CTI-Fetcher: A Framework for Continuous Cyber Threat Intelligence Ingestion, Normalisation, and Enrichment for Continuous Cyber Risk Management

Ahmed Amro, Lama Amro, and Sarang Shaikh

Department of Information Security and Communication Technology (IIK), Norwegian University of Science and Technology (NTNU), Gjøvik, Norway

Abstract. Cyber threat intelligence (CTI) repositories provide continuously evolving knowledge about adversary techniques, tactics, mitigations, assets, and related security information. However, exploiting heterogeneous CTI sources in cyber risk management is challenging due to differences in their structures, semantics, relationships, completeness, and evolution over time. This paper presents CTI-Fetcher, a framework for systematically ingesting, normalising, enriching, and exporting CTI into a structured knowledge base for downstream cybersecurity analysis. CTI-Fetcher integrates heterogeneous CTI and security-requirement sources, transforms source-specific information into a common relational scheme, preserves relevant relationships and provenance, and selectively enriches information gaps required for risk analysis. The framework is evaluated through four CTI snapshots collected between December 2024 and June 2026, examining the evolution of the retrieved knowledge, resilience to upstream source changes, and reuse of the generated datasets. The results show that CTI evolution involves not only new threats and techniques but also substantial changes in their contextual relationships. Two confirmed upstream structural changes required adaptations to the source-specific ingestion logic, while the downstream data scheme remained stable. The resulting datasets were further consumed by ThreatSpider for CTI-driven threat modelling and risk assessment and by a separate knowledge-graph application for graph-based exploration. These results demonstrate the role of CTI-Fetcher as a reusable CTI engineering layer that decouples evolving intelligence sources from downstream cybersecurity analysis while providing structured and updatable knowledge for cyber risk management.

Keywords: Cyber Threat Intelligence, Cyber Risk Management, Cybersecurity

1 Introduction

Cybersecurity risk management depends on an accurate and up-to-date understanding of the threat landscape. However, this landscape evolves continuously

as new vulnerabilities, adversarial techniques, tactics, and defensive measures are identified. Maintaining current threat models therefore requires continuously monitoring and incorporating information from a wide range of Cyber Threat Intelligence (CTI) sources.

Public knowledge bases include MITRE ATT&CK [17,10] for enterprise, mobile, and Industrial Control Systems (ICS); ATLAS [9] for Artificial Intelligence and Machine Learning (AI/ML) systems; EMB3D [11] for embedded devices; and CAPEC [8] for mostly attack patterns against software. Together with vulnerability and cybersecurity requirement sources, they provide a foundation for automating threat modelling and risk management processes.

Operationalising this information for dynamic threat and risk analysis processes, however, is challenging. CTI sources differ in their scope, semantics, data models, and distribution mechanisms, with information made available through APIs, repositories, and formats such as JSON, CSV, YAML, and STIX [6,14]. Furthermore, the relationships required for risk analysis are not represented consistently across sources. For example, some sources explicitly relate adversarial techniques to tactics, affected assets, or mitigations, while others omit such relationships or represent conceptually similar information differently [4]. The sources also evolve independently, including both changes to their content and, in some cases, structural changes that can disrupt automated processing pipelines.

These challenges become particularly relevant when CTI is intended to support repeatable or continuous cybersecurity risk management. Rather than requiring each threat modelling or risk assessment application to independently retrieve and interpret heterogeneous CTI sources, an intermediate capability is needed to transform this information into a consistent representation that can be consumed by downstream processes. Such a capability should not only collect current CTI, but also normalise differences between sources, preserve and integrate relevant relationships, enrich information that is missing for the intended analysis, and provide structured outputs that can be reused independently of the original sources.

This paper contributes the design and evaluation of CTI-Fetcher as a reusable CTI preparation layer. It collects heterogeneous CTI, normalises and integrates source-specific entities and relationships, enriches selected gaps with provenance, and exports interconnected datasets. Evaluation combines periodic executions against evolving sources with downstream use in ThreatSpider for threat modelling and risk assessment and in a knowledge graph exploration tool for relationship analysis.

The remainder of this paper is organised as follows. Section 2 presents the background, project context, and related work. Section 3 describes the CTI-Fetcher framework, including its data scheme, architecture, and outputs. Section 4 evaluates the generated datasets, responses to upstream source changes, and downstream applications. Section 5 discusses the findings, Section 6 addresses threats to validity, and Section 7 concludes the paper and outlines future work.

2 Background and Related Work

2.1 Cyber Threat Intelligence

Cyber Threat Intelligence (CTI) concerns the collection, processing, analysis, and contextualisation of information about existing and emerging cyber threats to support cybersecurity decision-making [15]. CTI can originate from a wide range of internal and external sources and describe different aspects of the threat landscape, including threat actors, campaigns, tactics, techniques and procedures (TTPs), vulnerabilities, indicators of compromise, and defensive measures.

CTI is commonly distinguished according to the level at which the intelligence is intended to support decision-making. Although terminology varies across the literature, a frequently adopted classification distinguishes strategic, operational, tactical, and technical CTI [15,18]. Strategic CTI provides high-level knowledge about the threat landscape, such as threat trends, adversary motivations, and their potential implications for an organisation. Operational CTI provides contextual information concerning adversaries, campaigns, capabilities, and ongoing or anticipated attacks. Tactical CTI focuses on adversarial behaviours and TTPs, whereas technical CTI concerns low-level and often short-lived information that can be directly consumed by defensive technologies, such as malicious IP addresses, domains, file hashes, and other indicators of compromise.

This paper primarily targets structured knowledge about adversarial behaviour and related cybersecurity concepts, rather than short-lived technical indicators associated with individual incidents. The selected sources include MITRE ATT&CK, which describes adversarial tactics and techniques across Enterprise, ICS, and Mobile environments; MITRE ATLAS, which focuses on adversarial threats to AI and machine-learning systems; MITRE EMB3D, which provides threat and mitigation knowledge for embedded devices; and CAPEC, which describes common attack patterns. These sources provide complementary knowledge across the technology domains considered in the CertifAI project (2.3) and contain information that can be reused in threat modelling and risk assessment.

2.2 CTI for Threat Modelling and Risk Management

Threat modelling and cybersecurity risk assessment require knowledge about both the system under analysis and the external threat landscape. Cyber Threat Intelligence (CTI) can provide the latter through knowledge about adversarial behaviours, TTPs, vulnerabilities, and available mitigations. Tactical and operational CTI can therefore support threat identification using observed and curated adversarial knowledge rather than relying exclusively on generic threat categories [18].

Prior work integrated ATT&CK into FMECA-based cyber-risk assessment [3], evaluated it across cyber-physical domains [13,2], extended it to security architecture selection [1], and incorporated ATLAS for AI-enabled systems [20]. These

studies show the value of combining structured CTI with system-specific information.

However, supporting multiple heterogeneous CTI sources introduces an additional data-engineering challenge. Source-specific extraction and manual integration become increasingly difficult to maintain as the number and diversity of knowledge sources increase. The development of database-backed platforms such as ThreatSpider [4] further creates a need for a systematic and repeatable mechanism for populating and maintaining the underlying cybersecurity knowledge base. CTI-Fetcher addresses this requirement through a common pipeline for collecting, normalising, integrating, enriching, and exporting cybersecurity knowledge for downstream applications.

The importance of maintaining relevant threat information is also reflected in cybersecurity standards. IEC 62443-3-2 considers threat information during cybersecurity risk assessment and requires relevant threat sources to be documented, while IEC 62443-4-1 incorporates threat modelling into secure product development and requires threat models to be periodically reviewed. Together with the previous research, these requirements motivate a reusable data preparation layer through which heterogeneous and evolving cybersecurity knowledge can be systematically made available to threat modelling and risk management processes.

2.3 CertifAI Project Context

CTI-Fetcher was developed in CertifAI [7], which addresses automated and continuous cybersecurity assessment. Its railway, energy, maritime, and emergency-services use cases motivated sources spanning enterprise IT, ICS, mobile, AI/ML, and embedded systems. Its certification focus also motivated linking mitigations to requirements from IEC 62443 and the Cyber Resilience Act. Although this context scoped the initial sources, the architecture supports additional domains.

2.4 Related Work

2.4.1 CTI Collection and Source Integration Previous work has collected CTI from heterogeneous sources using different acquisition mechanisms. Babenko et al. [5] gathered approximately 4.8 million OSINT records from DNS, WHOIS, SSL/TLS certificates, and Certificate Transparency logs. ThreatWise AI [16] combines internal organisational data collected through honeypots and Wazuh with external information obtained through web crawlers from CERT/CSIRT sources, malware repositories, and social media. Niveditha M et al. [12] combine OSINT, vulnerability databases, and threat intelligence feeds, while CTI-Thinker [19] collects unstructured CTI reports, APT analyses, and vulnerability alerts using a customised crawler.

CTI-Fetcher differs from these approaches by focusing on repeatable ingestion of structured CTI and cybersecurity requirements from authoritative knowledge sources. Its current inputs include MITRE ATT&CK, ATLAS, EMB3D, and a requirements publishing API. Rather than aiming for broad OSINT discovery,

the framework retrieves predefined knowledge sources and transforms them into a reusable knowledge base for subsequent threat and risk analysis.

2.4.2 CTI Integration and Normalisation The reviewed approaches also differ in how heterogeneous information is integrated. Babenko et al. [5] transform collected information through feature engineering rather than semantic CTI normalisation. ThreatWise AI [16] represents information from different sources through the common MISP event structure, whereas Niveditha M et al. [12] apply cleaning, parsing, deduplication, and IoC extraction before further analysis. CTI-Thinker [19] uses LLM-based semantic grouping and vector similarity to align equivalent entities and relations before constructing a unified CTI knowledge graph.

CTI-Fetcher adopts a schema-driven integration approach. Source-specific objects are mapped to a unified relational data scheme containing common entities such as techniques, tactics, mitigations, requirements, assets, and properties, together with explicit mapping tables among them (Section 3.2). This provides a consistent representation for downstream processing while retaining the source context of the collected information.

2.4.3 AI-Assisted CTI Processing, Enrichment, and Reasoning ML and LLM techniques are increasingly used to extend CTI processing. Babenko et al. [5] use Gradient Boosted Decision Trees for asset risk scoring and DBSCAN for anomaly detection. ThreatWise AI [16] supplements MISP's native correlation with ML-based correlation, while Niveditha M et al. [12] use an LLM for threat categorisation and indicator correlation. CTI-Thinker [19] applies in-context learning and LoRA-fine-tuned LLMs for entity and relation extraction and semantic alignment, followed by a Graph Retrieval-Augmented Generation (GraphRAG)-based reasoning engine for threat-oriented reasoning.

In CTI-Fetcher, AI is not the primary mechanism for collection or normalisation. These functions remain deterministic, while AI is applied selectively to address information gaps. Current enrichment functions propose missing mitigation-to-requirement mappings and estimate unresolved exploitability parameters for threat techniques. The enrichment process prioritises previously curated information and deterministic evidence before using LLM estimation, while retaining provenance so that automatically generated information remains identifiable and subject to validation.

2.4.4 CTI Representation and Downstream Utilisation The reviewed systems further differ in how processed CTI is represented and used. Babenko et al. [5] combine their analytical outputs into asset-oriented risk assessments. ThreatWise AI [16] stores information in MISP and supports interoperable export through formats such as STIX for integration with systems including SIEM platforms. Niveditha M et al. [12] use processed CTI for threat classification,

risk scoring, and reporting for SOC activities. CTI-Thinker [19] represents integrated CTI as a knowledge graph supporting APT profiling, TTP mapping, attack-chain inference, and question answering.

CTI-Fetcher instead uses interconnected relational tables as an intermediate representation for downstream analysis. These tables directly populate ThreatSpider for CTI-driven threat modelling and risk assessment, while their explicit entity relationships can also be transformed into graph representations for CTI exploration. CTI-Fetcher is therefore positioned primarily as a reusable CTI engineering layer rather than a single-purpose analysis system.

Overall, CTI-Fetcher complements large-scale collection, platform integration, AI-assisted analysis, and knowledge-graph systems by preparing heterogeneous CTI as reusable, traceable data for downstream risk management.

3 CTI-Fetcher Framework Design

3.1 Design Principles

The design of CTI-Fetcher was guided by four principles derived from the requirements of the CertifAI use cases:

- **Repeatable CTI preparation:** Replace source-specific extraction and manual integration with a repeatable pipeline that supports periodic knowledge updates.
- **Multi-source and multi-domain coverage:** Integrate complementary sources selected according to the technology domains of the targeted use cases.
- **Semantic unification and integration:** Transform source-specific entities and relationships into a common scheme suitable for downstream analysis while retaining their source context.
- **Traceable enrichment:** Derive missing information and relationships required by downstream processes while preserving provenance and supporting expert validation.

3.2 Unified Data Scheme

To enable information from heterogeneous CTI and policy sources to be processed consistently, CTI-Fetcher maps the collected information into a unified relational data scheme. The scheme serves in CTI-Fetcher as a common intermediate representation between external knowledge sources and downstream applications.

The scheme separates cybersecurity entities from the relationships among them. Its main entities include adversarial techniques or threats defined in a Technique/Threat Description Table (TDT), mitigations defined in a Mitigation Description Table (MDT), security requirements defined in a Requirement Description Table (RDT), in addition to dedicated tables for tactics, assets, and system properties. Additional information relevant to threat prioritisation

is captured in the Threat Maturity Index (TMI). Relationships are represented through dedicated mapping tables, including technique (Failure mechanism) to Mitigation Table (FMT), Requirement to Mitigation Table (RMT), Technique to Tactic Table (TTT), Technique to Property Table (TPT), and Technique to Asset Table (TAT).

This separation enables concepts originating from different sources to be represented through a consistent set of entities and relationships while retaining their source context. For example, techniques obtained from ATT&CK, ATLAS, and EMB3D are represented within the TDT, while source-defined relationships between those techniques and mitigations are represented consistently through the FMT. The resulting structure therefore provides a common representation without requiring the underlying sources to use identical schemas or semantics.

Table 1. Dependencies among CTI-Fetcher generated tables. An "X" indicates that the table in the column depends on the table in the corresponding row.

	FMT	**RMT**	**TTT**	**TPT**	**TAT**	**TMI**
RDT		X				
MDT	X	X				
TDT	X		X	X	X	X
Tactics			X			
Properties				X		
Assets					X	

Table 1 specifies the generation order: each mapping table depends on the entity tables it connects.

3.3 Architecture

Figure 1 presents the CTI-Fetcher architecture and its main inputs, processing operations, outputs, and external dependencies. The main architectural elements are described below.

3.3.1 Configuration Each CTI-Fetcher execution specifies a *source*, *function*, and *output*. The source identifies one or more knowledge sources to process, while the function identifies the target data table(s), such as TDT or MDT. Multiple functions can be executed subject to the dependencies defined in Table 1. The output configuration specifies how the resulting datasets are exported.

3.3.2 Inputs to CTI-Fetcher CTI-Fetcher consumes three categories of input: CTI sources, policy sources, and, optionally, historical data.

– **CTI sources** provide machine-readable knowledge about adversarial techniques, tactics, mitigations, assets, properties, and their relationships. Current sources include ATT&CK (Enterprise, ICS, and Mobile), ATLAS, and EMB3D.

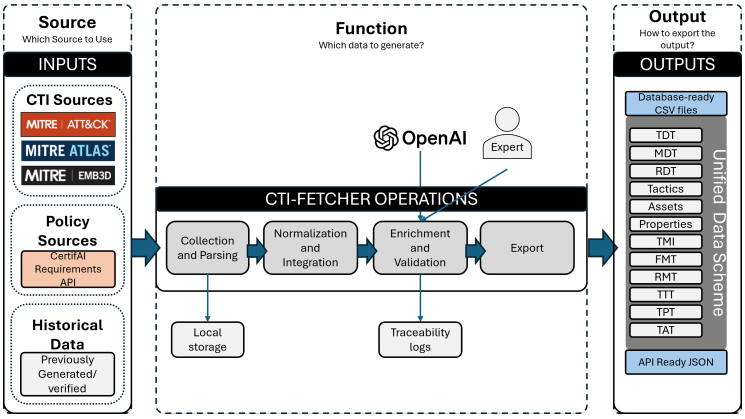


Fig. 1. Overview of the CTI-Fetcher architecture.

- **Policy sources** provide structured security requirements. The current implementation retrieves CertifAI-relevant requirements, including IEC 62443 and the Cyber Resilience Act, through a Requirements publishing API developed in CertifAI due to the lack of such a tool.
- **Historical data** are optional previously generated or curated datasets that can be supplied to the processing pipeline. Historical data serve two purposes. First, previously generated tables can act as stable local inputs when an external source is temporarily unavailable. Second, human-reviewed datasets can bootstrap enrichment by allowing previously verified information to be reused before new information is inferred. This gives curated historical information precedence over less certain automated estimation.

3.3.3 CTI-Fetcher Operations The internal processing of CTI-Fetcher can be divided into three main operations: collection and parsing, normalisation and integration, and enrichment.

Collection and parsing. CTI-Fetcher retrieves the configured knowledge sources from predefined official locations and processes the downloaded data locally. The sources expose their knowledge through different formats and distribution mechanisms, including STIX, JSON, CSV, and YAML. Source-specific parsers are therefore used to access the relevant objects and relationships. For the ATT&CK knowledge bases, CTI-Fetcher uses the `mitreattack-python` library together with functionality provided by the STIX2 library. Connections to external repositories are required for data retrieval, while subsequent processing is performed locally. At the time of the evaluation of the implementation, it was discovered that the ATLAS source was configured to retrieve a specific repository revision rather than the latest revision of the ATLAS knowledge base. This represents a known source-configuration limitation and is considered in the evaluation.

Normalisation and integration. Extracted information is mapped to the unified data scheme while retaining its source context. Source-specific adaptations are applied where concepts or relationships are represented differently. For example, EMB3D does not define tactics in the same manner as ATT&CK and ATLAS, requiring source-specific handling when mapping its techniques into the common scheme.

Enrichment and validation. Two enrichment functions are currently implemented. The RMT enrichment proposes relationships between CTI-derived mitigations and security requirements. Candidate mappings are retrieved using lexical and optional semantic similarity and subsequently assessed using LLM-based reasoning. TDT enrichment estimates missing exploitability parameters (attack vector, attack complexity, privileges required, and user interaction), prioritising curated values and inheritance from related techniques before LLM estimation. Generated outputs retain their provenance and require expert validation.

3.4 CTI-Fetcher Outputs

The datasets defined in Section 3.2 are primarily exported as CSV files using local identifiers to preserve referential relationships across tables. They can then be loaded into relevant databases like ThreatSpider's to support processes like threat modelling and risk assessment, while remaining independent of the utilising application logic and reusable by other applications. This work evaluates this reuse through both ThreatSpider and graph-based analysis.

4 Evaluation

CTI-Fetcher was evaluated from three perspectives: (1) whether repeated executions provide visibility into the evolution of CTI through the unified data scheme described in Section 3.2; (2) whether the tool remains operational when upstream CTI sources change; and (3) whether its outputs can support downstream cybersecurity analysis independently of the original sources. Four retained snapshots were considered: 13 December 2024 (R1), 1 June 2025 (R2), 10 March 2026 (R3), and 10 June 2026 (R4). We determined the intervals between successive snapshots based on project progression and available time for pilot testing, rather than a predefined experimental design.

4.1 Evolution of CTI Knowledge

Table 2 shows the evolution of selected entity and relationship tables across the four snapshots.

Across the four runs, changes were observed in both entity and relationship tables. TDT increased from 1,028 to 1,090 entries, while FMT and TTT increased from 2,160 to 2,334 and from 1,345 to 1,484 relationships, respectively. The largest change was observed in TAT, which increased from 524 to

Table 2. Evolution of CTI-Fetcher output tables across the four evaluation runs. Counts represent items retrieved from the different sources and not necessarily unique entries.

Table	R1	R2	R3	R4
TDT	1,028	1,055	1,070	1,090
MDT	218	224	224	224
RDT	185	328	328	328
Tactics	52	56	56	57
Properties	47	59	59	59
Assets	14	14	18	18
FMT	2,160	2,232	2,260	2,334
RMT_Enriched	0	628	630	0
RMT	209	222	221	221
TTT	1,345	1,446	1,463	1,484
TPT	87	90	90	90
TAT	524	524	685	842

842 technique-to-asset relationships. Other tables remained stable after particular updates; for example, Properties increased from 47 to 59 between R1 and R2, while Assets increased from 14 to 18 between R2 and R3. The number of RDT entries increased markedly from 185 to 328 following the integration of the requirement-publishing API, which was implemented after identifying the comparatively low number of requirements in R1. This increase is attributable to modifications in the requirements ingestion process rather than to changes in the underlying CTI threat-source data.

RMT_Enriched reports mappings produced by enrichment, which was introduced after R1 and executed only in R2 and R3; zero therefore means "not executed." The four additional tactics in R2 are EMB3D system layers used as higher-level groupings because EMB3D lacks tactics. Finally, the values in Table 2 represent records in the generated tables rather than globally unique cybersecurity concepts. Equivalent or closely related threats may occur in multiple CTI sources under different source-specific identifiers and are retained as separate records.

These results show that CTI evolution is not limited to the addition of new threats. Existing knowledge is also extended through new relationships to assets, tactics, mitigations, properties, and requirements. The unified representation therefore provides a common view of changes occurring across heterogeneous sources without requiring each source-native representation to be analysed separately for these concepts.

The same representation can support incremental analysis. Since successive executions use the same tables and identifiers, downstream processes can compare corresponding datasets and focus on additions, removals, and modifications rather than repeatedly processing complete source representations. This evaluation demonstrates this property through changes in the generated datasets; automated record-level incremental updating is left for future work.

A limitation identified during inspection of the source configurations concerns MITRE ATLAS. While the ATT&CK sources were retrieved from repository branches that could incorporate subsequent updates, the evaluated implemen-

tation referenced a specific ATLAS repository revision. Consequently, repeated executions could refresh the locally stored ATLAS file but would retrieve the same pinned revision rather than necessarily the latest ATLAS knowledge base. The longitudinal results should therefore be interpreted as changes in the CTI-Fetcher generated knowledge base across the retained runs, rather than as a complete measurement of the evolution of every underlying CTI source.

4.2 Resilience to Source Changes

The repeated executions also exposed limitations in the resilience of the source-specific ingestion logic described in Section 3.3.3. Two upstream changes caused confirmed interruptions.

The more substantial change concerned EMB3D, which moved from its earlier representation to a STIX-based representation. The change required major adaptation of the EMB3D ingestion and normalisation logic. Threats, mitigations, properties, and their relationships had to be extracted from STIX objects and relationships before being mapped back into the same CTI-Fetcher tables.

A later ATT&CK change removed a detection field previously accessed by CTI-Fetcher, causing a missing-key error. Because this field was not required by the downstream threat modelling and risk assessment processes, the fix was minor: the field was made optional so that its absence no longer interrupted table generation.

The two cases demonstrate different levels of source evolution. A major representation change required substantial modification of a source adapter, whereas removal of a non-essential field required only a local correction. Thus, CTI-Fetcher maintained a stable downstream data scheme, but its source-specific ingestion logic was not fully resilient to upstream structural changes.

4.3 Utility for Downstream Analysis

The downstream utility of CTI-Fetcher was evaluated through its use in ThreatSpider and in a separate knowledge-graph application. For ThreatSpider, the assessment combines feedback from CertifAI use-case providers who tested the tool with observations made during its development. The focus was on how the generated CTI tables supported the threat modelling and risk assessment functions, as well as the advantages and limitations observed in their use.

The CTI-Fetcher tables were used throughout the ThreatSpider threat modelling workflow, from system definition to threat, mitigation, and requirement identification. They were also used in several steps of the risk-assessment process defined by Amro et al [3]. Table 3 summarises the main utility and limitations observed.

The relationships support several workflow stages: Assets and Properties suggest candidate threats, FMT links them to mitigations, and RMT links mitigations to requirements. ThreatSpider consumes these relationships through the common tables, placing source-specific parsing within CTI-Fetcher.

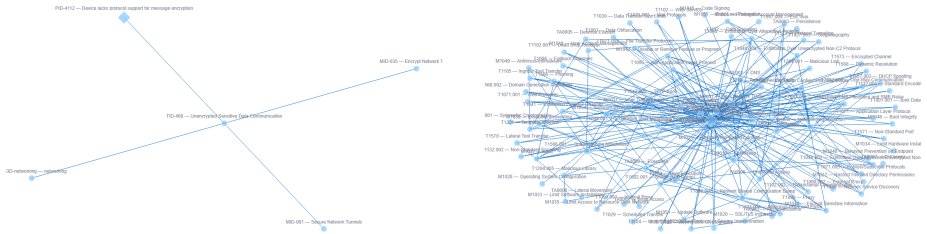
Table 3. Utility and limitations of CTI-Fetcher outputs in ThreatSpider.

ThreatSpider function	**CTI-Fetcher output**	**Utility**	**Observed limitations**
System definition	CTI Sources, Assets, Properties	Provides technology-domain, asset, and property information that supports modelling different system types and components.	Coverage depends on the source: assets are mainly provided by ATT&CK ICS and properties by EMB3D. Supporting additional domains may require changes to both CTI-Fetcher and ThreatSpider.
Threat identification	TDT, Tactics, Assets, Properties	Provides broad coverage of community-known threats and higher-level threat categories. Asset and property mappings support filtering of candidate threats by asset and property	Threat relevance still requires human verification. Tactics and contextual mappings are not consistently available across all knowledge sources and may require additional inference or adaptation.
Mitigation identification	MDT, FMT	Provides community-known mitigations and technique-to-mitigation mappings, supporting automated recommendation of mitigations for identified threats.	Mitigation relevance requires verification, while granular mappings may produce repeated mitigations for multiple threats. Aggregating such results requires careful preservation of provenance and traceability.
Requirement identification	RDT, RMT	Supports automated identification of security requirements related to selected mitigations.	Requirements are not available through a common machine-readable model, requiring a dedicated API in CertifAI. Native mitigation-to-requirement mappings are limited, motivating enrichment, whose results require human validation.
Risk assessment: failure modes and impact	Tactics	Supports scoping the assessment according to threat categories and provides candidate security consequences for impact estimation.	Some sources do not define tactics, while some tactic names provide weak impact semantics. Additional mappings to impact dimensions such as safety, financial, or operational impact are therefore required.
Risk assessment: mitigations and likelihood	MDT, RMT, TMI	Supports definition of existing mitigations and provides an automatable basis for likelihood estimation, reducing manual effort.	Organisation-specific controls are difficult to map to generic CTI mitigations. Compliance does not guarantee implementation of every mapped mitigation, and likelihood metrics require enrichment because they are not natively provided for all threats.

At the same time, the evaluation identifies a recurring trade-off between automation and human verification. CTI-Fetcher supports narrowing or automating candidate identification, but relevance cannot always be inferred directly from CTI. Human verification remains necessary when selecting relevant threats and mitigations, validating enriched mappings, relating organisation-specific controls to generic mitigations, and interpreting CTI concepts for impact assessment. Thus, the utility of the framework is primarily in reducing and structuring the analysis space rather than fully replacing expert judgement.

A second downstream use was demonstrated through a separate Graph Explorer tool that consumes the same CTI-Fetcher snapshot independently of ThreatSpider. The tool transforms the entity and relationship tables in Section 3.2 into an interactive knowledge graph, allowing CTI to be explored through direct and multi-hop relationships. For example, requirements can be explored through their directly linked mitigations, while a property can be followed through relevant threats and subsequently to their tactics and mitigations. Figure 2(a) illustrates such multi-hop analysis. This provides an analytical capability for investigating interconnected CTI outside the predefined ThreatSpider workflow while retaining the same underlying CTI snapshot.

The comprehensive linkage can, however, become difficult to interpret for highly connected entities. Figure 2(b), for example, shows how mitigation-centred exploration can produce a dense network of relationships. The Graph Explorer therefore demonstrates another potentially useful downstream application of CTI-Fetcher data, while also highlighting the need for filtering, relationship-depth control, or other mechanisms for managing graph complexity during in-depth CTI analysis.



(a) Property-centred multi-hop exploration.

(b) Dense mitigation-centred exploration.

Fig. 2. Examples of CTI-Fetcher output analysed through the Graph Explorer. The property-centred view illustrates traversal across multiple interconnected tables, while the mitigation-centred view illustrates the increasing complexity that can result from highly connected CTI entities.

5 Discussion

The longitudinal results indicate that the value of normalising CTI extends beyond integrating heterogeneous sources. The unified scheme also provides an observation layer for CTI evolution. In Section 4.1, the number of threats increased moderately compared with some of their relationships, particularly technique-to-asset mappings. Consequently, monitoring only additions to threat or technique lists can provide an incomplete picture of how CTI evolves. Changes in contextual relationships may be equally relevant for threat modelling and risk assessment.

Normalisation also creates a useful basis for incremental analysis. Once different source versions have been transformed into the same representation, changes can be identified at the level of threats, mitigations, assets, tactics, or their mappings rather than through repeated source-specific analysis. However, this visibility is limited to the concepts represented in the unified scheme. CTI-Fetcher should therefore be considered a purpose-oriented representation for downstream threat and risk analysis, rather than a complete replacement for the original knowledge bases.

The resilience results in Section 4.2 highlight an important distinction between *downstream stability* and *upstream resilience*. The output scheme remained stable for consumers, while source changes could still break the ingestion layer. This separation is beneficial because adaptations can be contained within CTI-Fetcher instead of being propagated to every downstream application. Nevertheless, fully automated continuous ingestion will require improved source-version tracking, schema-change detection, compatibility checks, and failure recovery. The current controlled approach, in which generated datasets are validated before downstream use, trades some freshness and automation for operational stability.

The two downstream applications in Section 4.3 also illustrate the benefit of separating CTI preparation from CTI utilisation. ThreatSpider consumed the tables directly for threat modelling and risk assessment, while the knowledge-graph application transformed the same entities and relationships into nodes and edges. This suggests that the relational output can function as an intermediate representation rather than being tied exclusively to ThreatSpider. Broader interoperability, however, requires evaluation with additional downstream applications and explicit versioning of the output schema.

Extendability introduces a related semantic challenge. Adding a new source is not only a matter of implementing a parser; its concepts must also map meaningfully to the unified scheme. Experimental integration of CAPEC was conducted and concluded with challenges in linking CAPEC semantics to the unified scheme, which suggests that superficially similar concepts are not necessarily semantically equivalent. For example, CAPEC prerequisites were considered as properties but then discovered that they may describe weaknesses, attack preconditions, or missing mitigations. New sources should therefore undergo both *syntactic integration* and *semantic integration*, extending the common scheme when appropriate rather than forcing incompatible concepts into existing tables.

Finally, the evaluation focuses primarily on ingestion, normalisation, evolution, resilience, and downstream utility. The AI-assisted enrichment functions described in Section 3.3.3 address information that is unavailable or incomplete in the original sources, but their accuracy is not systematically evaluated in this work. Their outputs should therefore remain traceable and subject to expert validation, as discussed in Section 3.1.

6 Threats to Validity

Several limitations should be considered when interpreting the evaluation. First, the longitudinal analysis is based on four retained snapshots collected between December 2024 and June 2026. The execution dates were determined by project activities rather than a predefined sampling schedule. Consequently, the snapshots demonstrate that CTI-Fetcher can expose changes between source versions, but they do not provide a complete history of all changes that occurred during the evaluation period.

Second, changes in the number of generated records should not be interpreted directly as changes in the overall threat landscape. The reported values represent information retrieved and normalised from the supported sources, and some tables combine information originating from multiple sources. Moreover, CTI-Fetcher itself evolved during the evaluation period, including changes to source adapters and enrichment functions. Some differences between runs may therefore result from changes in source content, source structure, tool implementation, or configuration. The longitudinal results are accordingly used to evaluate the ability of the framework to capture and represent evolving knowledge rather than to quantify the growth of CTI repositories themselves.

Third, the evaluation covers a selected set of structured CTI sources, MITRE ATT&CK, ATLAS, and EMB3D, together with structured security requirements. These sources were selected according to the technology domains and risk management needs of the targeted use cases. All evaluated CTI sources are MITRE knowledge bases; although they differ in domain coverage, data structures, distribution formats, and semantics, this selection limits the evidence for integration across independent CTI providers. The results may therefore not generalise directly to other providers or substantially different forms of intelligence, particularly unstructured reports and rapidly changing technical indicators. Integrating such sources may require extensions to both the source adapters and the unified data scheme, and their suitability for downstream threat modelling and risk assessment requires further evaluation.

Finally, downstream utility was demonstrated through ThreatSpider and a knowledge-graph application rather than through a controlled comparison with alternative CTI integration approaches. These applications show that the same normalised datasets can support different forms of analysis, but do not by themselves establish superiority, completeness, or semantic accuracy relative to other representations. Similarly, the AI-assisted enrichment mechanisms were not systematically evaluated for accuracy in this study. Their generated information therefore remains traceable and subject to expert validation. Future evaluations should include additional sources and downstream tools, controlled comparison of source versions, and dedicated assessment of the accuracy of enriched information.

An additional threat concerns source freshness. During inspection of the evaluated implementation, the ATLAS acquisition configuration was found to reference a fixed repository revision rather than the continuously updated version of the knowledge base. Consequently, changes introduced into ATLAS after that

revision are not necessarily represented in the later CTI-Fetcher snapshots. This limits the use of the longitudinal results for drawing source-specific conclusions about ATLAS evolution, although it does not invalidate the observed changes in the generated datasets themselves. Future versions should explicitly track the retrieved version or commit of every source and verify source freshness as part of each execution.

7 Conclusion

This paper presented CTI-Fetcher, a framework for preparing heterogeneous and evolving cyber threat intelligence (CTI) for downstream cybersecurity analysis. CTI-Fetcher separates CTI acquisition and interpretation from its subsequent utilisation by collecting information from multiple authoritative sources, normalising source-specific entities and relationships into a unified relational scheme, enriching selected information gaps, and exporting interconnected datasets that can be reused by external applications.

The longitudinal evaluation across four snapshots collected between December 2024 and June 2026 showed that CTI evolution extends beyond the introduction of new threats and techniques. Changes were also observed in relationships between techniques, assets, tactics, mitigations, properties, and requirements. Representing these concepts through a common scheme provides a consistent basis for observing such changes across otherwise heterogeneous sources. The evaluation also exposed two types of upstream structural change: a substantial representation change in EMB3D and the removal of a previously available ATT&CK field. While these changes required adaptations to the source-specific ingestion logic, the downstream representation remained unchanged. This illustrates the role of CTI-Fetcher as a decoupling layer between evolving knowledge sources and applications that consume CTI.

The generated datasets were further consumed by ThreatSpider for CTI-driven threat modelling and risk assessment and by a separate graph-exploration application, demonstrating that the resulting representation is not restricted to a single downstream implementation. CTI-Fetcher therefore provides a reusable data-engineering foundation through which structured CTI can be maintained and made available to different cybersecurity analysis processes.

Future work will focus on improving resilience to upstream changes through source-version tracking, schema-change detection, automated validation, and failure recovery. Further work will also investigate record-level change detection, explicit versioning of the unified output scheme, integration of additional CTI sources, and systematic evaluation of the AI-assisted enrichment functions. These developments are intended to move CTI-Fetcher from its current controlled update process towards more robust and automated continuous CTI integration for cybersecurity risk management.

Acknowledgements This work was supported by the European Union's Horizon Europe research and innovation programme through the CERTIFAI project under Grant Agreement No. 101120606.

References

1. Amro, A., Gkioulos, V.: Cyber risk management for autonomous passenger ships using threat-informed defense-in-depth. International Journal of Information Security **22**(1), 249–288 (2023)
2. Amro, A., Gkioulos, V.: Evaluation of a cyber risk assessment approach for cyber–physical systems: Maritime-and energy-use cases. Journal of Marine Science and Engineering **11**(4), 744 (2023)
3. Amro, A., Gkioulos, V., Katsikas, S.: Assessing cyber risk in cyber-physical systems using the att&ck framework. ACM Transactions on Privacy and Security **26**(2), 1–33 (2023)
4. Amro, A., Kavallieratos, G.: Threatspider: Cti-driven semi-automated threat modelling for cybersecurity certification. In: 2025 IEEE International Conference on Cyber Security and Resilience (CSR). pp. 619–625. IEEE (2025)
5. Babenko, T., Kolesnikova, K., Abramkina, O., Vitulyova, Y.: Automated osint techniques for digital asset discovery and cyber risk assessment. Computers **14**(10), 430 (2025)
6. Bromander, S., Swimmer, M., Muller, L.P., Jøsang, A., Eian, M., Skjøtskift, G., Borg, F.: Investigating sharing of cyber threat intelligence and proposing a new data model for enabling automation in knowledge representation and exchange. Digital Threats: Research and Practice (DTRAP) **3**(1), 1–22 (2021)
7. CERTIFAI Consortium: CERTIFAI: Agile conformance assessment for cybersecurity certification enhanced by artificial intelligence. `https://certifai.info/` (2026), horizon Europe project 101120606; accessed 8 September 2026
8. MITRE Corporation: CAPEC: Common attack pattern enumeration and classification. `https://capec.mitre.org/` (2026), knowledge base of attack patterns used by adversaries to exploit known weaknesses; accessed 8 September 2026
9. MITRE Corporation: MITRE ATLAS: Adversarial threat landscape for artificial-intelligence systems. `https://atlas.mitre.org/` (2026), knowledge base of adversary tactics and techniques for AI-enabled systems; accessed 8 September 2026
10. MITRE Corporation: MITRE ATT&CK. `https://attack.mitre.org/` (2026), adversarial Tactics, Techniques, and Common Knowledge; accessed 8 September 2026
11. MITRE Corporation: MITRE EMB3D: A threat model for embedded devices. `https://emb3d.mitre.org/` (2026), knowledge base of cyber threats and associated mitigations for embedded devices; accessed 8 September 2026
12. Niveditha, M., Swetha, M., Krishna, M.S.R., Rashmi, S., Pallavi, B., Neelima, M.L.: Smart cyber threat intelligence system for autonomous threat aggregation and reporting. In: 2026 International Conference on AI-Driven Innovations & Applications (IC-AIDA). pp. 1–6. IEEE (2026)
13. Oruc, A., Amro, A., Gkioulos, V.: Assessing cyber risks of an ins using the mitre att&ck framework. Sensors **22**(22), 8745 (2022)
14. Ramsdale, A., Shiaeles, S., Kolokotronis, N.: A comparative analysis of cyber-threat intelligence sources, formats and languages. Electronics **9**(5), 824 (2020)
15. Sakellariou, G., Fouliras, P., Mavridis, I., Sarigiannidis, P.: A reference model for cyber threat intelligence (cti) systems. Electronics **11**(9), 1401 (2022)

16. Spyros, A., Koritsas, I., Papoutsis, A., Panagiotou, P., Chatzakou, D., Kavallieros, D., Tsikrika, T., Vrochidis, S., Kompatsiaris, I.: Ai-based holistic framework for cyber threat intelligence management. IEEE Access **13**, 20820–20846 (2025)
17. Strom, B.E., Applebaum, A., Miller, D.P., Nickels, K.C., Pennington, A.G., Thomas, C.B.: Mitre att&ck: Design and philosophy. Tech. rep., Technical report (2018)
18. Tatam, M., Shanmugam, B., Azam, S., Kannoorpatti, K.: A review of threat modelling approaches for apt-style attacks. Heliyon **7**(1) (2021)
19. Yang, X., Zhong, R., Chen, Y., Peng, G., Yao, D., Chen, C., Wang, C., Zhang, D., Zhou, Y., Yang, Z.: Cti-thinker: an llm-driven system for cti knowledge graph construction and attack reasoning. Cybersecurity **9**(1), 106 (2026)
20. Yousaf, A., Amro, A., Kwa, P.T.H., Li, M., Zhou, J.: Cyber risk assessment of cyber-enabled autonomous cargo vessel. International Journal of Critical Infrastructure Protection **46**, 100695 (2024)