility Scale to assign a confidence property [1]. The two blockchain-based studies focus on the computation of reliability and confidence scores by accumulating multiple data inputs [2][9]. Furthermore, the two sources specialised in graph matching employ various confidence and similarity measures to ultimately represent uncertainty [7][8]. Finally, the source researching Dynamic Bayesian networks implements a reliability score for each variable to analyse the significance of the variables in reaching a specified confidence threshold [21].

Key Finding: Using confidence or reliability scores as the main measure is researched in some of the collected sources. Four of the 10 technologies have one or two articles that focus on this. For the other technologies in the sampled literature this measure is absent.

Data Quality. Finally, four out of the 18 sources collected prioritise data quality as the main measure. Here, data quality is assessed by considering various essential metrics. The key recurring metrics are accuracy [12][14], completeness [6][12][14][18], consistency [6][14], timeli-

ness [6][14][18], uniqueness [6][12] and validity [6][18].

Key Finding: Combining diverse metrics provides insights into data quality. This measure is prevalent in the vector representation, explainable AI, and weighted evaluation method. Most technologies, notably ontologies and Bayesian networks, lack literature concerning data quality.

3.5 Summary

The compilation of the 18 sources collected provides a comprehensive perspective on existing research on the representation of uncertainty within the field of Threat Intelligence. The technologies used mostly are ontologies and Bayesian networks, but numerous alternative approaches exist. Table 3 presents a summary of the key findings derived from the sampled literature for each technology.

4 Deficiencies in Ontological Research

As noted in the previous section, most of the reviewed literature implements ontological research to represent uncertainty. Consequently, the results of the literature review suggest that ontologies hold significant promise among the ten distinct technologies examined. Revisiting Table 3, it becomes evident that the URREF ontology stands out as particularly noteworthy. However, the ontology cluster lacks validation through experiments, implementation within the national security domain, and evaluation using data quality measures. This thesis aims to address these deficiencies by performing temporal and ontological experiments using national security data and employing data quality evaluation metrics.

4.1 Research Questions

The core aspect of threat intelligence is assessing the likelihood of threat occurrences. Consequently, systems are developed to classify these chances. If the likelihood is not equal to 0 or 100 percent, some degree of uncertainty exists. This study explores how various aspects of threat intelligence information, such as temporal attributes and technical details, correspond and interact between two national security sources. Specifically, this research aims to investigate the alignment and consistency of threat intelligence data between APT and NCSC datasets to represent uncertainty through the following two research questions:

RQ1: *How do the reserved and exploitation times of the APT align with NCSC updates, and how do the justification and likelihood classifications of vulnerabilities impact the temporal differences between the publication and exploitation of threats?*

RQ2: *How do the product, operating system, and version details of the vulnerabilities from the APT align with the data from the NCSC, and how can this alignment be evaluated using the URREF ontology to assess the consistency and reliability of threat intelligence?*

5 NCSC Security Advisories

The first dataset comprises security alerts originating from the official website of the Dutch *Nationaal Cyber Security Centrum (NCSC)*[2]. The website explains that these security advisories are published in response to newly discovered vulnerabilities or identified threats. Each advisory includes detailed descriptions as well as possible consequences and solutions to the threats. Spanning from January 1, 2014, to the present day, these advisories are accessible and filterable using various criteria on the website.

The dataset available for this research includes security advisories over a longer period of time. The earliest recorded alert dates back to June 4, 2002, while the most recent entry is from January 20, 2023.

5.1 Exploratory Data Analysis

The initial step is to understand the dataset by analysing all variables. This entails acquiring the general insights, among which the number of rows and columns, the values and datatypes, as well as identifying any duplicates and missing values within the dataset.

Data Summary	
Instances	35,684
Attributes	15
Duplicates	6,861
Missing values	7,639

Table 4: Data summary of the original NCSC dataset.

Table 4 highlights that 6,861 of the total 35,684 rows in the original dataset are duplicates. This could suggest hidden information or errors

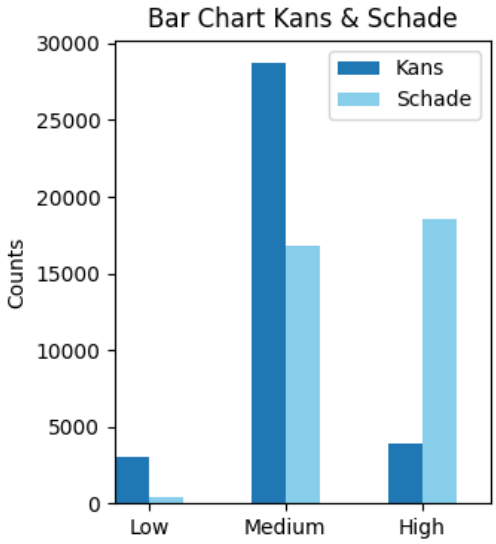
[2] NCSC Beveiligingsadviezen: `https://www.ncsc.nl/actueel/beveiligingsadviezen`

Technology	Sources	Summary
Ontologies	[1][4][5] [10][11][13]	The majority is validated using case studies in the domain of Threat Intelligence and based on the uncertainty measure from the URREF ontology. × *experiments, national security, data quality*
Blockchain	[2][9]	Both conduct experiments using tools within the realm of Threat Intelligence. Multiple data inputs are used to assess the confidence and reliability of threats. × *case studies, cyber & national security, uncertainty & data quality*
Bayesian Networks	[4][5][10] [11][21]	Most sources pair BNs with the URREF ontology. The sources are mainly validated by case studies for threat intelligence and focus on uncertainty. × *experiments, cyber security, data quality*
Vector Representation	[6][12]	Both tool-based sources use vectors to compute relevant values. One focusses on Threat Intelligence and the other on cyber security, both measuring data quality. × *case studies & experiments, national security, uncertainty & conf/reliability*
Inexact Graph Matching	[7][8]	The related studies are case studies within the domain of national security, with a focus on uncertainty and confidence/reliability by calculating similarity scores. × *experiments & tools, Threat Intelligence & cyber security, data quality*
Explainable AI	[14]	The source features the FAIXID tool, which performs numerous experiments within the Threat Intelligence domain for the evaluation of data quality. × *case studies, cyber & national security, uncertainty & confidence/reliability*
Combining Data Functions	[16]	This research is a case study addressing the representation of uncertainty in the Threat Intelligence domain using a tool that combines multiple data functions. × *experiments, cyber & national security, confidence/reliability & data quality*
Expert-based Profiling	[17]	The source illustrates a case study on the likelihood of a threat based on qualitative judgments, which ultimately represents uncertainty. × *experiments, cyber & national security, confidence/reliability & data quality*
Weighted Evaluation	[18]	The source validates the research through a case study that focusses on calculating the quality of the data using quantitative parameters for Threat Intelligence. × *experiments & tools, cyber & national security, uncertainty & conf/reliability*
Description Logic	[20]	The description logic tool to represent the uncertainty of threat intelligence data is validated through a case study. × *experiments, cyber & national security, confidence/reliability & data quality*

Table 3: Summary of the collected literature, including the selection criteria missing per technology.

in the data collection procedure. Of the $7,639$ missing values present, the majority, specifically $7,483$, are found in the `Schadeomschrijving` column. These missing values are not errors, but rather a consequence of data input only being reported if any of the sub-descriptions of the `Schadeomschrijving` are true. If all the sub-descriptions are not applicable to the corresponding advisory, no data input is shown, thus causing missing values. The other attributes containing missing values are `Beschrijving` and `Mogelijke oplossingen`, with a total of 95 and 61 missing values, respectively.

The attribute names of the dataset were found to be inconsistent with the terminology used on the NCSC website. Therefore, a data description file was generated to help understand all 15 variables within the dataset. Upon examination, it was discovered that a potentially significant attribute, the likelihood description (`Kansomschrijving`), was absent. This attribute directly influences the likelihood of a threat occurring. However, `Kansomschrijving` is not a primary focus of this investigation, as the emphasis lies on assessing the level of likelihood and the description of each threat. With the exception of the 95 missing values mentioned previously, all other data points are intact. Figure 3 shows the overall distribution of likelihood and impact levels within the NCSC dataset. Interestingly, most threats are associated with a `Medium` likelihood indicator.



Most vulnerabilities have a medium likelihood of occurrence, with an impact nearly evenly divided between medium and high.

Fig. 3: The number of `Low`, `Medium`, and `High` indicators for the `Kans` and the `Schade`.

5.2 Data Pre-processing

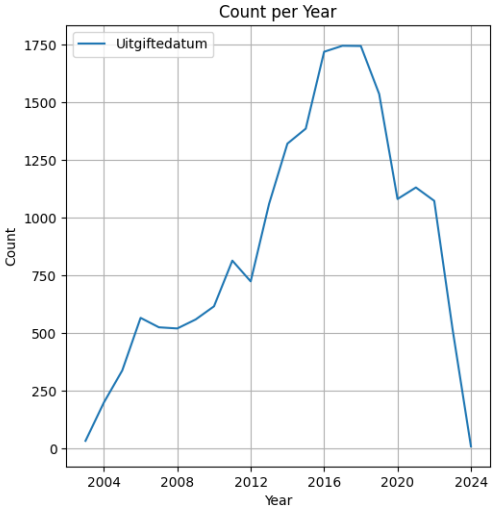
Following the initial exploration of the NCSC dataset, the focus shifts to data preprocessing to improve the accuracy and reliability of the research by cleaning the dataset. Key tasks include data selection, data cleaning, and feature engineering.

Data Selection In the beginning stages, it is significant to select only essential data to eliminate noise from the dataset. This process involves the removal of duplicate instances and the exclusion of irrelevant attributes. The column `NCSC inschaling` is removed from the data set, as it aggregates the likelihood and impact indicators in a single string. The rationale behind this exclusion is that segregating this information into distinct variables would enhance its utility. These separate components are already present within the data set, making `NCSC inschaling` redundant.

Furthermore, the column `Versie` is removed from the dataset due to its inconsistency. Some advisories use the notation 1.01 for the first updated version of the threat, while other advisories use 2.00 for a subsequent publication. Instead, the unique `NCSC ID` in combination with `Uitgiftedatum` could effectively represent this information.

ADD delete unused columns Mogelijke Oplossingen and Schadeomschrijving (+ schade + titel)

In addition, the objective of this study is to investigate changes in the likelihood classification. To facilitate this analysis, it is imperative that each threat possesses a minimum of two inputs. Consequently, all instances of NCSC IDs occurring only once are removed from the dataset as they fall outside the scope of this investigation. The line plot presented in Figure 4 illustrates that this elimination does not result in any data gaps. Across every year within the data set, a substantial number of instances remain, with most of the advisories spanning from 2015 to 2017. To develop the line plot, the data type of the column `Uitgiftedatum` was converted from a string object to a datetime format. Following the data selection process, the dataset comprises a total of $19,215$ instances and 13 attributes.



Vulnerabilities are published consistently between 2002 and 2023, with a peak in threat advisories occurring from 2015 to 2017.

Fig. 4: Line plot of `Uitgiftedatum`.

Data Cleaning The next step in the preprocessing phase is data cleaning, which involves identifying and rectifying errors and inconsistencies to enhance the quality and reliability of the data for analysis. In addition to the conversion of the `Uitgiftedatum` mentioned earlier, the data types of the attributes `CVE-ID`, `Toepassingen`, `Versies` and `Platformen` are corrected by converting them from a string representation of a list to an actual list of strings.

During this data cleaning process, it was observed that if no data points are available for the corresponding attribute, it is indicated by the string *[]*. Additionally, the strings *['niet beschikbaar']* and *['-']* frequently appear in the `CVE-ID` attribute. These instances are cleaned by imputing a $None$ value. Furthermore, inconsistencies were identified among the CVE-IDs. After cleaning these instances by removing extraneous punctuation and spaces, the CVE-IDs are validated against the correct format. A CVE-ID is correctly formatted if it is a string that starts with 3 letters, followed by a hyphen, and then contains only digits and hyphens, with a total length of either 13 or 14 characters. Instances with incorrectly formatted CVE-IDs are removed from the dataset. If the list of the CVE-IDs becomes empty, this consequently results in new missing values. Ultimately, the `CVE-ID` attribute contains $1,025$ missing values. These instances are subsequently deleted from the dataset, as the `CVE-ID` is essential to serve as the unique common identifier with the APT reference dataset. The Common Vulnerabilities and Exposures (CVE)[3] programme provides a reference method for publicly known cyber security vulnerabilities and exposures. The CVE-IDs are listed on MITRE, as well as on the American National Vulnerability Database (NVD) and Dutch NCSC platform.

ADD normalization of values of the `product`, *operating system* attributes

Outliers & Missing Values Typically, the subsequent step in pre-processing involves outlier detection and missing value imputation, as identifying and handling data points that deviate significantly from the rest of the data in the dataset is important to improve the accuracy of the data. However, the research objective is to find uncertainty within the advisory descriptions. Consequently, it would be counterproductive to impute outliers or missing values with new values, as these anomalies or missing entries could potentially signify uncertainty in the data input.

Feature Engineering The next phase involves leveraging the available dataset to generate new features. Previously, the attribute `Versie` was removed from the dataset due to inconsistencies. Given the importance of this information for the research, a new column, denoted as `Update`, is introduced. This attribute assigns an integer value to each update of every NCSC advisory in chronological order based on the corresponding `NCSC ID` and `Uitgiftedatum`.

Furthermore, two additional attributes are extracted from the `Beschrijving` column. Specifically, these features capture the word count and token count of each description, denoted as `Words` and `Tokens` respectively. The distinction between these attributes is caused by the tokenizer from the `NLTK` Python library, which also counts punctuation as separate tokens. These two attributes could be beneficial because they convert textual data into numeric representations, enabling interpretation by Machine Learning algorithms.

Classification Engineering A new dataset was developed to categorise changes in threat descriptions with respect to similar or modified likelihood classifications of threats. After sorting the NCSC data based on `NCSC ID` and `Update`, a dataframe was created with four new indicator variables, resulting in a total of $13,450$ instances and 6 attributes. In addition to the `NCSC ID` and `CVE-ID`, the four columns are defined as follows:

- `no_change` = same description & same likelihood
- `unjustified_change` = same description & change in likelihood
- `unimportant_change` = change in description & same likelihood
- `justified_important_change` = change in description & change in likelihood

This new table, named "*NCSC Update Justification*", will serve as a crucial component in subsequent temporal analyses of threats. By summing the indicator values for each attribute, it was determined that only 250 updates are justified and important based on their likelihood classification. As shown in Table 5, the majority of changes in the description either do not result in a different likelihood classification or are deemed irrelevant, suggesting that these changes did not affect the threat's likelihood of occurrence.

NCSC Update Justification	
No change	$11,507$
Unjustified change	122
Unimportant change	$2,363$
Justified & important change	250

Table 5: Total sums of the 4 indicator attributes.

[3] CVE: `https://www.cve.org/`

6 Advanced Persistent Threats

The Advanced Persistent Threats (APTs) dataset serves as a reference for this research. This manually curated database encompasses APT campaigns from 2008 to 2020 and was developed to support the research described in [3]. The dataset used in this investigation is provided in the CSV file named *campaigns_vulnerability_vector_product_version_os*[4]. The data comes from both unstructured and structured threat sources. The structured sources include MITRE Att&ck and NVD, which are both publicly available [3]. MITRE Att&ck, which stands for Advarial Tactics, Techniques, and Common Knowledge, includes category groups that classify and describe cyberattacks and intrusions. The NVD, or National Vulnerability Database, is a repository by the United States government that includes standards-based vulnerability management data [3].

6.1 Exploratory Data Analysis

An Exploratory Data Analysis (EDA) was conducted to analyse and better understand the dataset and assess its relevance to the research questions. Table 6 summarises the dataset, which contains a total of 10 attributes and $65,552$ instances. The high instance count arises because each APT is represented by multiple rows detailing different `product`, `version`, `update` and `os` information associated with the APT. For example, the $65,552$ instances contain only 118 unique IDs for the `vulnerability` attribute. While there are no duplicate entries, there are $1,463$ missing values, distributed across 7 attributes, each with exactly 209 missing data points. For these instances only the attributes `campaign`, `attack_vector`, `exploited_time` have data points available. Furthermore, the analysis includes understanding the contents of the attributes. A couple of attributes immediately stand out because they closely resemble those in the NCSC dataset, specifically:

Data Summary	
Instances	$65,552$
Attributes	10
Duplicates	-
Missing values	$1,463$

Table 6: Data summary of the original APT dataset.

`vulnerability` → `CVE-ID`
`published_time` → `Uitgiftedatum`
`product` → `Toepassingen`
`version` → `Versies`
`os` → `Platformen`

6.2 Data Pre-processing

Similar to the NCSC data, the APT dataset requires pre-processing to ensure that only relevant data is included in the analysis. This process focusses on data selection, data cleaning, missing values, and feature engineering.

Data Selection The first step in the data preprocessing phase is to select the relevant data from the APT dataset. The exploratory data analysis revealed that the first attribute, called `APT`, contains the name of the APT based on MITRE ATT&ck. This column can be dropped as the different MITRE category groups are beyond the scope of this investigation.

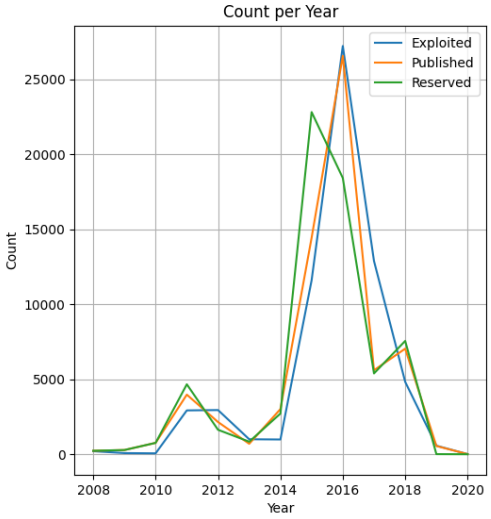
In addition, the `update` attribute is removed from the dataset due to its high proportion of missing values and its inconsistencies. The attribute contains the update number for the product affected by the CVE. When no update is available, the input value is denoted as *"*"*, representing a total of $62,801$ missing data points. Moreover, the remaining values are difficult to interpret due to errors and inconsistencies. For instance, some of the unique data entries include *"sp1", "sp2", "gold", "beta_4", "update11_b03", "update99", "r2_sp1", "unknown"*. Following the data selection process, the APT dataset comprises a total of $65,552$ instances and 9 attributes.

Data Cleaning During the data cleaning phase, attention is again focused on the data types of the attributes. The time-related columns, `exploited_time`, `reserved_time` and `published_time`, are converted from a string object to a uniform datetime format. The conversion aligns the columns with the datetime format used for the `Uitgiftedatum` attribute of the NCSC, ensuring consistency across datasets.

The conversion allows for a more accurate analysis of the time-related columns. Figure 5 shows the differences among the three columns, illustrating the temporal progression of the threat publications. The `reserved_time` attribute, representing the initial reservation

[4] APT GitHub: `https://github.com/giorgioditizio/APTs-database/blob/v1.0.1.1/data_analysis/campaigns_vulnerability_vector_product_version_os.csv`

of the CVE by MITRE, generally precedes the `published_time` attribute, indicating that threats are typically reserved by MITRE before they are officially published by the NVD. Following publication, the `exploited_time` attribute usually shows the latest dates, suggesting that exploitation of threats occurs after their publication.



Exploitation typically occurs after the vulnerability is reserved and published, indicating that most exploits happen for known vulnerabilities.

Fig. 5: Line plot of time-related columns.

This sequential pattern highlights the typical life cycle of a threat, starting with its reservation, followed by its official publication, and concluding with its exploitation. The time lag between these events is important for understanding the progression and potential response times for threat mitigation. The visualisation emphasises the importance of prompt sharing of threat intelligence and the subsequent actions taken to address potential threats.

In addition to the conversion of the data type of the time-related attributes, data cleaning is applied to the variables `version` and `os`. Similarly to the previously removed `update` attribute, the `version` and `os` columns contain entries marked with "*" when version numbers or operating system specifications are unavailable. These "*" entries, along with instances marked by "-", are cleaned by imputing a $None$ value, ensuring uniformity and accuracy in the dataset.

Outliers & Missing Values As with the NCSC dataset, the presence of outliers and missing values in the APT dataset could indicate uncertainty regarding threats. The previously mentioned 209 instances with missing data points could still represent some level of uncertainty. For example, these instances will have more uncertainty around the corresponding threats compared to other data in the APT dataset, but less uncertainty than instances that are completely absent from the dataset. However, data from the three available attributes are insufficient for this research, because the common identifier to connect the instance to the correct NCSC threat is missing. Imputing new values is also not viable as it would counterproductively affect the representation of uncertainty. Therefore, the 209 instances with missing values are deleted from the dataset.

7 Temporal Analysis of Threats

With both the NCSC and APT datasets prepared and cleaned, the primary research question can be investigated. The temporal analysis aims to analyse how the reserved and exploitation times of the APT dataset correlate with both initial and subsequent updates from the NCSC platform. Specifically, it examines whether the reserved time aligns with the initial NCSC publication and whether the exploitation time aligns with subsequent updates. These temporal analyses are essential because NCSC updates often indicate key events of threats, such as the identification or exploitation of vulnerabilities. By understanding these alignments, research can provide valuable information on the timeliness and effectiveness of threat responses, improve vulnerability management practices, and more accurately represent uncertainty in threat intelligence.

Furthermore, this study explores the influence of update justification and likelihood classification on the temporal relationships between the publication of threats on the NCSC platform and their exploitation within the APT dataset. By examining these factors, the research seeks to uncover patterns that could inform better risk assessment and resource allocation strategies in cybersecurity. Representing uncertainty in threat intelligence is crucial to providing more accurate and actionable insights to decision makers.

To address the first research question, the study employs a comprehensive methodology that includes temporal analyses, statistical analyses, and the examination of temporal differences to identify significant relations. The process begins with merging the NCSC, NCSC Update Justification,

and APT datasets, all of which are linked by the common unique identifier, `CVE-ID`.

To ensure that each `CVE-ID` is treated as a distinct instance, the NCSC datasets are expanded based on this attribute. Of the 118 unique CVE-IDs in the APT dataset, the merged dataset comprises a total of 86 unique CVE-IDs. Given the focus of the research on temporal data, the `Uitgiftedatum` of the NCSC datasets are standardised to the first day of the month. This ensures consistent granularity with the APT dataset, as for the APT dataset all dates of the three time columns are set to the first day of the month.

The final merged dataset forms the basis for subsequent analyses, which include calculating time differences between publication dates, performing correlation analyses to identify potential relationships, and conducting statistical tests to explore the connections between time-based metrics, update justifications, and likelihood classifications.

7.1 Reserved Time to Initial Release

The first temporal analysis examines the alignment between the `reserved_time` attribute from the APT dataset and the `Uitgiftedatum` of the initial NCSC release of each instance. To ensure that the analysis focusses solely on initial updates, the newly engineered `Update` column is set to 1.

The analysis involves calculating the Pearson correlation between the `reserved_time` and the `Uitgiftedatum`, including its p-value to illustrate significance. The Pearson correlation coefficient provides a measure of the strength and direction of the linear relationship between the two time columns. A strong correlation would suggest that the initial publication of a threat on the NCSC platform closely follows its reservation time in the APT dataset, indicating the timely sharing of threat information.

Additionally, the Mean Absolute Error (MAE) is calculated to measure the average difference between the reserved times and the initial publication dates. The MAE is reported in rounded days and provides a straightforward interpretation of the total average deviation between the two temporal points. A lower MAE indicates better alignment and less delay between reservation and initial publication. Again, reflecting efficient communication of threat information.

7.2 Exploited Time to Subsequent Releases

The second temporal analysis investigates the alignment between the `exploited_time` attribute from the APT dataset and the `Uitgiftedatum` of all subsequent NCSC updates. Here, the `Update` column is set to greater than 1 to select only subsequent NCSC updates.

The second analysis also calculates the Pearson correlation and MAE. A strong correlation would indicate that subsequent updates on the NCSC platform are closely related to the actual exploitation events recorded in the APT dataset, suggesting responsive and timely updates following exploitation. A lower MAE would reflect more prompt updates, indicating effective communication and reduced uncertainty about threat exploitation.

7.3 Update Justification on Timing

The final three temporal analyses focus on the time differences between the `Uitgiftedatum` of the NCSC and the `exploited_time` from the APT dataset. Using these time differences, the relationship between the justified and non-justified NCSC updates can be investigated. Justified updates, as detailed in Table 5, include updates of which the changes in the description are justified and important. Unjustified updates include updates with no changes, unjustified changes, and unimportant changes in the description.

For this analysis, the Pearson correlation and the T-statistic are calculated, including the p-value for both. The Pearson correlation coefficient measures the linear relationship between the time differences and the update justifications. The T-test assesses whether the mean time difference for justified updates differs significantly from that of unjustified updates.

By examining the two metrics, the analysis aims to determine whether justified updates are more timely and responsive compared to unjustified updates. Therefore, the results could offer information on the effectiveness and reliability of threat communication.

7.4 Likelihood on Timing

The final temporal analysis investigates the relationships between the three likelihood classifications (`Low`, `Medium`, and `High`) with the time difference between the publication by the NCSC and the exploitation time by the APT.

This analysis involves calculating the Pearson correlation and the ANOVA F-statistic. The Pearson correlation coefficient quantifies the strength of the linear relationship between the time difference and the likelihood classifications. Unlike the t-test, which is limited to comparing two

groups, the ANOVA F-test evaluates the variance between multiple groups. Thus, the F-statistic can measure whether there are significant differences in the mean time difference between the three likelihood classifications.

Through this analysis, the research aims to uncover how likelihood classifications influence the timeliness of threat exploitation. Understanding these impacts provides valuable information on vulnerability prioritisation and management, contributing to more effective threat response strategies. In addition, identifying patterns and correlations between likelihood classifications and exploitation timeliness helps to represent and quantify the inherent uncertainty in threat intelligence. It informs risk assessments based on the predicted likelihood and timing of threat exploitation.

8 Ontological Analysis of Threats

In addition to the temporal analysis, this research includes an ontological analysis to examine the alignment and consistency of the two national threat intelligence datasets. The ontological analysis aims to address the second research question, which focusses on the technical details of the vulnerabilities. The ontological analysis employs the URREF ontology to systematically evaluate and compare the technical attributes of vulnerabilities recorded in the NCSC and APT datasets. The literature findings in Section 3 highlighted the URREF ontology as an efficient framework for representing uncertainty in threat intelligence. The use of an ontology, specifically the URREF, is crucial for this research for several reasons:

- **Structured Data Mapping**: The ontology provides a common framework for effectively aligning technical details, such as product, operating system, and version details, across the NCSC and APT datasets.

- **Consistency Evaluation**: The ontology helps identify and report discrepancies in the data, ensuring a more accurate comparison and enabling the representation of threat intelligence.

- **Reusability & Communal Coherence**: Extending an established ontology ensures uniformaty of concepts, relationships, and terminology, which enhances productivity and comprehension by researchers and stakeholders alike.

8.1 URREF Ontology

The core of this ontological analysis involves accurately mapping threat intelligence data from the NCSC and APT datasets to the URREF ontology. This process begins by downloading the URREF ontology, provided in `.owl` format, from its official GitHub repository[5]. To leverage the Resource Description Framework (RDF) Schema for ontology description, the URREF ontology is converted from its original OWL/XML syntax to RDF/XML syntax using `Protégé`, an open-source ontology editor. The RDF Schema provides a standard language, based on the RDF Knowledge Representation Data Model, to define relationships between different data points, ensuring consistency and allowing semantic interoperability[6].

The RDF schema developed for this research outlines the structure of the URREF ontology, detailing how various classes and properties interconnect. RDF is chosen for its effectiveness in describing resources and their relationships using triples consisting of a subject, a predicate, and an object, making it particularly suitable for representing complex interrelations within threat intelligence data.

8.2 Classes & Properties

The subsequent phase of the ontological analysis involves the definition and creation of classes and properties within the URREF ontology. These components are essential for structuring the ontology, enabling it to accurately represent the complexities of threat intelligence data. This process includes defining several key classes, as described in Table 7. Each class represents a core concept within the domain of threat intelligence. In conjunction with these classes, properties are established to articulate the relationships between them, as detailed in Table 8.

ADD ontology graph

8.3 Ontology Population

After the development of the applicable classes and properties, the ontology is populated using the merged NCSC and APT datasets, processed incrementally in batches to ensure efficient handling and updating. For each data instance, a unique URI was generated based on the CVE-ID. This approach allows each vulnerability to be distinctly identified within the ontology framework.

To distinguish the origin of data, each instance is linked to its respective dataset, either NCSC or APT, through the creation and assignment of dataset-specific URIs. This association is critical for the consistency and reliability analysis between the two dataset.

The likelihood of each vulnerability is classified as `Low`, `Medium`, or `High`. These classifications are represented as instances and linked directly to the corresponding vulnerabilities. Additionally, the justification for any adjustments in the likelihood assessment resulting from description updates is recorded, ensuring that justified and non-justified changes are documented within the ontology.

[5] URREF GitHub: `https://github.com/adelphi23/urref/blob/master/URREF.owl`
[6] RDF Schema: `https://www.w3.org/TR/rdf-schema/`

Classes	Description
`Threat Intelligence`	Serves as the top-level class, containing all aspects of threat intelligence data.
`Vulnerability`	Represents individual vulnerabilities identified using the CVE-IDs employed.
`Product`	Captures the specific software or hardware product linked to the identified vulnerability.
`Version`	Captures the particular version number of the product affected by the vulnerability.
`OS`	Captures the operating system on which the affected product version is capable of running.
`Dataset`	Distinguishes between the sources of data, namely the NCSC and APT datasets.
`Likelihood`	Classifies the probability of the vulnerability being exploited, categorized as `Low`, `Medium` or `High`.
`Justification`	Represents whether a change in likelihood is justified or not by an update of the vulnerabilities's description.

Table 7: Ontology classes and their descriptions.

Properties	Description
`hasCVEID`	Links a vulnerability to its unique identifier (CVE-ID).
`fromDataset`	Indicates the dataset from which a particular instance originates.
`affectsProduct`	Describes the relationships between a vulnerability and the affected product.
`affectsVersion`	Describes the relationships between the affected product and the affected version of the vulnerability.
`runsOn`	Describes the relationships between the affected version and the operating system of the vulnerability.
`hasLikelihood`	Connects vulnerabilities to their assessed likelihood classification.
`hasDescriptionChange`	Captures whether the update in the description of the vulnerability is justified by a change in likelihood.

Table 8: Ontology properties and their descriptions.

The ontology also includes the detailed technical attributes related to each vulnerability, including the affected product, version, and operating system. These attributes are represented as instances of the `Product`, `Version`, and `OS` classes and are linked back to the vulnerabilities they pertain to, ensuring a comprehensive representation of each vulnerability's technical context.

Following the incremental processing of each data batch, the ontology graph is serialised and stored in RDF/XML format. This systematic approach ensures a comprehensive and structured representation of the technical details derived from both the NCSC and APT datasets. The resulting ontology serves as the foundation for subsequent execution of SPARQL queries, allowing an experimental analysis of cybersecurity threats.

8.4 SPARQL Queries

The ontological framework populated from the NCSC and APT datasets is subjected to a series of SPARQL queries executed within the `GraphDB` environment. These queries are designed to investigate various aspects of the data, facilitating a detailed comparative analysis across different dimensions of vulnerabilities, products, and operating systems. Figure 6 shows an example of a query that counts the number of unique vulnerabilities in the merged dataset. The queries for the ontological analysis can be found in Appendix A. The experiments are structured in five sub-analyses.

```
PREFIX TI:<http://example.org/
          threatintelligence/>

SELECT (COUNT(*) as ?count)
WHERE {
    ?vulnerability TI:hasCVEID ?CVEid
}
```

Fig. 6: SPARQL query example.

1. Vulnerability Analysis: aims to identify and compare unique vulnerabilities in NCSC and APT datasets. The SPARQL queries in this analysis focus on evaluating the alignment and divergence of vulnerabilities, particularly in relation to affected products, operating systems, and version details.

- 1.1 Unique vulnerabilities with **any** product in common between NCSC and APT.
- 1.2 Unique vulnerabilities with **all** products identical for NCSC and APT.
- 1.3 Unique vulnerabilities with **no** NCSC products present in APT.
- 1.4 Unique vulnerabilities with **no** APT products present in NCSC.
- 1.5 Unique vulnerabilities with **all** products and **any** operating system in common between NCSC and APT.
- 1.6 Unique vulnerabilities with **all** products and **all** operating systems identical for NCSC and APT.

1.7 Unique vulnerabilities with **all** products and **any** version in common between NCSC and APT.
1.8 Unique vulnerabilities with **all** products and **all** versions identical for NCSC and APT.

2. Product Analysis: investigates the similarities and differences between the NCSC and APT datasets with respect to the products affected by the identified vulnerabilities. This examination aims to clarify the degree of alignment in product focus between the two datasets.

2.1 What products do the NCSC and APT share most frequently?
2.2 Do NCSC and APT prioritise the same products, or are there differences?
2.3 What products are exclusively listed in the NCSC dataset?
2.4 What products are exclusively listed in the APT dataset?

3. Operating System Analysis: focusses on the operating systems associated with vulnerabilities in the NCSC and APT datasets, aiming to identify areas of alignment or divergence.

3.1 What operating systems do the NCSC and APT share most frequently?
3.2 Do NCSC and APT prioritise the same operating systems, or are there differences?
3.3 What operating systems are exclusively listed in the NCSC dataset?
3.4 What operating systems are exclusively listed in the APT dataset?

4. Update Justification Analysis

4.1 How many vulnerabilities with description changes in NCSC, are available in the APT dataset?
4.2 Absolute (and relative) Risk Reduction: change - no change NCSC vs present - not present APT
4.3 Absolute (and relative) risk reduction: justified - non-justified change NCSC vs present - not present APT
4.4 ARR: justified - non-justified change NCSC vs popular non-popular APT

5. Likelihood Analysis

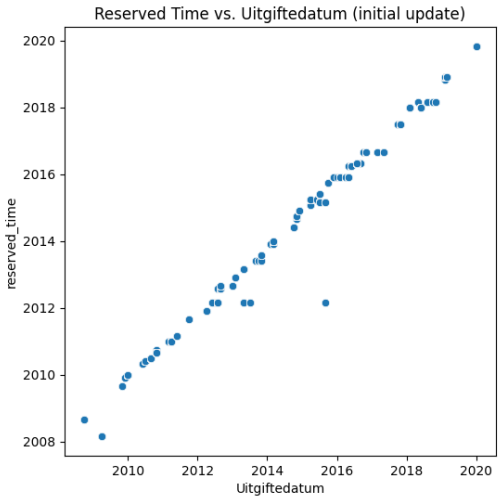
5.1 How many vulnerabilities where the NCSC has a HIGH likelihood of occurring are APT related? > do again ARR

9 Results Temporal Analyses

The temporal analysis comprises four distinct investigations, each yielding specific outcomes. These results provide insights into how the reserved and exploitation times of threats align with the initial NCSC publication updates and the subsequent NCSC updates, respectively. Furthermore, the findings show whether the justification and likelihood classifications of vulnerabilities impact the temporal differences between the publication times of the APT and NCSC.

9.1 Reserved Time to Initial Release

The Pearson correlation between the APT reserved time and the initial NCSC release time is extremely high, with a coefficient of 0.98 and a p-value smaller than 0.001. This suggests a very strong positive linear and significant relationship between the two attributes. The scatter plot in Figure 7 visually confirms this strong positive relationship. A correlation coefficient close to 1 indicates that the reserved and publication times are highly dependent on each other. This implies that the reserved time for the APT is closely tied to the initial publication of a threat on the NCSC platform.



A strong positive relationship exists between the APT reservation and NCSC publication of vulnerabilities, with only a few outliers.

Fig. 7: Scatter plot `reserved_time` to `Uitgiftedatum`.

The Mean Absolute Error (MAE) between the APT reserved time and the initial NCSC publication time is 76 days. This metric indicates that, on average, the model's predictions for the initial NCSC publication deviate from the APT reserved time by approximately 76 days. Both results, as detailed in Table 9, suggest a close alignment between the reserved APT time and the initial publication time of the same vulnerability on the NCSC platform.

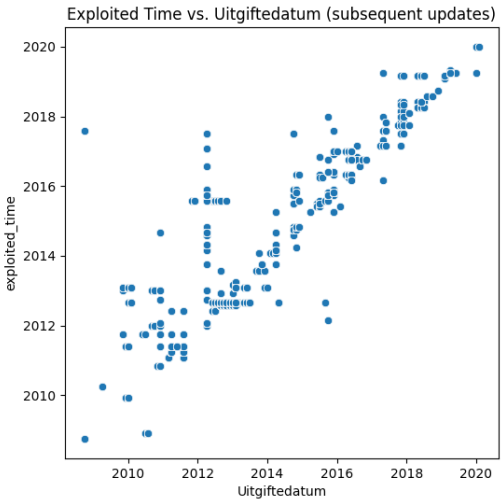
Reserved Time	
Pearson correlation	0.982^{***}
Mean Average Error	76 days

$^{***}(p < 0.001)$

Table 9: Results of APT reserved time to NCSC initial publication.

9.2 Exploited Time to Subsequent Releases

The Pearson correlation between the APT exploitation time and the publication time of subsequent NCSC updates is notably high, with a coefficient of 0.94 and a p-value of < 0.001. This strong positive linear relationship suggests a significant dependence between the two variables, although not as strong as the correlation observed for the analysis on the reserved time to initial updates. The relationship is illustrated in the scatter plot in Figure 8, where the data points are spread more widely, indicating some variability.



A moderate positive relationship exists between APT exploitation and NCSC republication, with some variability and a couple outliers.

Fig. 8: Scatter plot `exploited_time` to `Uitgiftedatum`.

Despite the slightly lower correlation compared to the previous analysis of the initial updates, the correlation remains significant, implying a notable dependence between the exploitation by the APT and subsequent updates by the NCSC. The lower correlation coefficient could indicate that the relationship between the APT exploitation time and the NCSC publication time becomes more complex or nuanced as more updates on the same threat are published.

The MAE between the APT exploitation time and the publication time of subsequent NCSC updates is 123 days. This means that, on average, the model predictions for the publication time of subsequent updates by the NCSC are off by about 123 days from the APT exploited times. The larger error, compared to the previous analysis, aligns with the lower correlation coefficient. A less strong relationship generally results in a higher prediction error, reflecting the increased complexity and variability in the timing of subsequent threat updates.

Exploited Time	
Pearson correlation	0.938^{***}
Mean Average Error	123 days

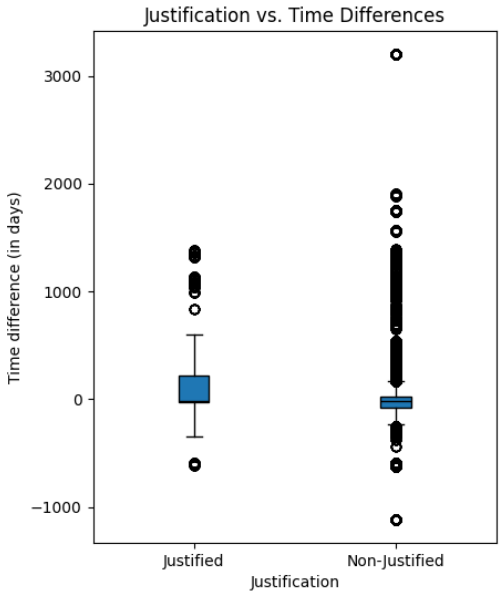
$^{***}(p < 0.001)$

Table 10: Results of APT exploited time to NCSC subsequent publications.

9.3 Update Justification on Timing

The results of the third temporal analysis examine the impact of justified versus unjustified changes in the description of threat updates on the time difference between the publication times by the APT and the NCSC. The Pearson correlation coefficient of 0.04 indicates a very weak to negligible linear relationship between the justification of threat changes and the difference in publication time. However, the p-value of < 0.001 indicates that the correlation, although weak, is statistically significant. This suggests that the classification of the justification of updates does not substantially influence the publication timing of updates.

A more detailed view of the time differences between justified and unjustified updates can be observed in the box plot shown in Figure 9. The box plot reveals that the interquartile range (IQR) of justified threat publications is larger compared to those that are not justified, indicating a wider spread of time differences. Furthermore, the `Justified` group has fewer outliers, suggesting that time differences are more consistently distributed around the median. In contrast, the `Non-Justified` group has a very small IQR with many outliers, which indicates a higher concentration of time differences within a narrow range, but with frequent extreme values.



The distribution of vulnerabilities with non-justified changes exhibits greater dispersion and more outliers than justified updates.

Fig. 9: Box plot of justification.

Further supporting these observations, Table **??** summarises the key measures of this analysis. The T-statistic of 86.04 with a p-value of < 0.001 indicates a substantial significant difference in the mean exploitation times between threats with justified and unjustified changes in the description. This statistical test highlights that the mean time differences are not the same between the two groups. This is supported by the box plot in Figure 9, where justified updates have a less extreme distribution compared to unjustified updates.

	Justification
Pearson correlation	0.042^{***}
T-statistic	86.04^{***}

$^{***}(p < 0.001)$

Table 11: Results of justification on time differences.

9.4 Likelihood on Timing

The final temporal analysis investigates the relationship between the likelihood classifications (`Low`, `Medium`, and `High`) and the time differences between the NCSC publication dates and the APT exploitation times. Table 12 shows that the Pearson correlation coefficient is -0.43, with a p-value of < 0.001. The negative correlation coefficient suggests a moderate inverse relationship between likelihood and time differences, indicating that as the likelihood of a threat increases, the time difference between the NCSC publication and the APT exploitation decreases. The extremely low p-value confirms that this correlation is statistically significant.

	Likelihood
Pearson correlation	-0.427^{***}
F-statistic	232500^{***}

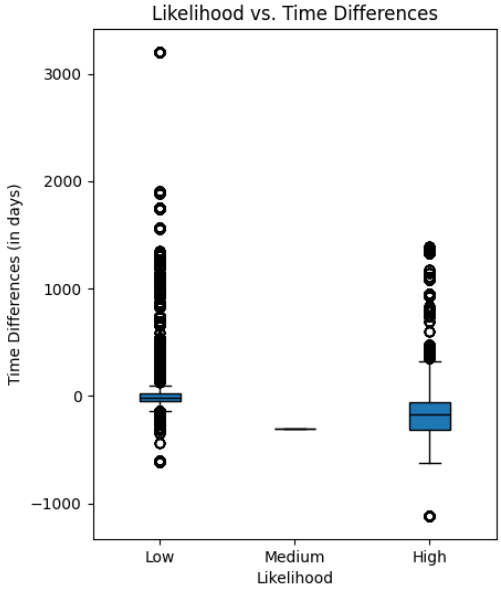
$^{***}(p < 0.001)$

Table 12: Results of likelihood on time differences.

In addition, an ANOVA test was performed to compare the means of time differences in the three likelihood classifications. The ANOVA F-statistic is 232500.28, with a p-value of < 0.001. The high F-statistic and the very low p-value indicate that there are significant differences in the

mean time differences among the `Low`, `Medium`, and `High` likelihood groups. This suggests that the likelihood classification is a meaningful factor in explaining the variability in time differences between when threats are published by the NCSC versus the APT.

The box plot in Figure 10 illustrates a visual representation of these findings by plotting the time differences for each likelihood group. The `Low` likelihood group exhibits numerous outliers, indicating a wide range of time differences. In contrast, the `Medium` likelihood group shows no outliers, suggesting a very consistent publication of threats. The `High` likelihood group has some outliers, reflecting a generally faster exploitation time with a few exceptions.

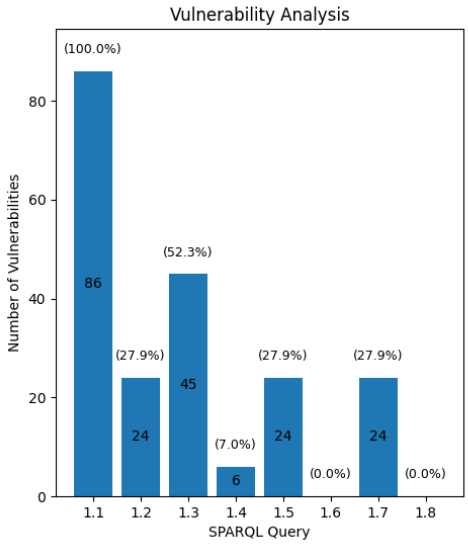


The distribution of vulnerabilities with low likelihood shows high dispersion and many outliers, whereas medium likelihood is nearly uniform.

Fig. 10: Box plot of likelihood.

10 Results Ontological Queries

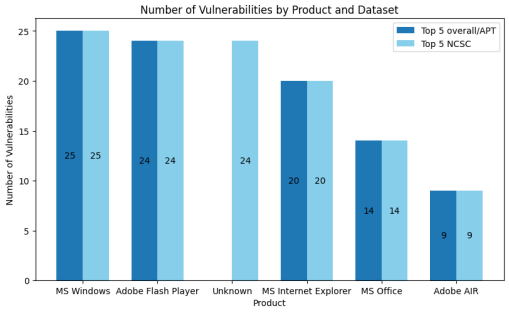
10.1 Vulnerability Analysis



...

Fig. 11: Bar chart results vulnerability analysis.

10.2 Product Analysis



...

Fig. 12: Bar chart results product analysis.

10.3 Operating System Analysis

10.4 Update Justification Analysis

10.5 Likelihood Analysis

10.6 Impact Analysis

11 Discussion

11.1 Answers to Research Questions

11.2 Threats to Validity

Literature Review I could have adjusted the keyword-based searches on the predefined selection criteria and sub-criteria to obtain even more relevant searches. However, since the topic was fairly new to me, I had to read multiple articles first to be able to decide on good selection criteria. On the other hand, I could have invented a better refined search taking into account the criteria.

Research Question My initial idea for the research question and the research set-up was way too extended. I know understand that it is important for a thesis to focus on one main aspect and really delve into that.

11.3 Web Scraper

I created a web scraper to obtain the data of the `chance description`. However, only after already developing the code did I realise that I was not able to retrieve all the necessary data. Therefore, my web scraper was useless and caused me to be inefficient with my time. On the other hand, I refreshed my memory of how to scrape data from a website.

As noted in Section 5.1 as part of the Exploratory Data Analysis, the dataset lacks the specific column with respect to the information on the `chance`. Since a change in the `chance description` could possibly influence the `chance` for a threat to occur, acquiring the data of the `chance description` could have significant value. Consequently, the idea was to develop a web scraper mechanism to retrieve these missing data from the NCSC security advisories website.

Unfortunately, the format of the pages that display each unique security advisory restricts access only to the `chance description` associated with the most recent update of each NCSC advisory. This limits the ability to scrape data from previous updates. Efforts to use tools such as the Wayback Machine and Google Web Cache Viewer, typically used to access archived web content, also were unsuccessful in retrieving any previous `chance descriptions`. This inability arises from the lack of archival records for the individual pages of NCSC advisories. Therefore, the `chance description` linked to former updates sharing the same NCSC ID is impossible.

11.4 Natural Language Processing

I developed two distinct Natural Language Processing (NLP) algorithms, each using different Python libraries: spaCy and NLTK. The objective was to normalise the unstructured textual data from the `Beschrijving` attribute to extract relevant information for the ontology model. The normalisation procedure included lowercasing, special character removal, stopword elimination, tokenization, stemming, lemmatisation, Part-of-Speech (POS) tagging, Named Entity Recognition (NER), and Bag-of-Words representation.

However, it emerged that both libraries failed to achieve satisfactory accuracy levels. This inadequacy was mainly due to the Dutch language of the `Beschrijving` texts. Unfortunately, most language models are only optimal for English texts. For example, in the case of Named Entity Recognition, the operating system "Windows" was categorised both as a person, geo-political entity, and an organisation. Therefore, implementing NLP techniques would increase errors, consequently amplifying uncertainty. Given the aim of this research, making extensive alternations of the data is restricted to preserve the inherent uncertainty.

11.5 URREF Ontology Usage

URREF ontology chosen to be used in the research by extending it. I explain that its reusability can be a good reason, but that is only the case if URREF already covers a significant portion of the domain working in. This was not fully the case. The other reasons (literature findings + community consistency) do stand.

11.6 Future Work

RQ Description	Results

Table 13: Caption

12 Conclusion

Appendix

A Classes & Properties Ontology

B SPARQL Queries

```
PREFIX TI: <http://example.org/threatintelligence/>
PREFIX RDF: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX RDFS: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (COUNT(DISTINCT ?vulnerability) AS ?count)
WHERE {
    ?vulnerability RDF:type TI:Vulnerability .

    ?vulnerability TI:affectsProduct ?aptProduct .
    ?aptProduct TI:fromDataset ?aptDataset .
    ?aptDataset RDF:type TI:Dataset ;
                RDFS:label "APT" .

    ?vulnerability TI:affectsProduct ?ncscProduct .
    ?ncscProduct TI:fromDataset ?ncscDataset .
    ?ncscDataset RDF:type TI:Dataset ;
                RDFS:label "NCSC" .

FILTER(?aptProduct = ?ncscProduct)
}
```

Query 1.1: Total unique vulnerabilities where any product is the same for NCSC and APT.

```
PREFIX TI: <http://example.org/threatintelligence/>
PREFIX RDF: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX RDFS: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (COUNT(DISTINCT ?vulnerability) AS ?count)
WHERE {
  ?vulnerability RDF:type TI:Vulnerability .
  ?vulnerability TI:affectsProduct ?aptProduct .
  ?aptProduct TI:fromDataset ?aptDataset .
  ?aptDataset RDF:type TI:Dataset ;
              RDFS:label "APT" .
  ?vulnerability TI:affectsProduct ?ncscProduct .
  ?ncscProduct TI:fromDataset ?ncscDataset .
  ?ncscDataset RDF:type TI:Dataset ;
              RDFS:label "NCSC" .
  FILTER NOT EXISTS {
    ?vulnerability TI:affectsProduct ?product .
    ?product TI:fromDataset ?dataset .
    ?dataset RDF:type TI:Dataset ;
           RDFS:label ?datasetLabel .
    FILTER (?datasetLabel IN ("APT", "NCSC") &&
            (EXISTS { ?vulnerability TI:affectsProduct ?otherProduct .
                      ?otherProduct TI:fromDataset ?otherDataset .
                      ?otherDataset RDF:type TI:Dataset ;
                             RDFS:label ?otherDatasetLabel .
                      FILTER (?otherDatasetLabel = ?datasetLabel &&
                                ?product != ?otherProduct) }
             ||
             !sameTerm(?product, ?aptProduct) && ?datasetLabel = "APT"
             ||
             !sameTerm(?product, ?ncscProduct) && ?datasetLabel = "NCSC"))
  }
}
```

Query 1.2: Total unique vulnerabilities where all products are the same for NCSC and APT.

```
PREFIX TI: <http://example.org/threatintelligence/>
PREFIX RDF: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX RDFS: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (COUNT(DISTINCT ?vulnerability) AS ?count)
WHERE {
  ?vulnerability RDF:type TI:Vulnerability .
  ?vulnerability TI:affectsProduct ?ncscProduct .
  ?ncscProduct TI:fromDataset ?ncscDataset .
  ?ncscDataset RDF:type TI:Dataset ;
              RDFS:label "NCSC" .
  FILTER NOT EXISTS {
    ?ncscProduct TI:fromDataset ?aptDataset .
    ?aptDataset RDF:type TI:Dataset ;
           RDFS:label "APT" .
  }
}
```

Query 1.3: Total unique vulnerabilities where no NCSC products are in APT.

```
PREFIX TI: <http://example.org/threatintelligence/>
PREFIX RDF: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX RDFS: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (COUNT(DISTINCT ?vulnerability) AS ?count)
WHERE {
  ?vulnerability RDF:type TI:Vulnerability .
  ?vulnerability TI:affectsProduct ?aptProduct .
  ?aptProduct TI:fromDataset ?aptDataset .
  ?aptDataset RDF:type TI:Dataset ;
              RDFS:label "APT" .
  FILTER NOT EXISTS {
    ?aptProduct TI:fromDataset ?ncscDataset .
    ?ncscDataset RDF:type TI:Dataset ;
           RDFS:label "NCSC" .
  }
}
```

Query 1.4: Total unique vulnerabilities where no APT products are in NCSC.

```
PREFIX TI: <http://example.org/threatintelligence/>
PREFIX RDF: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX RDFS: <http://www.w3.org/2000/01/rdf-schema#>

SELECT (COUNT(DISTINCT ?vulnerability) AS ?count)
WHERE {
  ?vulnerability RDF:type TI:Vulnerability .
  ?vulnerability TI:affectsProduct ?aptProduct .
  ?aptProduct TI:fromDataset ?aptDataset .
  ?aptDataset RDF:type TI:Dataset ;
              RDFS:label "APT" .
  FILTER NOT EXISTS {
    ?aptProduct TI:fromDataset ?ncscDataset .
    ?ncscDataset RDF:type TI:Dataset ;
           RDFS:label "NCSC" .
  }
}
```

Query 1.5: Total unique vulnerabilities where no APT products are in NCSC.

References

1. E. Casey, S. Barnum, R. Griffith, J. Snyder, H. van Beek, and A. Nelson. Advancing coordinated cyber-investigations and tool interoperability using a community developed specification language. *Digital Investigation*, 22:14–45, 2017.
2. Jeonghun Cha, Sushil Kumar Singh, Yi Pan, and Jong Hyuk Park. Blockchain-based cyber threat intelligence system architecture for sustainable computing. *Sustainability*, 12(16):6401, 2020.
3. Giorgio Di Tizio, Michele Armellini, and Fabio Massacci. Software updates strategies: a quantitative evaluation against advanced persistent threats. *IEEE Transactions on Software Engineering*, 49(3):1359–1373, March 2023. arXiv:2205.07759 [cs].
4. V. Dragos, J. Ziegler, J.P de Villiers, A. de Waal, A-L. Jousselme, and E. Blasch. Entropy-based metrics for urref criteria to assess uncertainty in bayesian networks for cyber threat detection. In *2019 22th International Conference on Information Fusion (FUSION)*, page 1–8, Jul 2019.
5. Valentina Dragos, Jürgen Ziegler, and Johan Pieter De Villiers. Application of urref criteria to assess knowledge representation in cyber threat models. In *2018 21st International Conference on Information Fusion (FUSION)*, page 664–671, Jul 2018.
6. J. Freudiger, S. Rane, A.E. Brito, and E. Uzun. Privacy preserving data quality assessment for high-fidelity data sharing. volume 2014-November, page 21–29, 2014.
7. Geoff Gross, Rakesh Nagi, and Kedar Sambhoos. Situation assessment: Uncertainty representation in inexact graph matching. In *IIE Annual Conference. Proceedings*, page 1. Institute of Industrial and Systems Engineers (IISE), 2010.
8. Geoff Gross, Rakesh Nagi, and Kedar Sambhoos. Soft information, dirty graphs and uncertainty representation/processing for situation understanding. In *2010 13th International Conference on Information Fusion*, page 1–8. IEEE, 2010.
9. Massimo Guarascio, Nunziato Cassavia, Francesco Sergio Pisani, and Giuseppe Manco. Boosting cyber-threat intelligence via collaborative intrusion detection. *Future Generation Computer Systems*, 135:30–43, Oct 2022.
10. Carlos C. Insaurralde, Erik P. Blasch, Paulo CG Costa, and Krishna Sampigethaya. Uncertainty-driven ontology for decision support system in air transport. *Electronics*, 11(3):362, 2022.
11. Carlos C. Insaurralde, Paulo CG Costa, Erik Blasch, and Krishna Sampigethaya. Uncertainty considerations for ontological decision-making support in avionics analytics. In *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, page 1–9. IEEE, 2018.
12. Philipp Kuehn, Markus Bayer, Marc Wendelborn, and Christian Reuter. Ovana: An approach to analyze and improve the information quality of vulnerability databases. In *Proceedings of the 16th International Conference on Availability, Reliability and Security*, ARES 21, page 1–11, New York, NY, USA, Aug 2021. Association for Computing Machinery.
13. Claire Laudy and Valentina Dragos. Use cases for social data analysis with urref criteria. In *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, page 1–8. IEEE, 2020.
14. H. Liu, C. Zhong, A. Alnusair, and S.R. Islam. Faixid: A framework for enhancing ai explainability of intrusion detection results using data cleaning techniques. *Journal of Network and Systems Management*, 29(4), 2021.
15. Francesco Minna and Fabio Massacci. Sok: Run-time security for cloud microservices. are we there yet? *Computers & Security*, 127:103119, Apr 2023.
16. Barry E. Mullins. *The Integration of Artificial Intelligence Techniques to Improve the Effectiveness of Electronic Countermeasure Strategies in a Tactical Environment*. 1987.
17. Mattias Oredsson. *Bridging the gap between information security risk assessments and enterprise risk management*. Master's thesis, University of Stavanger, Norway, 2018.
18. Thomas Schaberreiter, Veronika Kupfersberger, Konstantinos Rantos, Arnolnt Spyros, Alexandros Papanikolaou, Christos Ilioudis, and Gerald Quirchmayr. A quantitative evaluation of trust in the quality of cyber threat intelligence sources. In *Proceedings of the 14th International Conference on Availability, Reliability and Security*, ARES '19, page 1–10, New York, NY, USA, Aug 2019. Association for Computing Machinery.
19. Daniel Schatz, Rabih Bashroush, and Julie Wall. Towards a more representative definition of cyber security. *The Journal of Digital Forensics, Security and Law*, 2017.
20. L.F. Sikos. Handling uncertainty and vagueness in network knowledge representation for cyberthreat intelligence. volume 2018-July, 2018.
21. Kilian Vasnier, A. I. Mouaddib, S. Gatepaille, and S. Brunessaux. Multi-level information fusion and active perception framework: towards a military application. *Procedings of NATO SET-262 RSM on Artificial Intelligence for Military Multisensor Fusion Engines*, 2018.