IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 October 2022, accepted 3 November 2022, date of publication 8 November 2022, date of current version 14 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3220637

RESEARCH ARTICLE

Multiontology Construction and Application of Threat Model Based on Adversarial Attack and Defense Under ISO/IEC 27032

XUAN HU, DEBIN CHENG, JUNMING CHEN, XIANTAO JIN, AND BO WU
CEPREI Laboratory, Information Security Center, Guangzhou 510610, China
Key Laboratory of Ministry of Industry and Information Technology, Guangzhou 510610, China

Corresponding author: Debin Cheng (435207985@qq.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2021YFB2012200, and in part by the National Key Research and Development Program of China under Grant 2019YFC0118800.

ABSTRACT Research pertaining to threat modeling is significant. However, the existing threat modeling methods suffer from ambiguity, heterogeneity and incompleteness; furthermore, the threat models at different abstraction levels are separated from each other, and the model elements are fragmented. In the knowledge engineering community, an ontology is an explicit specification of a conceptualization. Introducing the ontology method into the study of threat models is an effective way to solve the above problems. This paper creates a multiontology framework for the threat model of information systems (IS) based on domain knowledge (attack and defense knowledge), engineering experience, and industry standards (ISO/IEC 27032). The multiontology framework includes a generalized ontology (GO), a domain ontology (DO), and an application ontology (AO). This paper builds the ontology of each layer and ultimately presents case studies. The results show that the multiontology threat model based on adversarial attack and defense effectively solves the above problems of the existing threat modeling methods. In addition, systematic threat modeling using the multiontology method can be used not only for attack path-based threat analysis but also for adversarial attack and defense-based threat analysis. This method can help detect security issues and effectively guide security personnel.

INDEX TERMS Cybersecurity, threat modeling, STRIDE, ATT&CK, adversarial attack and defense, ontology.

I. INTRODUCTION

The cyberspace is a complex environment resulting from the interaction of people, software and services on the internet, supported by worldwide distributed physical information and communications technology (ICT) devices and connected networks. However there are security issues that are not covered by current information security, internet security, network security and ICT security best practices as there are gaps between these domains, as well as a lack of communication between organizations and providers in the cyberspace. This is because the devices and connected networks that have supported the cyberspace have multiple owners, each with their own business, operational and regulatory concerns. The different focus placed by each organization and provider in the cyberspace on relevant security domains where little or no input is taken from another organization or provider has resulted in a fragmented state of security for the cyberspace. In order to effectively solve the above problems, the international organization for standardization (ISO) has released the ISO/IEC 27000 series of standards for information security and information security management system (ISMS) [1]. And the ISO/IEC 27032 [2] is one of them. The first area of focus of this international standard is to address cyberspace security or cybersecurity issues which concentrate on bridging the gaps between the different security domains in the cyberspace [2]. The second area of focus of this international standard is collaboration, as there is a need for efficient

The associate editor coordinating the review of this manuscript and approving it for publication was Kashif Saleem.

and effective information sharing, coordination and incident handling amongst stakeholders in the cyberspace. This collaboration must be in a secure and reliable manner that also protects the privacy of the individuals concerned. Many of these stakeholders can reside in different geographical locations and time zones, and are likely to be governed by different regulatory requirements [2].

With the development of new technologies such as big data, cloud computing, and the internet of things (IoT), the scale and complexity of information systems (IS) are increasing, and cybersecurity issues have become more serious. Cybersecurity threat refers to the potential cause of an unwanted incident, which may result in harm to a system, individual or organization. The harm is caused by an attacker attempting to destroy, expose, alter, disable, steal or gain unauthorized access to or make unauthorized use of an asset. Therefore, an analysis of cybersecurity threats is needed. Threat modeling is an important part of cybersecurity threat analysis. It aims to improve security through the practice of identifying potential threats, attacks, and vulnerabilities for the purpose of defining countermeasures to prevent or mitigate loss, damage or destruction of an application, system or data. With the development of attack tools and methods, the large-scale integration of security data and adversarial attack and defense has become an important developmental direction. How to effectively discover the clues of advanced threats in massive data and how to effectively transform the experience and knowledge of cybersecurity experts into replicable and scalable data analysis capabilities are urgent problems to be solved. At present, representative threat modeling methods include (1) the attack tree model-based threat modeling method; (2) Microsoft STRIDE (**s**poofing, **t**ampering, **r**epudiation, **i**nformation disclosure, **d**enial of service, and **e**levation of privilege); and (3) MITRE **a**dversarial **t**actics, **t**echniques, and **c**ommon **k**nowledge (ATT&CK) framework, which is a globally-accessible knowledge base of adversary tactics and techniques based on real-world observations [3], [4]. Attack trees provide a "formal, methodical way of describing the security of systems, based on varying attacks". They are extensively used in threat assessment. However, attack tree methods require extensive security skills and adversarial attack and defense experience of attack tree designers. Moreover, security skills and adversarial attack and defense experience are scattered, and a systematic knowledge system has not been formed, which poses difficulty for sharing and reuse. STRIDE is a model of threats. It is used in conjunction with a model of the target system that can be constructed in parallel. This includes a full breakdown of processes, data stores, data flows, and trust boundaries. However, due to the high abstraction level of the STRIDE model, it is difficult to express specific attack behaviors and related data, countermeasures, and configuration resources. The ATT&CK framework can effectively solve the above problems. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community [3]. In short, the primary difference between models at different levels is that different abstraction levels determine the expressiveness of the model and the granularity of the concepts that can be covered.

This paper adopts the hierarchical ontology modeling method to establish the mapping between threat models at different abstraction levels under ISO/IEC 27032. The ontology method can solve the problems of ambiguity, heterogeneity, and incompleteness of the existing threat models. The hierarchical modeling method can solve the problem that the threat models at different abstraction levels are separated from each other and avoid the fragmentation problem caused by the scattered model elements. Threat models based on multiple ontologies are structured and reasonable data models with inheritance and association properties. Moreover, this approach can realize the accurate descriptions and classifications of security events, lay the foundation for automated reasoning and detection of security threats, and realize threat knowledge sharing and reuse among different security vendors (devices) [4]. In summary, the paper provides the following main contributions:

1) This paper adopts the hierarchical ontology modeling method to establish the mappings between threat models at different abstraction levels under ISO/IEC 27032. This method can solve the problem that the threat models at different abstraction levels are separated from each other and avoid the fragmentation problem caused by the scattered model elements.
2) This paper selects generalized ontology concept classes according to ISO/IEC 27032 and MITRE ATT&CK: Design and Philosophy [7] and establishes the relationships between these concept classes. It solves the following problems: (1) Ambiguity and heterogeneity. At the beginning of ontology modeling, multiple roles, such as user, regulator, provider, and third-party evaluation agency, are integrated into the generalized ontology. This approach can eliminate ambiguity and heterogeneity to a certain extent. (2) Incompleteness. The threat model proposed in this paper is based on adversarial attack and defense, avoiding the incompleteness problem caused by the single perspective of the traditional threat modeling method to a certain extent.
3) This paper divides the system into (1) the information system (IS), including the conventional IS, cloud computing systems, and internet of things (IoT); and (2) the industrial control system (ICS), which can be mapped with the concepts "Assets" and "IS" in the generalized ontology. The concepts and associations contained in the domain ontology established on this basis are specific, which is an important feature that distinguishes it from other existing threat models.
4) According to the protocol type of the application layer in the open system interconnection (OSI) seven-layer model, the concepts and associations contained

in the application ontology established on this basis are specific, which is also an important feature that distinguishes it from other existing threat models. The mappings between the concepts in application ontology and the concepts in domain ontology (IProtocols, CISProtocols, DISProtocols, and NProtocols) can then be further established.

5) By integrating the ontology method into threat modeling and vulnerability analysis, this paper presents an improved penetration testing process. Therefore, the systematic collection and modeling of fragmented knowledge can be achieved based on ontology methods. The results can be solidified to form a systematic knowledge system to guide other security personnel.

The rest of this paper is divided into the following sections: section 2 presents the state of the art in threat modeling technologies, security ontology construction and threat ontology construction. Section 3 describes the multiontology framework of threat model. Section 4 describes the multiontology construction process of threat model. Section 5 presents case studies involving, (1) the improved penetration testing process; (2) the threat analysis of IOT system; (3) the attack path-based threat analysis of a conventional IS; (4) the adversarial attack and defense-based threat analysis of an industrial control system (ICS). Finally, section 6 concludes the study.

II. RELATED WORK

A. THREAT MODELING TECHNOLOGIES

1) ATTACK TREE MODEL

The tree models in the security analysis community mainly include attack trees, fault trees and threat trees. They all have the advantage of being structured. The fault trees serve as an effective modeling tool in the software engineering field; however, they are not suitable for external attack modeling but are more suitable for internal fault analysis. The threat trees use threats as the basis for building a system attack tree. Therefore, they can be considered a subset of the attack trees. The attack trees provide a formal methodology for analyzing the security of systems and subsystems. They provide a way to think about security, to capture and reuse expertise about security and to respond to changes in security [5]. However, the primary disadvantage of attack trees is that the threat and asset ontologies are not fully constructed. For example, an attack tree only focuses on hosts, vulnerabilities, authorities, and cyberspace. The lack of an ontological method makes the attack tree theory difficult to apply in real scenarios.

2) MICROSOFT STRIDE

STRIDE threat model divides threats into 6 categories: ‘‘spooling’’, ‘‘tampering’’, ‘‘repudiation’’, ‘‘information disclosure’’, ‘‘denial of service (Dos)’’, and ‘‘elevation of privilege’’. The STRIDE chart involving properties, threats, definitions and examples is shown in Table 1 [6].

TABLE 1. Stride chart.

Property	Threat	Definition	Example
authentication	**s**poofing	impersonating something or someone else.	pretending to be any of billg, microsoft.com or ntdll.dll
integrity	**t**ampering	modifying data or code	modifying a DLL on disk or DVD, or a packet as it traverses the LAN.
non-repudiation	**r**epudiation	claiming to have not performed an action.	“I didn’t send that email,” “I didn’t modify that file,” “I certainly didn’t visit that web site, dear!”
confidentiality	**i**nformation Disclosure	exposing information to someone not authorized to see it	allowing someone to read the Windows source code; publishing a list of customers to a web site.
availability	**d**enial of Service	deny or degrade service to users	crashing Windows or a web site, sending a packet and absorbing seconds of CPU time, or routing packets into a black hole.
authorization	**e**levation of Privilege	gain capabilities without proper authorization	allowing a remote internet user to run commands is the classic example, but going from a limited user to admin is also EoP.

3) MITRE ATT&CK

MITRE ATT&CK is a curated knowledge base and model for cyber adversary behavior, reflecting the various phases of an adversary’s attack lifecycle and the platforms they are known to target. It focuses on how external adversaries compromise and operate within computer information networks [7]. To date MITRE has defined three technology domains-enterprise (representing traditional enterprise networks and cloud technologies), mobile (for mobile communication devices), and the ICS (for industrial control systems). Within each technology domain, the ATT&CK defines multiple ‘‘platforms’’-the system an adversary is operating within. A platform may be an operating system or application (e.g. Microsoft Windows). Techniques and sub-techniques can apply to multiple platforms [7]. Each high-level component of ATT&CK is related to other components in some way. ATT&CK model relationships can be visualized in Figure 1.

The STRIDE model introduced above has a high level of abstraction, and it is difficult to express specific attack behaviors and related data, countermeasures, and configuration resources. For example, we can map an indicator of compromise (IOC) or an attack behavior to the command and control (C2) step of the attack chain. This reminds the defender that necessary measures need to be taken. However,

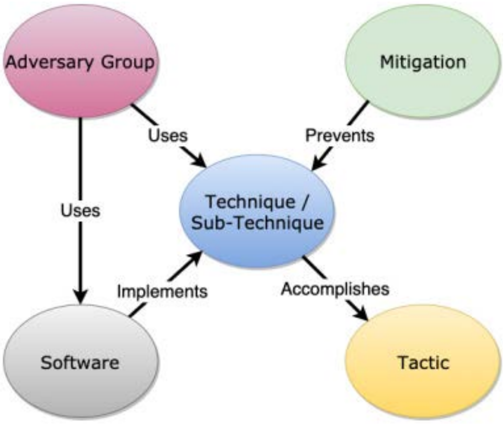


FIGURE 1. ATT&CK model relationships.

the attack chain model faces difficulty describing what kind of measures to take. In the ATT&CK model, this IOC may correspond to the tactic ''C2''. A ''multihop proxy'' is used to achieve tactical goals; thus, the corresponding general defensive measures can be further obtained. However, what the ATT&CK (middle-level) describes is still the abstraction of tactics, techniques and procedures (TTPs), and fine-grained division is still needed to describe specific behaviors.

4) CAPEC & CWE

Understanding how the adversary operates is essential to effective cybersecurity. Common attack pattern enumeration and classification (CAPEC) helps by providing a comprehensive dictionary of known patterns of attack employed by adversaries to exploit known weaknesses in cyber-enabled capabilities. ''Attack Patterns'' are descriptions of the common attributes and approaches employed by adversaries to exploit known weaknesses in cyber-enabled capabilities. Each attack pattern captures knowledge about how specific parts of an attack are designed and executed, and gives guidance on ways to mitigate the attack's effectiveness. Attack patterns help those developing applications, or administrating cyber-enabled capabilities to better understand the specific elements of an attack and how to stop them from succeeding [8]. Common weakness enumeration (CWE) is a community-developed list of software and hardware weakness types. It serves as a common language, a measuring stick for security tools, and as a baseline for weakness identification, mitigation, and prevention efforts. Targeted at both the development and security practitioner communities, the main goal of CWE is to stop vulnerabilities at the source by educating software and hardware architects, designers, programmers, and acquirers on how to eliminate the most common mistakes before products are delivered [9]. CAPEC and CWE are at a relatively low level compared to STRIDE and ATT&CK.

The conceptual abstraction level of a threat model is crucial to distinguishing it from other threat models and threat knowledge bases. The MITRE conducts a coarse-grained division of the conceptual abstraction levels of threat models and threat knowledge bases, as shown in Figure 2. The STRIDE threat model and cyber kill chain framework [10] are high-level models. ATT&CK is a mid-level model.

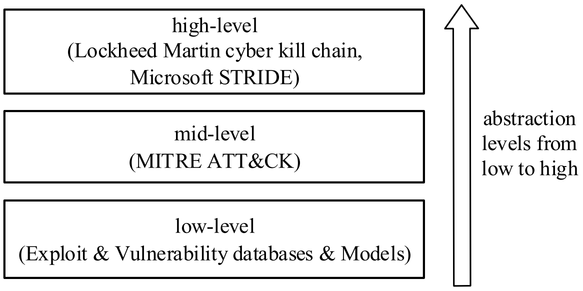


FIGURE 2. Abstraction levels of threat models.

B. SECURITY ONTOLOGY CONSTRUCTION

Donner first proposed the need for ontology in the field of information security in 2003 [11]. Schumacher presented a security ontology for maintaining a knowledge base of security patterns [12]. The core concepts of this ontology include assets, threats, attacks, vulnerabilities, attackers, risks, countermeasures, asset owners, security objectives (confidentiality, integrity, availability, etc.), and their relationships. Stefan Fenz et al. [13], [14], [15] designed their security ontology based on Landwehr's [16] summary of concepts and classifications in the field of information security and combined it with the ISO/IEC 27001 standard ontology for risk analysis and security standard certification. The OntoWorks platform was introduced, enabling users to access, reason and observe visualized ontology data. Almut Herzog et al. [17] constructed a security ontology based on ''Principles of information security (second edition)'' by Whitman and Mattord [18]. The core concepts of this ontology are threats, assets, countermeasures, and vulnerabilities. This ontology contains 88 threat classes, 79 asset classes, 133 countermeasures, and 34 relationships. Carlos Blanco et al. [19] analyzed each security ontology and concluded that most of the research on security ontologies focuses on a specific domain, and a complete security ontology has not yet been realized. Teresa Pereira and Henrique Santos [20] proposed a conceptual model of an information security ontology and designed the core concepts of security ontology. Carlos Blanco et al. [21] summarized and compared the previous security ontologies and conducted in-depth research in the following three aspects: (1) they presented the core requirements for security ontology integration; (2) they proposed that the best way to obtain integrated ontologies is through formal comparative research; and (3) they confirmed that it is most appropriate to use Protégé software and the ontology web language (OWL) recommended by the Worldwide Web Consortium (W3C) to complete the ontology integration. Gao et al. proposed an ontology-based model of network and computer attacks for security assessment and standards classifications that establishes relationships among network security services, threats, vulnerabilities and causes of failures [22]. Gyrard et al.

TABLE 2. Comparison table.

Threat modeling methods and related techniques	Abstraction level	Focus/ perspective	Tool support	Extensible/ flexible	Good documentation	Easy to use	High maturity
STRIDE	√high □mid □low	defender	√	√	√	√	√
Attack trees	√high □mid □low	attacker		√	√	√	√
KillChain	√high □mid □low	attacker			√		√
MCKC	√high □mid □low	attacker			√		√
ICS Cyber Kill Chain	√high □mid □low	attacker			√		√
ATT&CK	□high √mid □low	attacker	√	√	√		√
STUCCO	□high √mid □low	automatic platform	√	√	√		√
CWE	□high □mid √low	principle			√		√
CCE	□high □mid √low	principle			√		√
CVSS	□high □mid √low	scoring			√		√
CAPEC	□high □mid √low	principle	√	√	√		√
CTI	□high □mid √low	principle			√		√
STIX	□high □mid √low	principle			√		√
CybOX	□high □mid √low	principle			√		√
OVM	□high □mid √low	knowledge model	√		√		√
CVO	□high □mid √low	knowledge model	√		√		√

proposed an ontology for security toolboxes, attacks and countermeasures from a secure e-governance applications perspective for capturing and presenting concepts of security requirements in the application development of security expert knowledge [23].

C. THREAT ONTOLOGY CONSTRUCTION

Cyber threat ontology (CTO) from a security perspective describes organizational security concepts, properties relationships, and interdependencies in a formal and structured approach for analysis and intelligence gatherings [24]. The goal of CTO considers the extraction of relevant attack instances and threat information from data to ensure consistency and accuracy in the cybersecurity concepts for knowledge reuse in the threat intelligence domain [24]. The work of Ulicny et al. [25] is considered among the first in this field and has taken an important step forward. The authors have manually constructed a cyber threat intelligence (CTI) ontology based on structured threat information expression (STIX) to support cybersecurity operators in their cyber threat hunting tasks. Unfortunately, the authors have overlooked the vocabulary overlapping problem [25]. Unified Cybersecurity Ontology (UCO) was proposed by Syed et al. [26] in 2016. It is an extension to an existing intrusion detection system (IDS) ontology to support the integration of heterogeneous data and knowledge schema from different cybersecurity systems. This ontology incorporates and maps some of the existing cybersecurity ontologies for information sharing and exchange, such as CVE, common configuration enumeration (CCE) [27], common vulnerability scoring system (CVSS) [28], CAPEC, cyber observable expression (CybOX), KillChain [29], and STUCCO [30]. An ontology can provide an analysis framework for cybersecurity intelligence and domain knowledge, such as the reachability matrix ontology (RMO) [31] and the modified cyber kill chain (MCKC) model [32], [33], which are used for cybersecurity situational calculation, analysis and prediction, respectively. Qamar et al. [34] proposed an ontology for risk analysis and threat actor profiling that represents STIX concepts with additional concepts from CybOX and CVE. However, due to the use of STIX 1.0, which is seen as an obstacle to sharing information about cyber threats, this ontology has a weakness in semantic expression [35]. In [36], a method was proposed to model a knowledge graph ontology based on STIX. This method generates an ontology schema by classifying the concepts in the network security field with low redundancy and a strong structural hierarchy. Also, Zhao et al. [37] proposed a unified representation of CTI using an ontology-based model built from concepts from STIX V2.1. Although these two works contribute strongly to providing a decision support tool in cybersecurity, the ontological reasoning has not been addressed. Mavroeidis et al. [38] presented an ontological approach to automatically deduce the types of threat actors by capturing their polymorphic techniques and studying their behavior over time. Wang et al. [39] proposed an ontology for vulnerability management and analysis (OVM) populated with instances of vulnerabilities from national vulnerability database (NVD). More recently, Syed [40] has also presented a conceptual knowledge model to the vulnerability management domain named cybersecurity vulnerability ontology (CVO). In the construction of CTI knowledge on malware, Rastogi et al. [41] have developed an ontology named MALOnt, which allows the extraction of information on this type of threat and the generation of knowledge graphs. Mozzaquatro et al. proposed an ontology-based cybersecurity framework for the IoT that considers design time and provides a dynamic method to build security and run time that monitors the IoT environment for analysis [42]. Jia et al. proposed a practical approach to constructing a knowledge graph for cybersecurity by using machine learning (ML) to extract entities and building ontologies to obtain cybersecurity based knowledge and security ontology with model-driven architecture for software development [43]. Liu Bin et al. proposed an ontology named OntoCSA4Sat designed for cybersecurity defense of satellites [44].

D. COMPARISON TABLE

The features of the threat modeling methods and related techniques introduced above are summarized in Table 2 [45], [46], [47], [48], [49], [50], [51].

III. MULTI-ONTOLOGY FRAMEWORK CONSTRUCTION OF THE THREAT MODEL OF IS

A. SHORTCOMINGS OF TRADITIONAL THREAT MODELING METHODS

Traditional threat modeling methods suffer from ambiguity, heterogeneity and incompleteness.

1) AMBIGUITY

An important reason for this problem is the lack of an effective knowledge sharing bridge between users, regulators, providers and third-party evaluation agencies.

2) HETEROGENEITY

Different teams with multiple views and multiparadigm development methods are widely used in the development of such complex information systems (ISs), which increases heterogeneity.

3) INCOMPLETENESS

Traditional threat modeling methods generally model from a single perspective, leading to incompleteness problems. It is necessary to construct a complete threat model from both the attacker's perspective and the defender's perspective.

B. ONTOLOGY FORMALIZATION

An ontology is an explicit specification of a conceptualization [52], [53]. Ontologies provide interrelations between elements, hierarchy among domain concepts, data structure and the integration of heterogeneous information [54]. The different ontology classes, relationships, constraints and axioms define a common vocabulary to share knowledge [55].

Formally, an ontology can be defined as the tuple [56]:

$$O = (C, H, I, R, P, A) \tag{1}$$

where: $C = C_C \cup C_I$ is the set of entities of the ontology. The set C_C consists of classes, i.e., concepts that represent entities that describe a set of objects, while the set C_I is constituted by instances.

$H = \{kind\_of(c_1, c_2)|c_1 \in C_C, c_2 \in C_C\}$ is the set of taxonomic relationships between the concepts, which define a concept hierarchy and are denoted by "kind_of (c$_1$, c$_2$)", meaning that c$_1$ is a subclass of c$_2$.

$I = \{is\_a(c_1, c_2)|c_1 \in C_I \wedge c_2 \in C_C\} \cup \{prop_K(c_i, value)|c_i \in C_I\} \cup \{rel_K(c_1, c_2, \ldots, c_n)|\forall i, c_i \in C_I\}$ is the set of relationships between ontology elements and its instances.

$R = \{rel_K(c_1, c_2, \ldots, c_n)|\forall i, c_i \in C_C\}$ is the set of ontology relationships that are neither "kind_of" nor "is_a". The relationships between concepts mainly have two types: hierarchical relationships and non-hierarchical relationships [57].

$P = \{prop_K(c_i, datatype)|c_i \in C_C\}$ is the set of properties of ontology entities and its basic datatype.

$A = \{condition_x \Rightarrow conclusion_y(c_1, c_2, \ldots, c_n)|\forall j, c_j \in C_C\}$ is a set of axioms, rules that allow checking the consistency of an ontology and infer new knowledge through some inference mechanism. The term "condition$_x$" is given by $condition_x = \{(cond_1, cond_2, \ldots, cond_n)|\forall z, cond_z \in H \cup I \cup R\}$.

The above ontology elements are highly compatible with the OWL. It is beneficial to use tools for ontology editing and automated reasoning. Therefore, threat models based on multiple ontologies are structured and computable data models with inheritance and association properties. Moreover, they can realize unambiguous, consistent and complete descriptions and classifications of security events, lay the foundation for automated reasoning and detection of security threats, and realize threat knowledge sharing and reuse among different security vendors (devices).

C. MULTIONTOLOGY FRAMEWORK OF THE THREAT MODEL OF IS

High-quality threat modeling requires a variety of knowledge; therefore, the knowledge system can be modeled by a knowledge-aided design system (KADS) [52]. The knowledge hierarchy in this model is clearly divided, and each layer of knowledge exhibits good maintainability and reusability. Figure 3 shows an example of a security model that involves elements such as concepts and relationships. However, these elements are obviously not sufficient for building a complete knowledge system. In addition, the division of the knowledge hierarchy is also lacking. Therefore, to enable the above knowledge model to play a role in knowledge sharing and reuse, it is necessary to integrate relatively independent knowledge layers through the ontology to form a knowledge system.

This paper constructs generalization layer, domain layer and application layer ontologies. The **m**ulti-**o**ntology **f**ramework of the **t**hreat model of **IS** (ISTMOF) is defined by

$$\text{ISTMOF} =< \text{ISTGO}, \text{ISTDO}, \text{ISTAO} >$$

where ISTGO, ISTDO and ISTAO represent IS threat generalization ontology, domain ontology and application ontology, respectively.

The relationships between the ontologies in this definition are shown in Figure 4. There are hierarchical relationships between the ontologies. The ISTGO can obtain the ISTDO through instantiation and then obtain the ISTAO. In this framework, domain knowledge, engineering experience, and industry standards are the source of ISTDO. Domain experts, users, regulators, providers, and third-party evaluation agencies can all participate in the construction of a multiontology

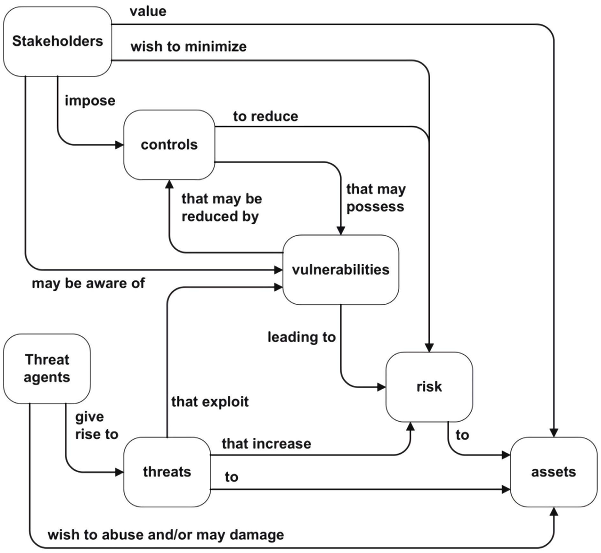


FIGURE 3. Security concepts and relationships.

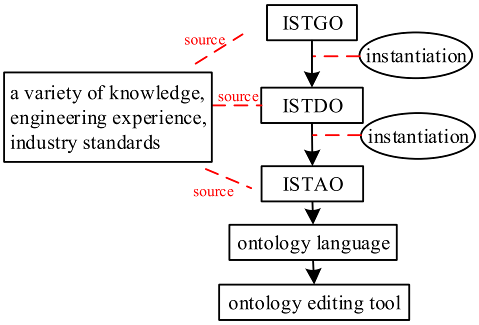


FIGURE 4. Multiontology framework of the threat model of IS.

framework; thus, the framework is based on multiple viewpoints.

IV. CONSTRUCTION OF ISTGO, ISTDO AND ISTAO

A. ONTOLOGY ELEMENTS

Ontology elements include basic elements and extended elements. Based on a tuple (1), the universal set of elements of ISTGO is given as Element Set = {C, H, I, R, OPs, DPs, P^R, P^C, M, A}, where C, H, I, R, and A are the same as defined in tuple (1). OWL Properties represent relationships between two individuals. There are two main types of properties, Object properties (OPs) and Datatype properties (DPs). OPs link an individual to an individual. Dps link an individual to an XML Schema Datatype value or an rdf literal [58]. An OP is the same as the R in content, but they have different meanings. P^R stands for property restrictions (type, range, maximum cardinality, minumum cardinality, etc.). P^C stands for property characteristics. M represents the mappings between different layers of ontology. The above elements constitute the basic skeleton of an ontology.

B. ONTOLOGY HIERARCHY

1) ISTGO CONSTRUCTION

a: ISTGO CONSTRUCTION PROCESS

The ISTGO construction process includes the elicitation of domain knowledge; the elicitation of concepts, concept attributes, concept hierarchies and concept relationships; and the use of a formal language to represent these definitions.

The ISTGO is defined by

$$\text{ISTGO} = < \text{C, OPs, } P^C\text{, H, R, A} >$$

The construction of ISTGO can be realized by constructing its concept classes, class hierarchies, relationships, properties, and property characteristics.

b: ISTGO CONCEPT CLASSES AND CLASS HIERARCHIES

Figure 5 shows a portion of the hierarchy of the concept classes and the relationships in the ISTGO. The concept class with a "*" is a nonterminating concept class, and the rest are all terminating concept classes. If an inheritance relationship is defined by

Definition 4.2.1 A subclass automatically shares the properties and structure of its superclass in the ISTGO concept class hierarchy.

then the subclass of the nonterminating concept class forms the inheritance relationship with its superclass.

c: ISTGO CONCEPT RELATIONSHIPS AND CONCEPT SPACE

ISTGO concept semantic associations are obtained on the basis of the hierarchy of the concept classes (the left of the arrow is the source concept node, and the right is the destination concept node). The concept space of ISTGO can be obtained by integrating concept classes, concept class hierarchies, relationships, properties, and property characteristics, as shown in Table 3. This is the informal representation of the initial ISTGO. It can be proved that the inheritance relationship is a partial ordering relation, which can be denoted as: $a \preceq b$.

Theorem 4.2.1 The inheritance relationship in the ISTGO is a partial ordering relation.

Proof: Let F be a nonempty set consisting of nonterminating concept classes and their subclasses. And the inheritance relationship H is a relation on the set F.

(1) $\forall a \in F$, $(a,a) \in H$. The binary relation H on the set F is reflexive if $(a,a) \in H$ for every $a \in F$.

(2) $\forall a, b \in F$, $((a, b) \in H) \cap ((b, a) \in H) \rightarrow (a = b)$. The relation H on the set F is antisymmetric if $(a,b) \in H$ and $(b,a) \in H$ imply $a = b$.

(3) $\forall a, b, c \in F$, $((a, b) \in H) \cap ((b, c) \in H) \rightarrow (a, c) \in H$. The binary relation H on the set F is transitive if, whenever $(a,b) \in H$ and $(b,c) \in H$, then $(a,c) \in H$.

Since the relation H on the set F satisfies the above three properties, the relation H is called a partial ordering relation on the set F, denoted as: $a \preceq b$. The set F with a partial ordering relation is called a partially ordered set (poset).

Figure 6 shows the unified model language (UML) diagram representations of the concepts and relationships of the ISTGO. "⟶" represents the inheritance relationship and "⇢" represents the relationships other than the inheritance relationship.

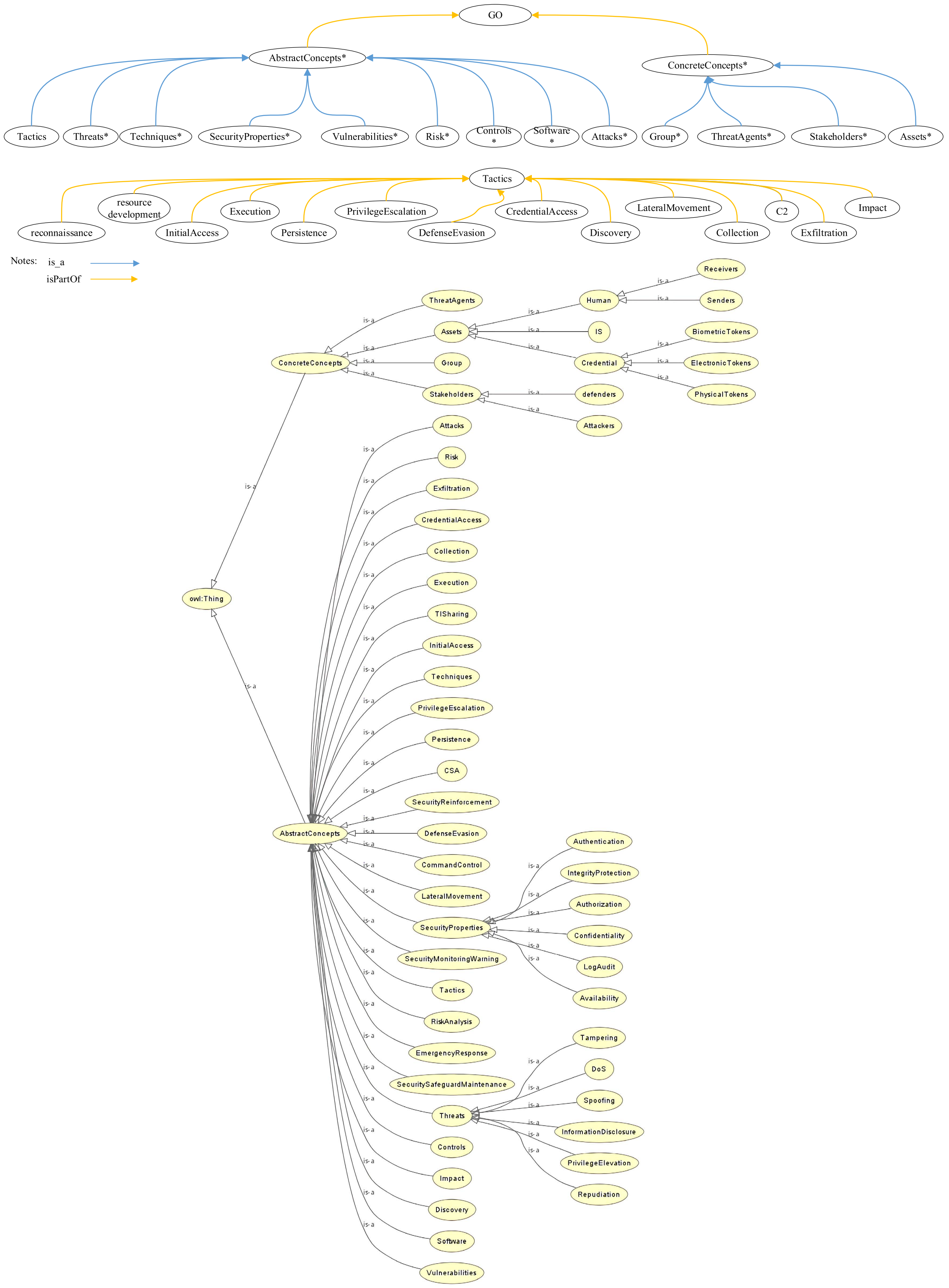


FIGURE 5. Portion of ISTGO concept class hierarchies and relationships.

2) ISTDO CONSTRUCTION

Definition 4.2.2 A domain is a collection of systems with similar or identical functions that address a specific domain problem. The systems exhibit variability in response to different application needs [59].

An ISTDO is used to describe knowledge in a specific domain. It describes domain concepts and association relationships, domain activities, and the characteristics and laws of the domain. These contents are obtained by instantiating the corresponding contents in the ISTGO.

TABLE 3. ISTGO concept space.

Relationships	Descriptions
value	Stakeholders→Assets
reduce	Controls→Risk
leadingto	Vulnerabilities→Risk
exploit	Threats→Vulnerabilities
increase	Threats→Risk
giveriseto	ThreatAgents→Threats
possess	Controls→Vulnerabilities
reduce	Controls→Vulnerabilities
expose	Vulnerabilities→Assets
attack	ThreatAgents→Assets
have	Assets→Vulnerabilities
threaten	Threats→Assets
implements	Software→Techniques
accomplish	Techniques→Tactics
uses	Group→Software
uses	Group→Techniques

a: ISTDO DEFINITION

The ISTDO is defined by

$$\text{ISTDO} =< \text{DomC, Op, Dp, P}^{\text{R}}, \text{P}^{\text{C}}, \text{DomH, DomR, DomM, A} >$$

where DomC, DomH, DomR, and A are similar to those defined in tuple (1), and the scope is limited to the domain layer. DomM is a full function from DomC to C, which maps each domain concept to an ISTGO concept class. This shows that each concept in DomC has a corresponding abstract concept in C. According to this function, the equivalence relation on DomC can be defined.

Definition 4.2.3 The equivalence relation $\equiv_{\text{domain}}$ on DomC can be defined by

$$\text{a} \equiv_{\text{domain}} \text{b iff Dommap (a)} = \text{Dommap (b)} = \text{t}$$

where a ∈ DomC, b ∈ DomC, t ∈ C. This equivalence relation can be denoted as: $[\text{t}]\equiv_{\text{domain}}$.

b: ISTDO CONCEPTS AND RELATIONSHIPS

The construction process of ISTDO is similar to that of ISTGO, so it is not repeated here.

The research field of this paper is limited to (1) the IS [60], including the conventional IS, cloud computing systems, and the IoT [60]; and (2) the ICS, which can be mapped with the concepts "Assets" and "IS" in the ISTGO. Due to the variety of concepts involved, in the concept selection stage, this paper uses the term "weighting technique" along with equation (2) [61]. A portion of the concept dictionary table is shown in Table 4. Figure 7 shows a portion of the hierarchy of the concept classes in the ISTDO.

$$AvgConceptScore = \frac{\sum ConceptScore}{\sum Concepts} \tag{2}$$

3) ISTAO CONSTRUCTION

a: ISTAO DEFINITION

The ISTAO is defined by

$$\text{ISTAO} =< \text{AppC, Op, Dp, P}^{\text{R}}, \text{P}^{\text{C}}, \text{AppH, AppR, AppM, A} >$$

The construction process of ISTAO is similar to that of ISTDO, so it is not repeated here.

Similarly, the equivalence relation with respect to AppC can be defined in terms of AppM.

Definition 4.2.4 The equivalence relation $\equiv_{\text{app}}$ on AppC can be defined by

$$\text{a} \equiv_{\text{app}} \text{b iff AppM(a)} = \text{AppM (b)} = \text{t}$$

where a ∈ AppC, b ∈ AppC, t ∈ C. This equivalence relation can be denoted as: $[\text{t}]\equiv_{\text{app}}$.

b: ISTAO CONCEPTS AND RELATIONSHIPS

The construction of ISTAO is based on engineering experience. The ISTAO concept classes originate from the protocol types of the application layer in the open system interconnection (OSI) model. Then, the mappings between the ISTAO concept classes and the ISTDO concept classes (IProtocols, CISProtocols, DISProtocols, and NProtocols) can be established. A portion of the hierarchy of the ISTAO concept classes is shown in Figure 8.

V. CASE STUDIES

The multiontology threat model constructed in this paper can play a guiding role for security personnel, especially for penetration testing. This section presents specific application cases to illustrate the effectiveness of this method.

A. IMPROVED PENETRATION TESTING PROCESS

Penetration testing comprehensively utilizes and evaluates a system by simulating the attacker's attack intentions and behaviors in a real environment to help discover potential attack chains. It has gradually become an important means of evaluating cybersecurity. James P. Anderson was one of the early pioneers in the development of penetration testing. In his report [62], he proposed a series of specific steps to penetrate and attack a system. The main idea is to detect the vulnerabilities first and then design the attack method. In this process, the weakness of the attack process and the way to fight against the threat can be found. This method is still used today. As a highly specialized activity, penetration testing requires testers to have a diversity of security knowledge and skills that must be accumulated through repeated practice. The systematic collection and modeling of fragmented knowledge can be achieved based on ontology methods. The results can be solidified to guide other security personnel; therefore, they are very useful.

Nickerson proposed the penetration testing execution standard (PTES) [63] in 2014. This standard divides the process of penetration testing into preengagement interactions, intelligence gathering, threat modeling, vulnerability analysis, exploitation, postexploitation, and reporting. Our paper integrates the ontology method into threat modeling and vulnerability analysis. The improved penetration testing process is shown in Figure 9. The whole penetration testing process is vulnerability oriented. In general, the penetration testing process consists of "target identification" and

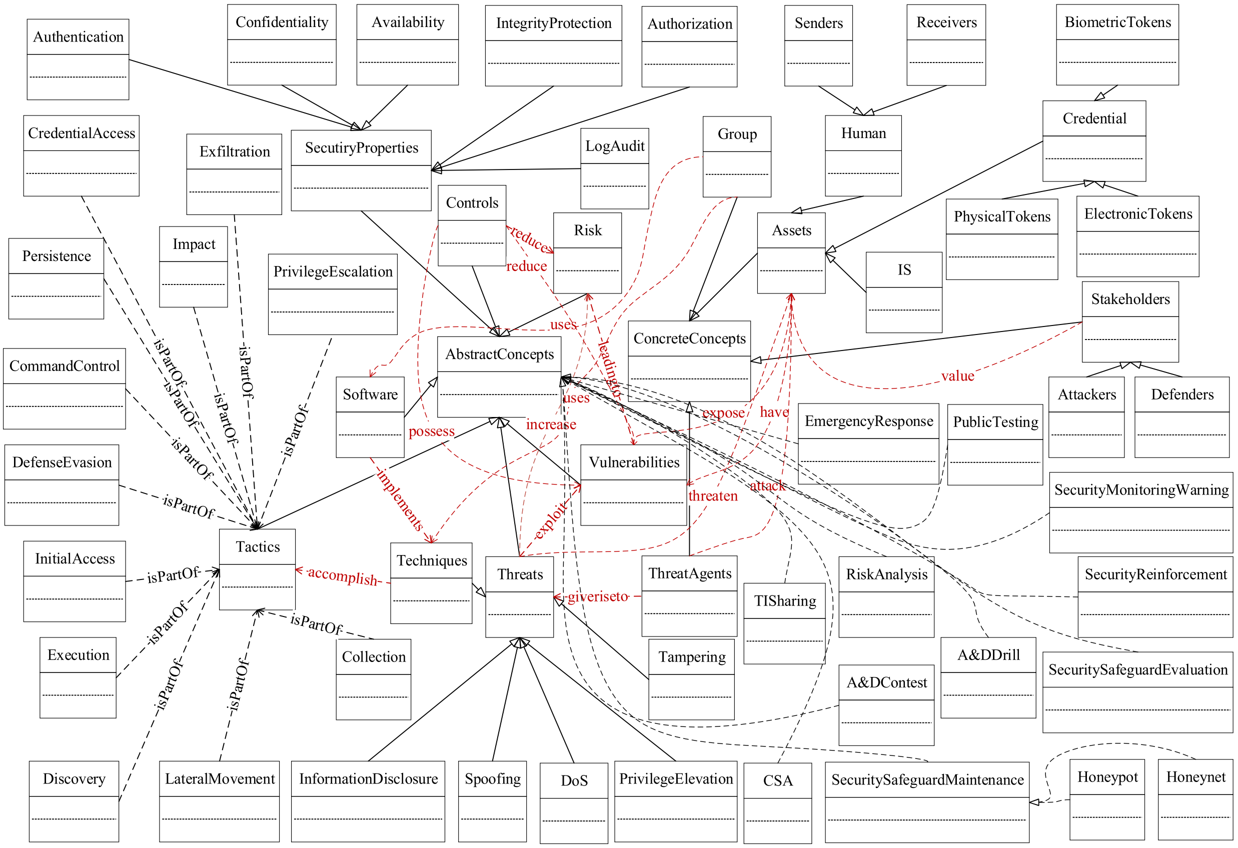


FIGURE 6. UML diagram representations of concepts and relationships of ISTGO.

TABLE 4. Concept dictionary table.

Concepts				
Conventional IS	CISHardware	CISComputer	CISCoreComponents	CISAccessory
PeripheralDevice	Scanner	PCamera	Microphone	CISNetworkEquipment
CISFirewall	CISRouter	CISSoftware	CISApplications	CISServer
CISUser	CISNetworkApplications	CISProtocols	CISOperatingSystem	CISWindows
CISUnix	CISMacOS	CloudMgmSystem	DISSoftware	DISOperatingSystem
DISWindows	DISUnix	DISMacOS	DISNetworkApplications	DISProtocols
DISApplications	DISUser	DISServer	DISHardware	DISNetworkEquipment
DISFirewall	DISRouter	DISComputer	DISCoreComponents	DISAccessory
IoTSystem	ApplicationLayer	InfoSecurityPlatform	MiddlewarePlatform	InfoProcessingPlatform
SupportPlatform	NetworkMgmPlatform	NetworkLayer	NHardware	TransLink
Terminal	Switch	NSoftware	SignallingSystem	NProtocols
SupportSystem	SenseLayer	SensorNetwork	SensorUnit	CommunicationUnit
ProcessingUnit	PowerUnit	SensingEquipment	GPS	RFIDLabel
SCamera	Reader	Sensors	-	

"attack". "Target identification" includes "interactive interfaces such as Shell" and "determines a destination host" in Figure 9. The "attack" consists of "threat modeling and vulnerability analysis based on multiontology", and "information collection", "vulnerability exploitation", "preliminary control of the host", and "privilege escalation, establishment of covert channel", which correspond to the steps of the ATT&CK model. The main tasks of the threat modeling and vulnerability analysis phase in the penetration testing process are to determine the risk types, attack points and attack surfaces and generate as many alternative attack scenarios as possible to improve the success rate of penetration testing. In addition, the dotted arrow in the figure indicates that when the final attack target has not been achieved, it is necessary to return to the "target confirmation" stage and implement the "attack" process again. The above process is repeated many times until the final attack target is achieved.

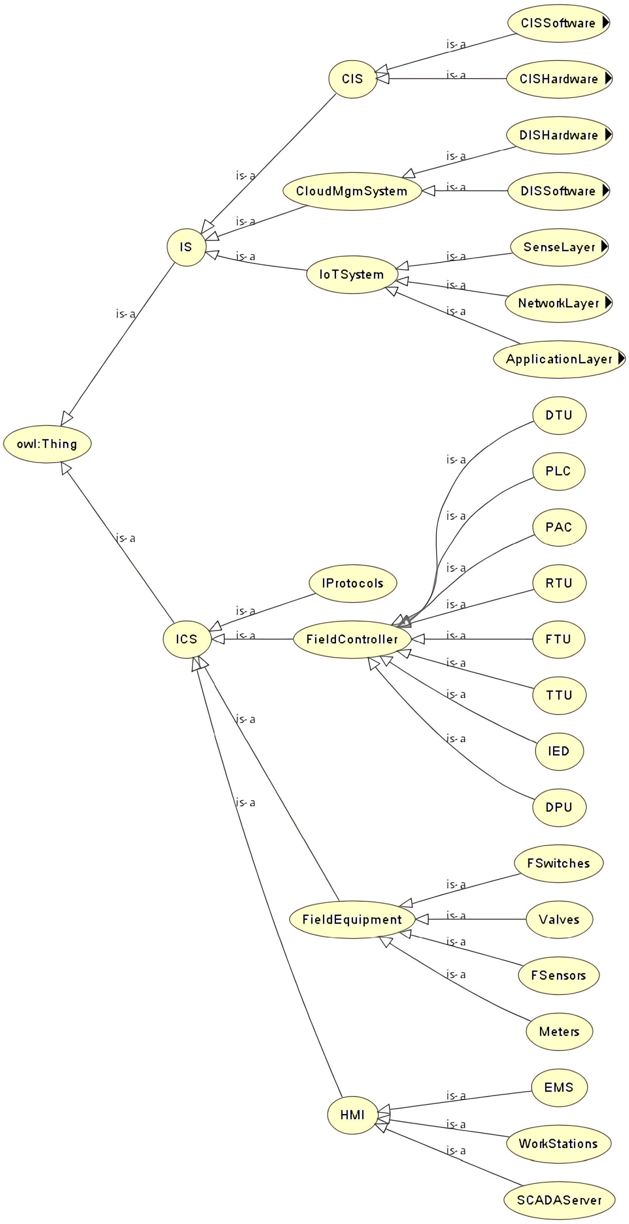


FIGURE 7. Portion of ISTDO concept class hierarchies.

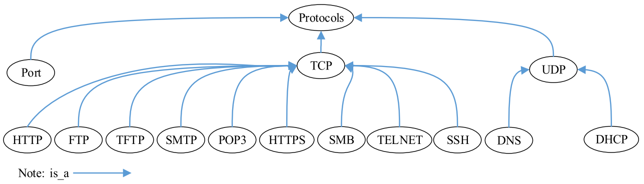


FIGURE 8. Portion of ISTAO concept class hierarchies.

B. THREAT ANALYSIS OF THE IoT SYSTEM

In section IV, the research field of this paper is limited to ISs, including conventional ISs, cloud computing systems, the IoT, and ICSs. This section first takes an IoT system as an example to describe the application process.

First, the concept classes ‘‘vulnerabilities’’, ‘‘risk’’ and ‘‘assets’’ in the ISTGO are mapped to the domain layer, and a portion of the UML diagram representations of the concepts and relationships of the ISTDO is shown in Figure 10. It is noted that DesignErrors represent the code design errors in Figure 10. However, in the IoT field, ‘‘design errors’’ should not be limited to the code level, but can also include errors in hardware design, sensor selection, etc.

Second, a risk analysis is performed, and the results are shown in Table 5.

Third, attack path-based threat analysis and adversarial attack and defense-based threat analysis of the IoT system are presented. The cases are discussed at the ISTAO layer.

1) ATTACK PATH-BASED THREAT ANALYSIS

Domestic IoT devices are increasingly being exploited in cybersecurity incidents [64]. Figure 11 shows the case of attack path-based threat analysis of an IoT system (a smart factory: FoT). The whole area is divided into two parts: (1) the local area, including a local production area and other areas, and (2) a nonlocal production area. The physical connection is shown in Figure 11, and both the nonlocal production area server and the local area master server contain vulnerabilities. Since the corresponding web service of the integrated Dell remote access controller (iDRAC) smart card has remote code execution (RCE) vulnerabilities (CVE-2021-21571, CVE-2021-21572, CVE-2021-21573, CVE-2021-21574, etc.), an attacker can exploit these vulnerabilities to gain control over the server’s hardware resources, thereby controlling the business platform running on the server and obtaining sensitive information such as data reporting interfaces and passwords. The nonlocal production area is compromised. Then, the attacker uses a similar method to compromise the local area master server and ultimately compromises the local production area. The complete attack path is as follows: compromise the nonlocal production area → compromise the local area master server → compromise the local production area. Based on the ATT&CK enterprise framework, the attack steps include ‘‘reconnaissance’’, ‘‘resource development’’, ‘‘initial access’’, ‘‘execution’’, ‘‘persistence’’, ‘‘privilege escalation’’, ‘‘defense evasion’’, ‘‘credential access’’, ‘‘discovery’’, ‘‘lateral movement’’, ‘‘collection’’, ‘‘C2’’, ‘‘exfiltration’’, and ‘‘impact’’. This paper takes the step of ‘‘reconnaissance’’ in the ATT&CK matrix as an example to provide screenshots of tool call commands, parameters, and the corresponding evidence.

- Collection of the exposed iDRAC-related asset information from the internet

This work is mainly performed for the collection of exposed iDRAC-related asset information from the internet in two ways: ‘‘supply chain penetration’’ and ‘‘exploitation of public vulnerabilities’’. Figure 12 shows a screenshot of the tool call command and parameters corresponding to the scene of Figure 11. Figure 13 shows the iDRAC control port exposed on the external network in the scene of Figure 11 (the corresponding evidence).

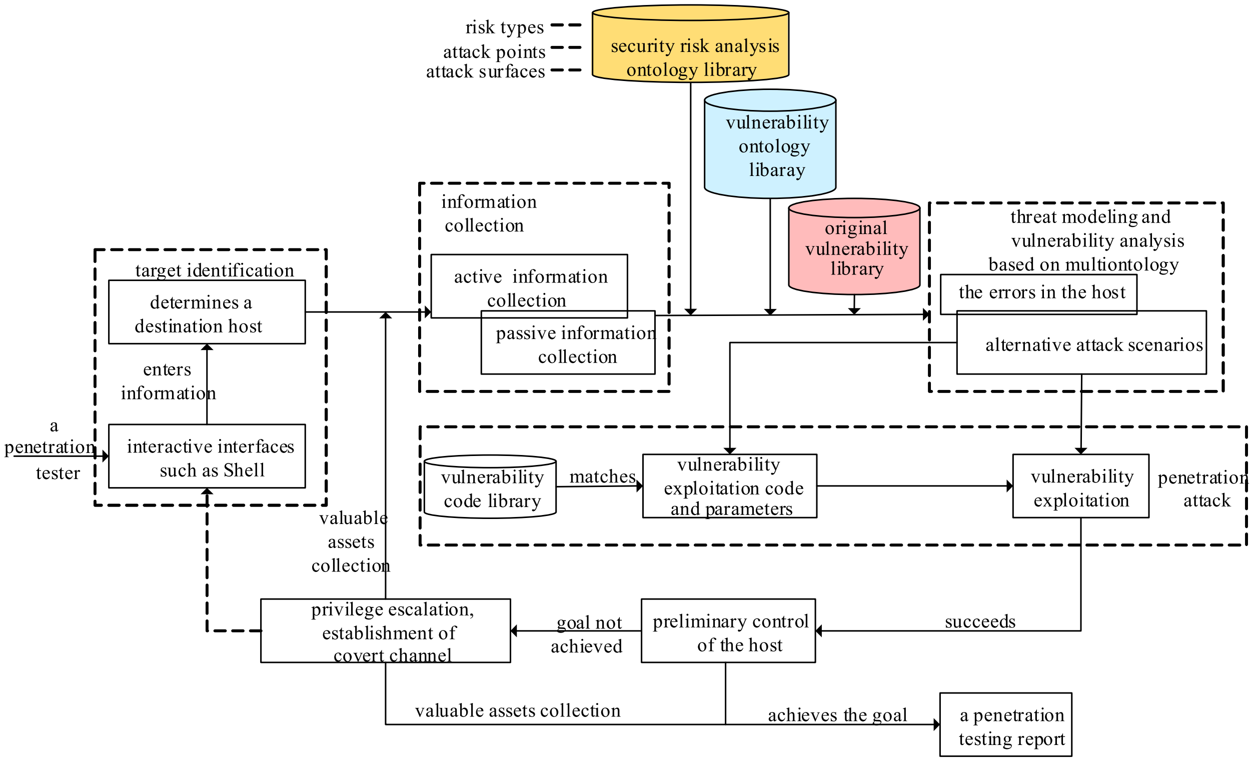

risk types
attack points
attack surfaces
security risk analysis ontology library
vulnerability ontology libaray
original vulnerability library
information collection
active information collection
passive information collection
threat modeling and vulnerability analysis based on multiontology
the errors in the host
alternative attack scenarios
target identification
determines a destination host
enters information
a penetration tester
interactive interfaces such as Shell
vulnerability code library
matches
vulnerability exploitation code and parameters
vulnerability exploitation
penetration attack
valuable assets collection
privilege escalation, establishment of covert channel
goal not achieved
preliminary control of the host
succeeds
valuable assets collection
achieves the goal
a penetration testing report

FIGURE 9. Improved penetration testing process.

2) ADVERSARIAL ATTACK AND DEFENSE-BASED THREAT ANALYSIS

Based on Figure 11, Figure 14 shows the case of adversarial attack and defense-based threat analysis for an IoT system (a smart factory: FoT). The local production area is a honeynet. From the attacker's perspective, an attacker carries out an effective attack according to the red line path in the figure (cameras → nonlocal production area → local area master server → local production area) and ultimately compromises the local production area. The attacker exploits the vulnerability CVE-2021-36260 to gain access from the camera. This is a command injection vulnerability in the web module of the camera. Due to the insufficient checking of input parameters, the unauthorized attacker can construct messages with malicious commands and send them to the affected devices, enabling remote command execution. The exploitation of this vulnerability does not require any user interaction. The attacker only needs to access the http/https server port (80/443) to exploit the vulnerability without the username, password, or other operations. The camera also cannot detect any login information. The vulnerability can affect IP cameras and network video recorder (NVR) devices, including the latest firmware released in June 2021 and the firmware released in 2006. From the defender's perspective, since the local production area is a honeynet, a defender has previously inserted a probe in this area. From the moment an attacker enters, his (her) behavior can be profiled. This paper takes several steps in the ATT&CK matrix as an example to provide screenshots of tool call commands, parameters, and the corresponding evidence.

- Collection of information from the internet (camera platform exposure + weak password)

a: CAMERA PLATFORM EXPOSURE

A screenshot of the tool call command and parameters corresponding to the scene of Figure 14 is shown in Figure 15. This work corresponds to the step of "reconnaissance" in the ATT&CK matrix.

b: WEAK PASSWORD

Figure 16 shows that the attacker has obtained control over the camera platform. This corresponds to the "C2" step in the ATT&CK matrix.

After the attacker gains control of the devices on the internet, he (she) continues to attack the intranet by the step of "lateral movement".

Collection of Information From the Intranet: A screenshot of the tool call command and parameters corresponding to the scene of Figure 14 is shown in Figure 17. This work corresponds to the step of "reconnaissance" in the ATT&CK matrix.

Figure 18 shows that the attacker has obtained control over the server console in the intranet. This work corresponds to steps "C2" and "credential access" in the ATT&CK matrix.

Compromise of the Intranet Assets: This work is mainly performed to attack the intranet through an external

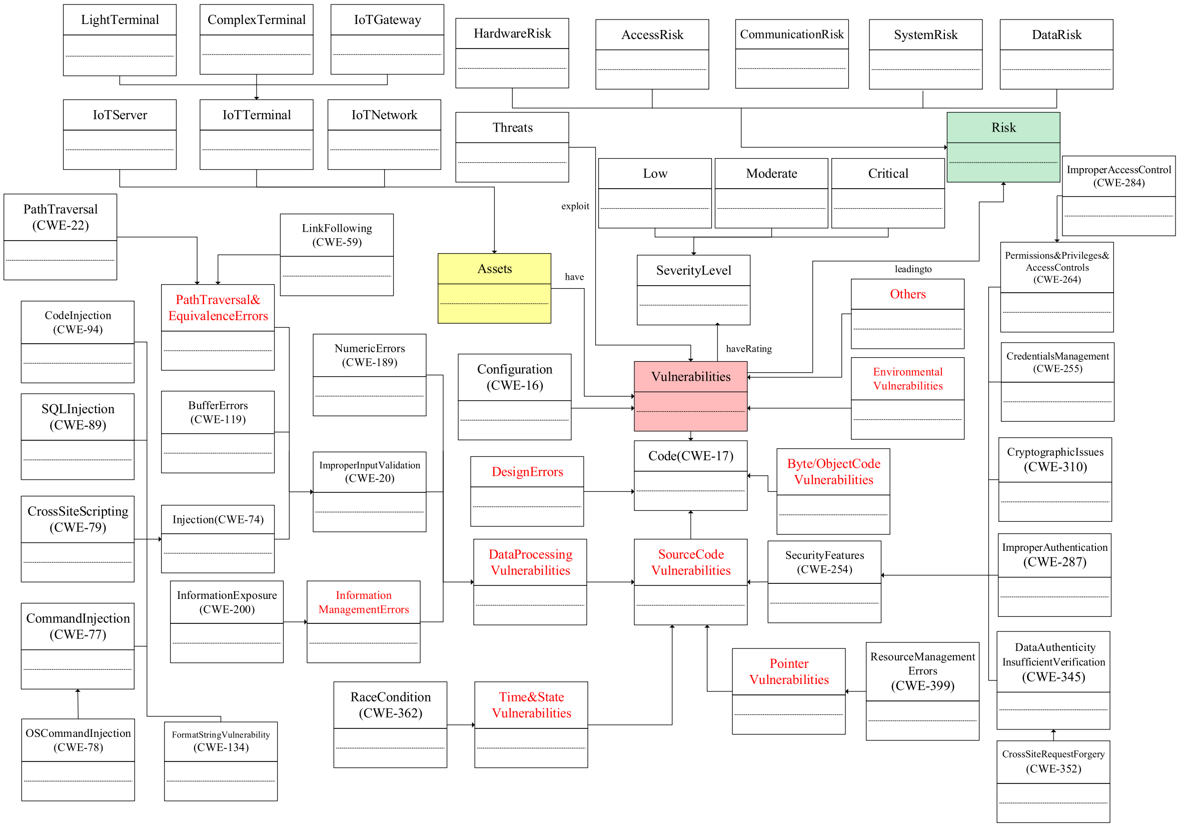

LightTerminal
ComplexTerminal
IoTGateway
HardwareRisk
AccessRisk
CommunicationRisk
SystemRisk
DataRisk
IoTServer
IoTTerminal
IoTNetwork
Threats
Risk
Low
Moderate
Critical
ImproperAccessControl (CWE-284)
PathTraversal (CWE-22)
LinkFollowing (CWE-59)
exploit
Assets
have
SeverityLevel
leadingto
Permissions&Privileges& AccessControls (CWE-264)
Others
CodeInjection (CWE-94)
PathTraversal& EquivalenceErrors
NumericErrors (CWE-189)
Configuration (CWE-16)
haveRating
Vulnerabilities
Environmental Vulnerabilities
CredentialsManagement (CWE-255)
SQLInjection (CWE-89)
BufferErrors (CWE-119)
ImproperInputValidation (CWE-20)
DesignErrors
Code(CWE-17)
Byte/ObjectCode Vulnerabilities
CryptographicIssues (CWE-310)
CrossSiteScripting (CWE-79)
Injection(CWE-74)
DataProcessing Vulnerabilities
SourceCode Vulnerabilities
SecurityFeatures (CWE-254)
ImproperAuthentication (CWE-287)
CommandInjection (CWE-77)
InformationExposure (CWE-200)
Information ManagementErrors
Pointer Vulnerabilities
ResourceManagement Errors (CWE-399)
DataAuthenticity InsufficientVerification (CWE-345)
RaceCondition (CWE-362)
Time&State Vulnerabilities
OSCommandInjection (CWE-78)
FormatStringVulnerability (CWE-134)
CrossSiteRequestForgery (CWE-352)

FIGURE 10. Portion of UML diagram representations of concepts and relationships of the ISTDO (IoT System).

TABLE 5. Security risk analysis of IoT terminal equipments.

Risk types	Attack points	Attack surfaces
hardware risk	deployment environment	Theft, sabotage by attackers; Harsh temperature or humidity environments; Signal interference or shielding.
	power supply capabilities	Power supply reliability is not sufficient to support normal operation.
	equipment interfaces	There are idle external interfaces (such as JTAG, UART, and TTL), and attackers can dump/reprogram flash.
access risk	access authentication	Failure of authentication mechanism based on MAC or network identity.
	access control	The terminal access control policy is not strict, resulting in unauthorized access or control.
communication risk	confidentiality	Clear text transfer of sensitive information; SSL/TLS unavailable or misconfigured.
	integrity	Due to the lack of integrity verification, attackers can tamper and replay after intercepting the data.
system risk	identification	Identity assumption; A weak or missing password for authentication.
	permission control	Read, write, and executable privileges for different users without strictly limiting the data range
	firmware	A hidden backdoor in the firmware;
	upgrade	Issuance of malicious update instructions; Firmware not tested for security; Inability to verify firmware legality or authenticity; Firmware does not support the upgrade function.
data risk	data confidentiality	Leakage of sensitive or private information stored locally by the terminal.
	data integrity	The data source cannot be traced; The data source has been maliciously tampered with and cannot be checked.
	data availability	The data acquired or monitored by the terminal exceed the regulations during the transfer process, resulting in failure.

springboard and follow up to attack the deeper intranet across network segments. Figure 19 shows the complete intranet attack path.

Fourth, a penetration testing report is presented. The template of penetration testing report is shown in table 6.

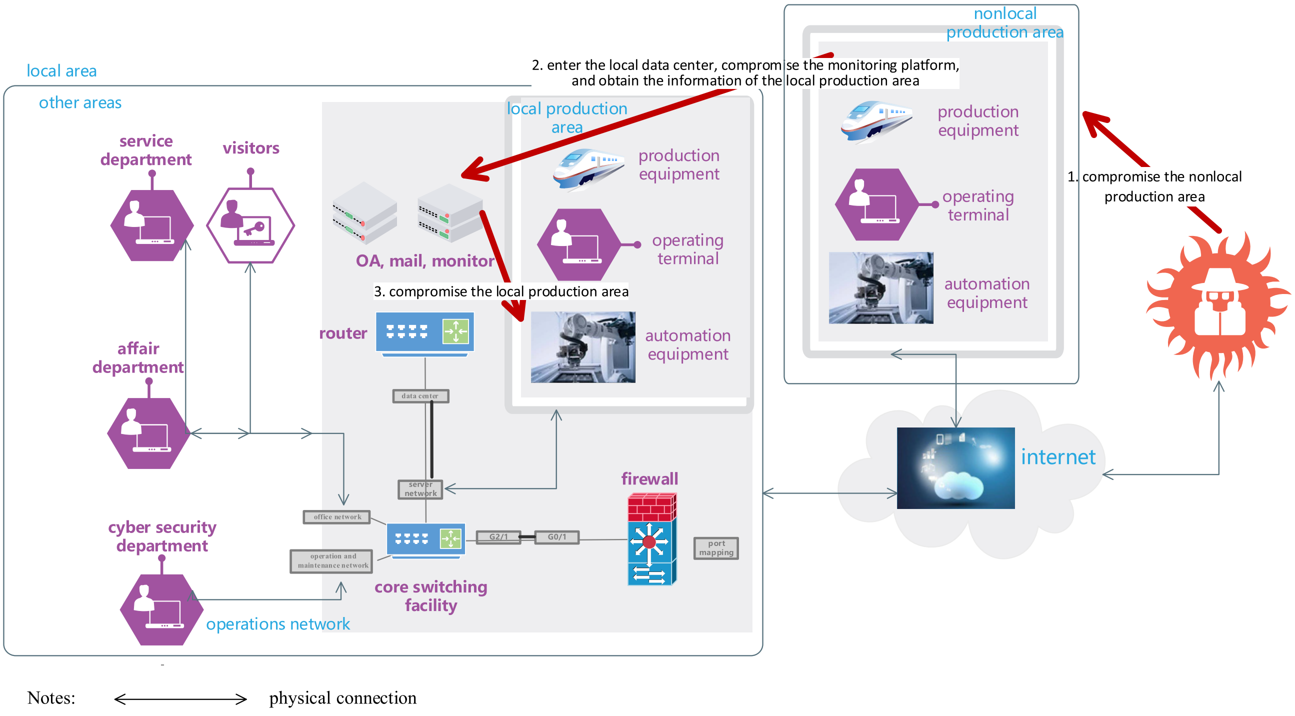
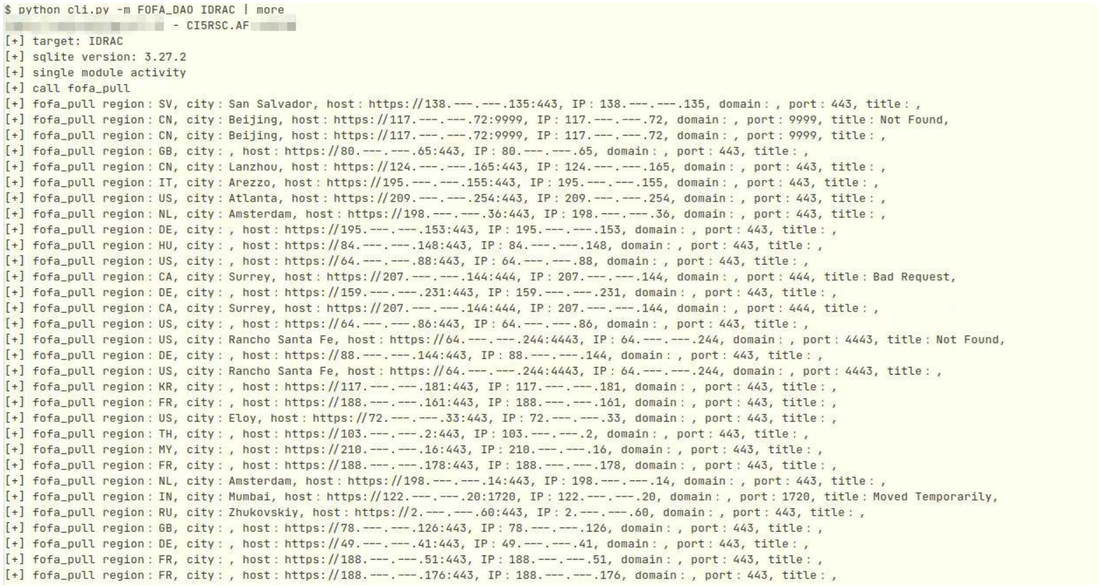


FIGURE 11. Attack path-based threat analysis of an IoT system.



```
$ python cli.py -m FOFA_DAO IDRAC | more
[redacted] - CI5RSC.AF [redacted]
[+] target: IDRAC
[+] sqlite version: 3.27.2
[+] single module activity
[+] call fofa_pull
[+] fofa_pull region : SV, city : San Salvador, host : https://138.---.---.135:443, IP : 138.---.---.135, domain : , port : 443, title : ,
[+] fofa_pull region : CN, city : Beijing, host : https://117.---.---.72:9999, IP : 117.---.---.72, domain : , port : 9999, title : Not Found,
[+] fofa_pull region : CN, city : Beijing, host : https://117.---.---.72:9999, IP : 117.---.---.72, domain : , port : 9999, title : ,
[+] fofa_pull region : GB, city : , host : https://80.---.---.65:443, IP : 80.---.---.65, domain : , port : 443, title : ,
[+] fofa_pull region : CN, city : Lanzhou, host : https://124.---.---.165:443, IP : 124.---.---.165, domain : , port : 443, title : ,
[+] fofa_pull region : IT, city : Arezzo, host : https://195.---.---.155:443, IP : 195.---.---.155, domain : , port : 443, title : ,
[+] fofa_pull region : US, city : Atlanta, host : https://209.---.---.254:443, IP : 209.---.---.254, domain : , port : 443, title : ,
[+] fofa_pull region : NL, city : Amsterdam, host : https://198.---.---.36:443, IP : 198.---.---.36, domain : , port : 443, title : ,
[+] fofa_pull region : DE, city : , host : https://195.---.---.153:443, IP : 195.---.---.153, domain : , port : 443, title : ,
[+] fofa_pull region : HU, city : , host : https://84.---.---.148:443, IP : 84.---.---.148, domain : , port : 443, title : ,
[+] fofa_pull region : US, city : , host : https://64.---.---.88:443, IP : 64.---.---.88, domain : , port : 443, title : ,
[+] fofa_pull region : CA, city : Surrey, host : https://207.---.---.144:444, IP : 207.---.---.144, domain : , port : 444, title : Bad Request,
[+] fofa_pull region : DE, city : , host : https://159.---.---.231:443, IP : 159.---.---.231, domain : , port : 443, title : ,
[+] fofa_pull region : CA, city : Surrey, host : https://207.---.---.144:444, IP : 207.---.---.144, domain : , port : 444, title : ,
[+] fofa_pull region : US, city : , host : https://64.---.---.86:443, IP : 64.---.---.86, domain : , port : 443, title : ,
[+] fofa_pull region : US, city : Rancho Santa Fe, host : https://64.---.---.244:4443, IP : 64.---.---.244, domain : , port : 4443, title : Not Found,
[+] fofa_pull region : DE, city : , host : https://88.---.---.144:443, IP : 88.---.---.144, domain : , port : 443, title : ,
[+] fofa_pull region : US, city : Rancho Santa Fe, host : https://64.---.---.244:4443, IP : 64.---.---.244, domain : , port : 4443, title : ,
[+] fofa_pull region : KR, city : , host : https://117.---.---.181:443, IP : 117.---.---.181, domain : , port : 443, title : ,
[+] fofa_pull region : FR, city : , host : https://188.---.---.161:443, IP : 188.---.---.161, domain : , port : 443, title : ,
[+] fofa_pull region : US, city : Eloy, host : https://72.---.---.33:443, IP : 72.---.---.33, domain : , port : 443, title : ,
[+] fofa_pull region : TH, city : , host : https://103.---.---.2:443, IP : 103.---.---.2, domain : , port : 443, title : ,
[+] fofa_pull region : MY, city : , host : https://210.---.---.16:443, IP : 210.---.---.16, domain : , port : 443, title : ,
[+] fofa_pull region : FR, city : , host : https://188.---.---.178:443, IP : 188.---.---.178, domain : , port : 443, title : ,
[+] fofa_pull region : NL, city : Amsterdam, host : https://198.---.---.14:443, IP : 198.---.---.14, domain : , port : 443, title : ,
[+] fofa_pull region : IN, city : Mumbai, host : https://122.---.---.20:1720, IP : 122.---.---.20, domain : , port : 1720, title : Moved Temporarily,
[+] fofa_pull region : RU, city : Zhukovskiy, host : https://2.---.---.60:443, IP : 2.---.---.60, domain : , port : 443, title : ,
[+] fofa_pull region : GB, city : , host : https://78.---.---.126:443, IP : 78.---.---.126, domain : , port : 443, title : ,
[+] fofa_pull region : DE, city : , host : https://49.---.---.41:443, IP : 49.---.---.41, domain : , port : 443, title : ,
[+] fofa_pull region : FR, city : , host : https://188.---.---.51:443, IP : 188.---.---.51, domain : , port : 443, title : ,
[+] fofa_pull region : FR, city : , host : https://188.---.---.176:443, IP : 188.---.---.176, domain : , port : 443, title : ,
```

FIGURE 12. Tool call command and parameters corresponding to the scene of Figure 11.

The rest of this section presents two more examples of a conventional IS and an ICS to describe the threat analysis process. The cases are discussed at the ISTAO layer.

C. ATTACK PATH-BASED THREAT ANALYSIS OF THE CONVENTIONAL IS

Figure 20 shows the case of attack path-based threat analysis of a conventional IS (the intranet of small and medium enterprises (SME)). The whole area is divided into three parts: a demilitarized zone (DMZ), an intranet server zone, and an intranet personal computer (PC) zone. The servers in the DMZ, the intranet zone server and the intranet zone PC are connected by a router. In addition, they are logically isolated. DMZ server A does not contain vulnerabilities; DMZ server B, the intranet zone server and the intranet zone PC all contain vulnerabilities. An attacker first scans

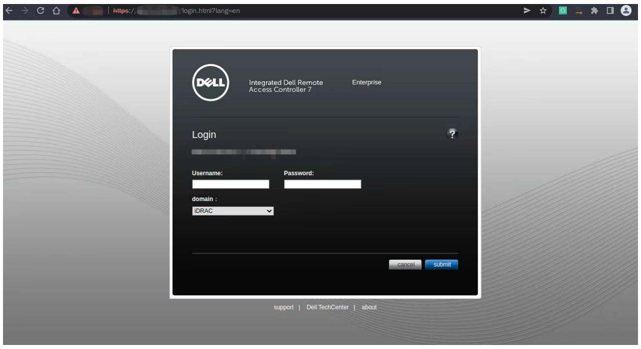

https://[redacted]/login.html?lang=en
DELL
Integrated Dell Remote Access Controller 7
Enterprise
Login
Username:
Password:
domain :
iDRAC
cancel
submit
support | Dell TechCenter | about

FIGURE 13. Exposed iDRAC control port.

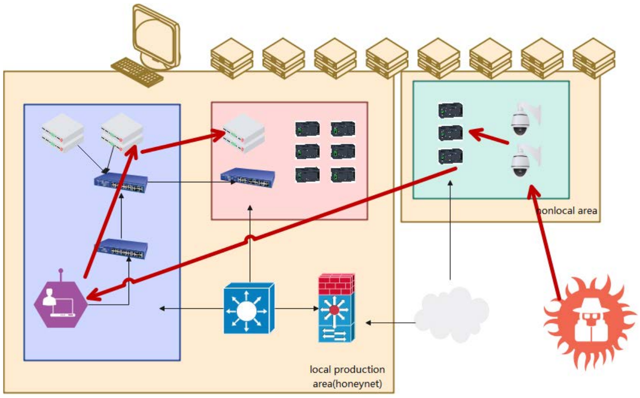
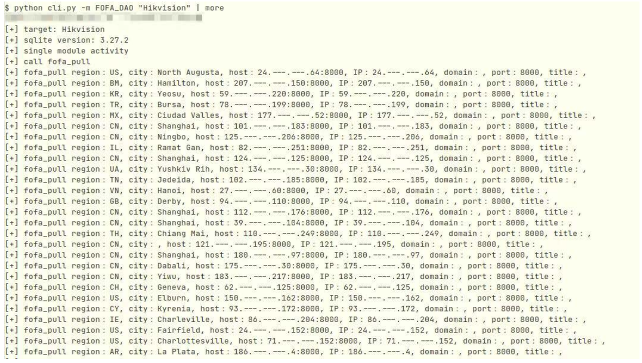


FIGURE 14. Adversarial attack and defense-based threat analysis of an IoT system.



```
$ python cli.py -m FOFA_DAO "Hikvision" | more
[redacted]
[+] target: Hikvision
[+] sqlite version: 3.27.2
[+] single module activity
[+] call fofa_pull
[+] fofa_pull region : US, city : North Augusta, host : 24.---.---.64:8000, IP : 24.---.---.64, domain : , port : 8000, title : ,
[+] fofa_pull region : BM, city : Hamilton, host : 207.---.---.150:8000, IP : 207.---.---.150, domain : , port : 8000, title : ,
[+] fofa_pull region : KR, city : Yeosu, host : 59.---.---.220:8000, IP : 59.---.---.220, domain : , port : 8000, title : ,
[+] fofa_pull region : TR, city : Bursa, host : 78.---.---.199:8000, IP : 78.---.---.199, domain : , port : 8000, title : ,
[+] fofa_pull region : MX, city : Ciudad Valles, host : 177.---.---.52:8000, IP : 177.---.---.52, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : Shanghai, host : 101.---.---.183:8000, IP : 101.---.---.183, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : Ningbo, host : 125.---.---.206:8000, IP : 125.---.---.206, domain : , port : 8000, title : ,
[+] fofa_pull region : IL, city : Ramat Gan, host : 82.---.---.251:8000, IP : 82.---.---.251, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : Shanghai, host : 124.---.---.125:8000, IP : 124.---.---.125, domain : , port : 8000, title : ,
[+] fofa_pull region : UA, city : Yushkiv Rih, host : 134.---.---.30:8000, IP : 134.---.---.30, domain : , port : 8000, title : ,
[+] fofa_pull region : TN, city : Jedeida, host : 102.---.---.185:8000, IP : 102.---.---.185, domain : , port : 8000, title : ,
[+] fofa_pull region : VN, city : Hanoi, host : 27.---.---.60:8000, IP : 27.---.---.60, domain : , port : 8000, title : ,
[+] fofa_pull region : GB, city : Derby, host : 94.---.---.110:8000, IP : 94.---.---.110, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : Shanghai, host : 112.---.---.176:8000, IP : 112.---.---.176, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : Shanghai, host : 39.---.---.104:8000, IP : 39.---.---.104, domain : , port : 8000, title : ,
[+] fofa_pull region : TH, city : Chiang Mai, host : 110.---.---.249:8000, IP : 110.---.---.249, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : , host : 121.---.---.195:8000, IP : 121.---.---.195, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : Shanghai, host : 180.---.---.97:8000, IP : 180.---.---.97, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : Dabali, host : 175.---.---.30:8000, IP : 175.---.---.30, domain : , port : 8000, title : ,
[+] fofa_pull region : CN, city : Yiwu, host : 183.---.---.217:8000, IP : 183.---.---.217, domain : , port : 8000, title : ,
[+] fofa_pull region : CH, city : Geneva, host : 62.---.---.125:8000, IP : 62.---.---.125, domain : , port : 8000, title : ,
[+] fofa_pull region : US, city : Elburn, host : 150.---.---.162:8000, IP : 150.---.---.162, domain : , port : 8000, title : ,
[+] fofa_pull region : CY, city : Kyrenia, host : 93.---.---.172:8000, IP : 93.---.---.172, domain : , port : 8000, title : ,
[+] fofa_pull region : IE, city : Charleville, host : 86.---.---.204:8000, IP : 86.---.---.204, domain : , port : 8000, title : ,
[+] fofa_pull region : US, city : Fairfield, host : 24.---.---.152:8000, IP : 24.---.---.152, domain : , port : 8000, title : ,
[+] fofa_pull region : US, city : Charlottesville, host : 71.---.---.152:8000, IP : 71.---.---.152, domain : , port : 8000, title : ,
[+] fofa_pull region : AR, city : La Plata, host : 186.---.---.4:8000, IP : 186.---.---.4, domain : , port : 8000, title : ,
```

FIGURE 15. Tool call command and parameters corresponding to the scene of Figure 14 (internet).

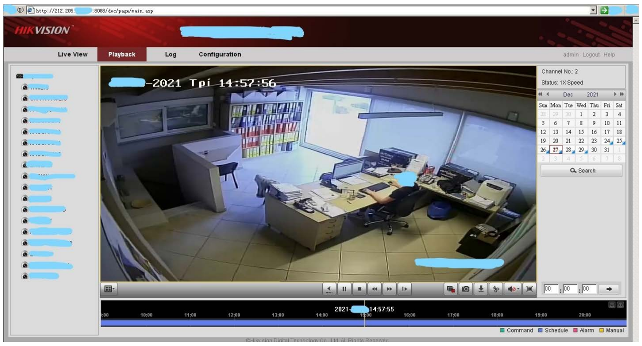
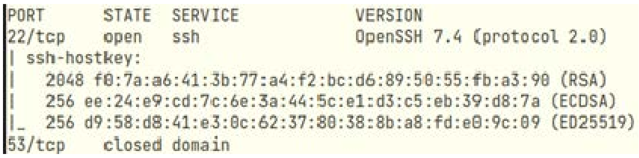
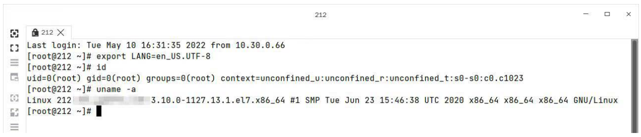


FIGURE 16. Evidence of "C2" for scene of Figure 14.



```
PORT      STATE  SERVICE               VERSION
22/tcp    open   ssh                   OpenSSH 7.4 (protocol 2.0)
| ssh-hostkey:
|   2048 f0:7a:a6:41:3b:77:a4:f2:bc:d6:89:50:55:fb:a3:90 (RSA)
|   256 ee:24:e9:cd:7c:6e:3a:44:5c:e1:d3:c5:eb:39:d8:7a (ECDSA)
|_  256 d9:58:d8:41:e3:0c:62:37:80:38:8b:a8:fd:e0:9c:09 (ED25519)
53/tcp    closed domain
```

FIGURE 17. Tool call command and parameters corresponding to the scene of Figure 14 (intranet).



```
212
Last login: Tue May 10 16:31:35 2022 from 10.30.0.66
[root@212 ~]# export LANG=en_US.UTF-8
[root@212 ~]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@212 ~]# uname -a
Linux 212 [redacted] 3.10.0-1127.13.1.el7.x86_64 #1 SMP Tue Jun 23 15:46:38 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
[root@212 ~]#
```

FIGURE 18. Evidence of "C2" and "credential access" for scene of Figure 14.

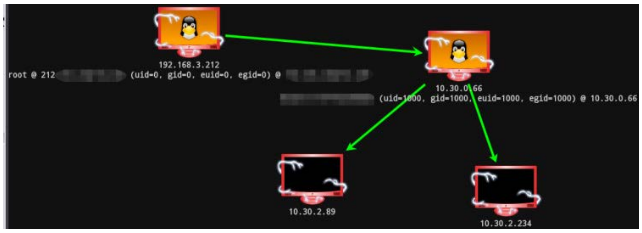

192.168.3.212
root @ 212 [redacted] (uid=0, gid=0, euid=0, egid=0) @ [redacted]
10.30.0.66
[redacted] (uid=1000, gid=1000, euid=1000, egid=1000) @ 10.30.0.66
10.30.2.89
10.30.2.234

FIGURE 19. Complete intranet attack path for scene of Figure 14.

all exposed ports in the DMZ: port 80 mapped by server A, and port 22 and port 21 mapped by server B. Since DMZ server B contains vulnerabilities, the attacker compromises DMZ server B and obtains a springboard for intranet access. Then, the attacker uses this springboard to gain access to the intranet zone PC and compromises the intranet zone PC. Similarly, the intranet zone server is compromised. At this point, the attacker's information collection from the intranet (password list, network structure, system software and hardware architecture, deployment scheme, etc.) has reached a relatively complete level; thus, DMZ server A is ultimately compromised, and actual control over the various devices in the intranet is obtained. It should be noted that when DMZ server B is compromised, the attempted attack on DMZ server A fails due to insufficient information collection. The complete attack path is DMZ server B → intranet zone PC → intranet zone server → DMZ server A.

The above process can be extended to the case in which an area contains n ($n \in N$) servers and m ($m \in N$) PCs. The attack path-based threat analysis shows that as the attack continues to deepen, the attacker obtains increasingly more information. This leads to an upward trend in both the threat value and information integrity. This is because, for an information system (a target system), when the attacker collects a certain level (threshold) of its key information (such as passwords and vulnerable component versions), the security line of defense of the system will be broken down. The

TABLE 6. Penetration testing report template.

OTT(object to test)	Issue No.	Test time	Tester name
URL/IP			
Vulnerability type			
Vulnerability description			
Vulnerability level	□critical	□high □medium	□low
Risk assessment			
Corrective actions			
Corrective results			
Vulnerability proof			

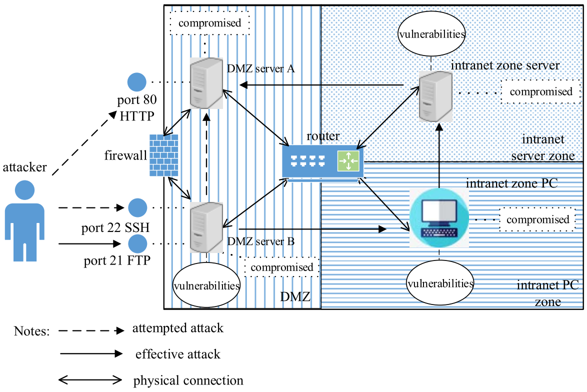


FIGURE 20. Attack path-based threat analysis of a conventional IS.

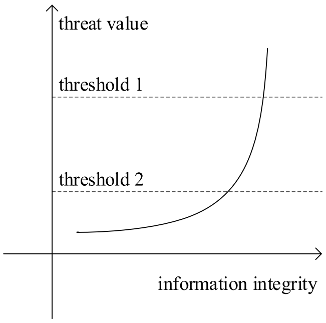


FIGURE 21. Relationship between threat value and information integrity.

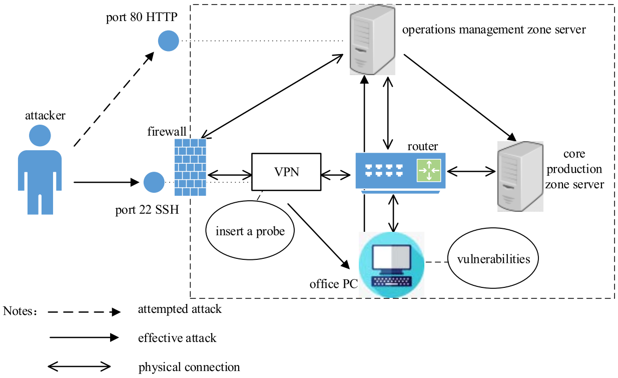
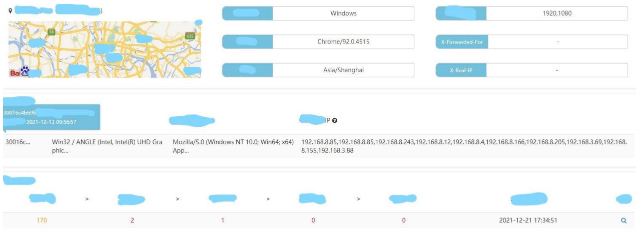


FIGURE 22. Adversarial attack and defense-based threat analysis of an ICS.

two dashed lines in the figure represent different thresholds. The higher the position of the dotted line is, the higher the



[redacted] [redacted])

[redacted]	Windows	[redacted]	1920,1080
[redacted]	Chrome/92.0.4515	X-Forwarded-For	-
[redacted]	Asia/Shanghai	X-Real-IP	-

[redacted]
30016c4b69b[redacted]
[redacted] 2021-12-13 09:56:57

	[redacted]	[redacted]IP	
30016c...	Win32 / ANGLE (Intel, Intel(R) UHD Graphic...	Mozilla/5.0 (Windows NT 10.0; Win64; x64) App...	192.168.8.85,192.168.8.85,192.168.8.243,192.168.8.12,192.168.8.4,192.168.8.166,192.168.8.205,192.168.3.69,192.168.8.155,192.168.3.88

[redacted]

[redacted] >	[redacted] >	[redacted] >	[redacted] >	[redacted]	[redacted]	[redacted]
170	2	1	0	0	2021-12-21 17:34:51	

FIGURE 23. Attacker tracing.

level of security protection. The above process is shown in Figure 21, which is also based on the attacker's perspective. It is noted that the figure is based on engineering experience and only shows a general trend, not a strictly mathematical representation.

D. ADVERSARIAL ATTACK AND DEFENSE-BASED THREAT ANALYSIS OF THE ICS

Figure 22 shows the case of adversarial attack and defense-based threat analysis of an ICS. The dashed box is a honeynet composed of multiple honeypot nodes. From the attacker's perspective, an attacker carries out an effective attack according to the path shown by the "⟶ " line in the figure (obtain a virtual private network (VPN) access point to enter the intranet → compromise the office PC → compromise the server in the operations management zone → obtain control of the industrial control equipment in the core production zone) and ultimately compromise the ICS production zone. From the defender's perspective, since this is a honeynet, a defender has previously inserted a probe in the VPN. From the moment an attacker enters, the attacker behavior can be profiled, and a comprehensive grasp of his (her) attack activities and attack capabilities can be obtained. Furthermore, the attacker characteristics, including social fingerprints and attack fingerprints, can be obtained, and attacker tracing can be conducted as shown in Figure 23.

VI. CONCLUSION

This paper focused on the problems of ambiguity, heterogeneity, and incompleteness of the existing threat models. By constructing an ISTMOF and the corresponding ontologies (an ISTGO, an ISTDO and an ISTAO) based on domain knowledge (attack and defense knowledge), engineering experience, and industry standards (ISO/IEC 27032), the above problems can be solved. In this paper, all steps of building the ISTMOF and the corresponding ontologies were mentioned. The case studies showed that systematic threat modeling using the multiontology method can be used not only for attack path-based threat analysis, but also for adversarial attack and defense-based threat analysis. This method can help detect security issues and effectively guide security personnel.

The research results in this paper have wider applications for future work. For example, (1) complete security requirements identification can be achieved using the above threat scenario analysis: identify the scenarios of business systems → identify the security requirements corresponding to the scenarios → build a knowledge base of security requirements → when encountering unidentified scenarios, identify threats using threat modeling, and then derive supplementary security requirements → integrate the supplementary security requirements into overall security requirements; (2) threat scenario analysis can be carried out during the security design phase: referring to the high-level design or low-level design, accurate resource identification, application architecture understanding, and fast application decomposition can be realized by interviews with architects and developers → produce detailed data flow diagrams to identify existing security threats → cybersecurity risk assessment. These two points also represent our future research directions.

REFERENCES

[1] *Information Technology—Security Techniques—Information Security Management Systems—Overview and Vocabulary*, Standard ISO/IEC 27000, 2018.

[2] *Information Technology—Security Techniques—Guidelines for Cybersecurity*, Standard ISO/IEC 27032, 2012.

[3] MITRE. *ATT&CK@*. [Online]. Available: https://attack.mitre.org/

[4] G. Lei, S. Ruibin, and T. Yu, "Research on key technologies of ontology based threat modeling for cyber range," *J. CAEIT*, vol. 15, no. 12, pp. 1139–1144, Dec. 2020.

[5] B. Schneier, "Attack trees: Modeling security threats," *Dr Dobb's J.*, vol. 24, no. 12, pp. 21–29, 1999.

[6] Microsoft Security. *STRIDE Chart*. [Online]. Available: https://www.microsoft.com/security/blog/2007/09/11/stride-chart/

[7] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "MITRE ATT&CK: Design and philosophy," MITRE Corp., McLean, VA, USA, Tech. Rep. MP180360R1, Mar. 2020.

[8] MITRE. *CAPEC*. [Online]. Available: https://capec.mitre.org/

[9] MITRE. *CWE*. [Online]. Available: http://cwe.mitre.org/

[10] Lockheed Martin. *Cyber Kill Chain*. [Online]. Available: https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

[11] M. Donner, "Toward a security ontology," *IEEE Secur. Privacy*, vol. 1, no. 3, pp. 6–7, May/Jun. 2003.

[12] M. Schumacher, *Security Engineering With Patterns: Origins, Theoretical Models, and New Applications* (Lecture Notes in Computer Science), vol. 2754. Berlin, Germany: Springer-Verlag, 2003, pp. 87–96.

[13] S. Fenz, G. Goluch, A. Ekelhar, B. Riedl, and E. Weippl, "Information security fortification by ontological mapping of the ISO/IEC 27001 standard," in *Proc. 13th IEEE Pacific Rim Int. Symp. Dependable Comput. (PRDC)*, Melbourne, VIC, Australia, Dec. 2007, pp. 381–388.

[14] A. Ekelhart, S. Fenz, M. D. Klemen, and E. R. Weippl, "Security ontology: Simulating threats to corporate assets," in *Proc. Inf. Syst. Secur. (ICISS)*, Kolkata, India, 2006, pp. 249–259.

[15] A. Ekelhart, S. Fenz, M. Klemen, and E. Weippl, "Security ontologies: Improving quantitative risk analysis," in *Proc. 40th HICSS*, Waikoloa, HI, USA, 2007, pp. 156–162.

[16] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Trans. Depend. Sec. Comput.*, vol. 1, no. 1, pp. 11–33, Jan./Mar. 2004.

[17] A. Herzog, N. Shahmehri, and C. Duma, "An ontology of information security," *Int. J. Inf. Secur. Privacy*, vol. 1, no. 4, pp. 1–23, Oct. 2007.

[18] M. E. Whitman and H. J. Mattord, *Principles of Information Security*, 2nd ed. Boston, MA, USA: Thomson Course Technology, 2005.

[19] C. Blanco, J. Lasheras, R. Valencia-Garíc, E. Fern, A. Toval, and M. Piattini, "A systematic review and comparison of security ontologies," in *Proc. 3rd Int. Conf. Availability, Rel. Secur.*, Barcelona, Spain, Mar. 2008, pp. 813–820.

[20] T. Pereira and H. Santos, "An ontology based approach to information security," in *Proc. MTSR*, Berlin, Germany, 2009, pp. 183–192.

[21] C. Blanco, J. Lasheras, E. Fernández-Medina, R. Valencia-García, and A. Toval, "Basis for an integrated security ontology according to a systematic review of existing proposals," *Comput. Standards Interfaces*, vol. 33, no. 4, pp. 372–388, Jun. 2011.

[22] J.-B. Gao, B.-W. Zhang, X.-H. Chen, and Z. Luo, "Ontology-based model of network and computer attacks for security assessment," *J. Shanghai Jiaotong Univ.*, vol. 18, no. 5, pp. 554–562, Oct. 2013.

[23] A. Gyrard, C. Bonnet, and K. Boudaoud, "The STAC (security toolbox: Attacks & countermeasures) ontology," in *Proc. 22nd Int. Conf. World Wide Web*, Rio de Janeiro, Brazil, May 2013, pp. 165–166.

[24] A. Yeboah-Ofori, U. M. Ismail, T. Swidurski, and F. Opoku-Boateng, "Cyber threat ontology and adversarial machine learning attacks: Analysis and prediction perturbance," in *Proc. ICCMA*. Esch-sur-Alzette, Luxembourg: Univ. of Luxembourg, Jul. 2021, pp. 71–77.

[25] Y. Merah and T. Kenaza, "Proactive ontology-based cyber threat intelligence analytic," in *Proc. ICRAMI*. Tébessa, Algeria: Tebessa Univ., Sep. 2021, pp. 1–7.

[26] Z. Syed, A. Padia, T. Finin, L. Mathews, and A. Joshi, "UCO: A unified cybersecurity ontology," in *Proc. AAAI Workshop Artif. Intell. Cyber Secur.*, Palo Alto, CA, USA, 2016, pp. 195–202.

[27] Information Technology Laboratory. *NCP, CCE*. [Online]. Available: https://ncp.nist.gov/cce/index

[28] *FIRST, CVSS-SIG*. [Online]. Available: https://www.first.org/cvss/

[29] Lockheed Martin. *Cyber Kill Chain*. [Online]. Available: https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html#OVERVIEW

[30] J. R. Goodall, R. A. Bridges, M. Iannacone, and K. M. Huffer. *Stucco: A Cyber Intelligence Platform*. [Online]. Available: https://stucco.github.io/

[31] N. Scarpato, N. D. Cilia, and M. Romano, "Reachability matrix ontology: A cybersecurity ontology," *Appl. Artif. Intell.*, vol. 33, no. 7, pp. 643–655, Jun. 2019.

[32] H. Kim, H. Kwon, and K. K. Kim, "Modified cyber kill chain model for multimedia service environments," *Multimedia Tools Appl.*, vol. 78, no. 3, pp. 3153–3170, Feb. 2019.

[33] A. Ju, Y. Guo, and T. Li, "MCKC: A modified cyber kill chain model for cognitive APTs analysis within enterprise multimedia network," *Multimedia Tools Appl.*, vol. 79, nos. 39–40, pp. 29923–29949, Oct. 2020.

[34] S. Qamar, Z. Anwar, M. A. Rahman, E. Al-Shaer, and B.-T. Chu, "Data-driven analytics for cyber-threat intelligence and information sharing," *Comput. Secur.*, vol. 67, pp. 35–58, Jun. 2017.

[35] *CTI TC, OASIS Open*. [Online]. Available: https://oasis-open.github.io/cti-documentation/stix/compare.html

[36] Z. Liu, Z. Sun, J. Chen, Y. Zhou, T. Yang, H. Yang, and J. Liu, "STIX-based network security knowledge graph ontology modeling method," in *Proc. ICGDA*, Marseille, France, Apr. 2020, pp. 152–157.

[37] Y. Zhao, B. Lang, and M. Liu, "Ontology-based unified model for heterogeneous threat intelligence integration and sharing," in *Proc. 11th ASID*, Xiamen, China, Oct. 2017, pp. 11–15.

[38] V. Mavroeidis, R. Hohimer, T. Casey, and A. Jøsang, "Threat actor type inference and characterization within cyber threat intelligence," in *Proc. 13th CyCon*, Tallinn, Estonia, May 2021, pp. 327–352.

[39] J. A. Wang and M. Guo, "OVM: An ontology for vulnerability management," in *Proc. CSIIRW*, Oak Ridge, TN, USA, 2009, pp. 1–4.

[40] R. Syed, "Cybersecurity vulnerability management: A conceptual ontology and cyber intelligence alert system," *Inf. Manag.*, vol. 57, no. 6, Sep. 2020, Art. no. 103334.

[41] N. Rastogi, S. Dutta, M. J. Zaki, A. Gittens, and C. Aggarwal, "MALOnt: An ontology for malware threat intelligence," in *Proc. 2nd MLHat*, San Diego, CA, USA, 2020, pp. 28–44.

[42] B. Mozzaquatro, C. Agostinho, D. Goncalves, J. Martins, and R. Jardim-Goncalves, “An ontology-based cybersecurity framework for the Internet of Things,” *Sensors*, vol. 18, no. 9, p. 3053, Sep. 2018.

[43] Y. Jia, Y. Qi, H. Shang, R. Jiang, and A. Li, “A practical approach to constructing a knowledge graph for cybersecurity,” *Engineering*, vol. 4, no. 1, pp. 53–60, 2018.

[44] L. Bin, Y. Jiacai, Y. Li, W. Yanjuan, D. Zhaoyun, and Z. Xianqiang, “Situational awareness ontology for threat from space cyber operations,” *Syst. Eng. Electron.* [Online]. Available: https://kns.cnki.net/kcms/detail/11.2422.TN.20220330.1628.007.html

[45] M. Souppaya and K. Scarfone. (Mar. 14, 2016). *Draft NIST Special Publication 800-154—Guide to Data-Centric System Threat Modeling*. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-154/draft

[46] N. Shevchenko, “Threat modeling for cyber-physical system-of-systems: Methods evaluation,” Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep., 2018.

[47] D. J. Bodeau, C. D. McCollum, and D. B. Fox, “Cyber threat modeling: Survey, assessment, and representative framework,” Homeland Secur. Syst. Eng. Develop. Inst. (HSSEDI), Tech. Rep., 2018.

[48] C. Y. Cheung, “Threat modeling techniques-program: MSc systems engineering, policy analysis and management,” Delft Univ. Technol., Delft, The Netherlands, Tech. Rep., 2016.

[49] E. Hutchins, M. J. Cloppert, and R. M. Amin. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. Lockheed Martin Corporation. [Online]. Available: https://www.lockheedmartin.com/content/dam/lockheed-Martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf

[50] R. Al-Shaer, J. Spring, and E. Christou, “Learning the associations of MITRE ATT&CK adversarial techniques,” in *Proc. IEEE CNS*, Avignon, France, Jun./Jul. 2020, pp. 1–9.

[51] A. Piazza, “ATT&CKing threat management: A structured methodology for cyber threat,” SANS Inst., Bethesda, MD, USA, Tech. Rep., 2019.

[52] T. R. Gruber, “A translation approach to portable ontologies,” *Knowl. Acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[53] T. R. Gruber, “Toward principles for the design of ontologies used for knowledge sharing,” *Int. J. Hum.-Comput. Stud.*, vol. 43, nos. 5–6, pp. 907–928, Nov./Dec. 1995.

[54] V. M. Kureychik and I. B. Safronenkova, “Ontology-based approach to design problem formalization,” in *Proc. SED*, Prague, Czech Republic, Apr. 2019, pp. 1–5.

[55] L. Lin and W. Hong, “Classification of fundamental geographic information based on formal ontology,” *Geomatics Inf. Sci. Wuhan Univ.*, vol. 31, no. 6, pp. 523–526, 2006.

[56] X. Hu and J. Liu, “Ontology construction and evaluation of UAV FCMS software requirement elicitation considering geographic environment factors,” *IEEE Access*, vol. 8, pp. 106165–106182, 2020.

[57] A. Hamdulla, H. Yilahun, K. Abdurahman, and S. Imam, “A hierarchical clustering based relation extraction method for domain ontology,” in *Proc. 9th PAAP*, Taiwan, Dec. 2018, pp. 36–40.

[58] M. Horridge, *A Practical Guide to Building OWL Ontologies Using Protégé 4 and CO-ODE Tools*, 1st ed. Manchester, U.K.: Univ. of Manchester, 2011.

[59] X. Hu, J. Liu, and Y. Wang, “Researches on software requirements elicitation approach of the aviation electronics systems based on multi-ontology,” in *Proc. ICACT*, Pyeongchang, South Korea, Feb. 2020, pp. 330–335.

[60] W. Zhong, “Research on ontology-based cyber attack modeling and analysis,” M.S. thesis, School Electr. Inf. Elect. Eng., Shanghai Jiao Tong Univ., Shanghai, China, 2018.

[61] Y. Liu, H. T. Loh, and A. Sun, “Imbalanced text classification: A term weighting approach,” *Expert Syst. Appl.*, vol. 36, no. 1, pp. 690–701, Jan. 2009.

[62] J. P. Anderson, “Computer security technology planning study,” James P. Anderson Co., Fort Washington, PA, USA, Rep. ESD-TR-73-51, Oct. 1972.

[63] C. Nickerson, D. Kennedy, E. Smith, A. Rabie, S. Friedli, J. Searle, B. Knight, C. Gates, and J. McCray, “Penetration testing execution standard,” Tech. Rep., 2014.

[64] S. V. Morais, A. O. de Sa, L. F. Rust, and C. M. Farias, “Malicious traffic description: Toward a data model for mitigating security threats to home IoT,” *IEEE Commun. Standards Mag.*, vol. 5, no. 3, pp. 48–55, Sep. 2021.

XUAN HU received the Ph.D. degree in aerospace science and technology from Beihang University, China, in 2010. She is currently a Senior Engineer with the CEPREI Laboratory. Her research interests include ontology modeling, software requirement engineering, and software reliability.

DEBIN CHENG received the B.S. degree in automation from the University of Science and Technology Beijing, China, in 1997. He is currently the Director of the Information Security Center, CEPREI Laboratory. His research interests include information security, software reliability and security, and software evaluation technology.

JUNMING CHEN received the B.S. degree in computer science from the Guilin University of Electronic Technology, China. He is currently an Engineer with the CEPREI Laboratory. His research interests include cybersecurity and adversarial attack and defense technique.

XIANTAO JIN received the M.Sc. degree in automation from the Wuhan University of Technology, China. He is currently a Senior Engineer with the CEPREI Laboratory. His research interests include vulnerability mining of ICS and adversarial attack and defense technique.

BO WU received the Ph.D. degree in pulping and papermaking engineering from the South China University of Technology, China, in 2012. He is currently a Senior Engineer with the CEPREI Laboratory. His research interests include industrial process control and computer simulation, information security of ICS, information test, and vulnerability discovery.