electronics

Article

Cyber Threat Intelligence Framework for Incident Response in an Energy Cloud Platform

Seonghyeon Gong and Changhoon Lee *

Department of Computer Science and Engineering, Seoul National University of Science and Technology, Seoul 01811, Korea; gongsh@seoultech.ac.kr
* Correspondence: chlee@seoultech.ac.kr

Abstract: Advanced information technologies have transformed into high-level services for more efficient use of energy resources through the fusion with the energy infrastructure. As a part of these technologies, the energy cloud is a technology that maximizes the efficiency of energy resources through the organic connection between the entities that produce and consume the energy. However, the disruption or destruction of energy cloud systems through cyberattacks can lead to incidents such as massive blackouts, which can lead to national disasters. Furthermore, since the technique and severity of modern cyberattacks continue to improve, the energy cloud environment must be designed to resist cyberattacks. However, since the energy cloud environment has different characteristics from general infrastructures such as the smart grid and the Advanced Metering Infrastructure (AMI), it requires security technology specialized to its environment. This paper proposes a cyber threat intelligence framework to improve the energy cloud environment's security. Cyber Threat Intelligence (CTI) is a technology to actively respond to advanced cyber threats by collecting and analyzing various threat indicators and generating contextual knowledge about the cyber threats. The framework proposed in this paper analyzes threat indicators that can be collected in the advanced metering infrastructure and proposes a cyber threat intelligence generation technique targeting the energy cloud. This paper also proposes a method that can quickly apply a security model to a large-scale energy cloud infrastructure through a mechanism for sharing and spreading cyber threat intelligence between the AMI layer and the cloud layer. Our framework provides a way to effectively apply the proposed technologies through the CTI architecture, including the local AMI layer, the station layer, and the cloud layer. Furthermore, we show that the proposed framework can effectively respond to cyber threats by showing a 0.822 macro-F1 score and a 0.843 micro-F1 score for cyberattack detection in an environment that simulates a model of an attacker and an energy cloud environment.

Keywords: cyber threat intelligence; energy cloud; advanced metering infrastructure; smart grid; incident response

check for updates

Citation: Gong, S.; Lee, C. Cyber Threat Intelligence Framework for Incident Response in an Energy Cloud Platform. *Electronics* **2021**, *10*, 239. https://doi.org/10.3390/electronics10030239

Received: 4 December 2020
Accepted: 28 December 2020
Published: 21 January 2021

1. Introduction

The development of Information Technologies (ITs) has emerged in various paradigms such as Artificial Intelligence (AI), blockchain, and the Internet of Things (IoT), and the smart energy environment represented by the smart grid is one of them. The energy cloud refers to an infrastructure in which energy can be used flexibly and efficiently by organically connecting energy producers and consumers to realize a smart energy environment. In the energy cloud, users exist as prosumers, meaning they are simultaneously consumers and producers, and each user consumes, generates, and trades energy for his or her own needs. This infrastructure is connected in an organic form, and it flexibly responds to energy demand and maximizes the overall energy efficiency by devising a plan that can optimally respond to the energy demand.

The advancement of various ITs technologies has brought the side effect of the advancement of cyberattacks. Currently, various IT technologies have been applied in people's daily lives. As a result, critical assets such as personal information, corporate confidentiality, and essential elements for life, such as energy and transportation, are managed based on ITs. This aspect means that the damage caused by cyberattacks can gradually directly impact people's lives.

Furthermore, since national infrastructures such as the electricity grid and nuclear power are also managed based on ITs, cyberattacks on these targets can cause severe damage. Cyberattacks against infrastructure have been steadily increasing, starting with Stuxnet, which occurred in 2010. A cyberattack by the Energetic Bear group in 2014 destroyed approximately 2800 power industry-related systems. The Black Energy attack, which attacked Ukraine's grid in December 2015, resulted in an energy loss of about 73 MWh and 225,000 victims [1]. Attacks on these countries' underlying energy infrastructure can destroy the energy supply chain, leading to national disasters such as blackouts.

Since the energy cloud environment is part of a country's critical infrastructure, it must satisfy a rigid security level. However, since the energy cloud environment and the AMI that compose the base infrastructure have different compositions compared to the general IT system, there are limits in applying a general security software [2]. Thus, the energy cloud system's security framework must consider the diversity and limitations of hardware resources, the real-time characteristics of the devices constituting the infrastructure, and the utility of energy data [3].

Cyber threat intelligence is a security system for actively responding to cyberattacks by forming contextual knowledge information about cyber threats that may exist based on data observed from various sources. In addition to general security systems such as anti-virus programs, Intrusion Detection Systems (IDSs), and firewalls, CTI analyses a cyberattack's characteristics and infers patterns by collecting data related to various types of cyberattacks and analyzing their correlations [4]. This method enables an effective response to high-level cyber threats and zero-day attacks with unknown vulnerabilities [5]. Moreover, CTI analyzes cyberattack trends to predict possible future attack patterns and their techniques [6].

In this paper, we propose an energy cloud security framework based on cyber threat intelligence to improve the security capabilities of the energy cloud environment and to respond to advanced cyber threats. Cyber threat intelligence infers countermeasures against cyber threats using various types of data. To reflect the specificity of the energy cloud environment properly, the proposed framework consists of three layers: the local AMI layer, the station layer, and the cloud layer.

The AMI layer includes various prosumer client devices that make up the energy cloud environment. These clients operate certain necessary security systems, such as the anti-virus and firewall systems, considering the performance capabilities of each device. Moreover, these devices collect indicators of compromises related to networks and systems to identify the presence of cyber threats. The collected indicator information is transmitted to the station layer with the device behavior information.

The station layer consists of management servers that are in charge of the lower energy cloud network. A sub-network is a network configured for a specific purpose, such as for an organization's IT environment or solar panel cluster. Each management server builds a security model and policy suitable for various types of sub-AMI networks. The station layer forms a hierarchical structure to encompass various network environments. This structure increases the overall compatibility of the energy cloud environment, composed of various devices and power generation systems. Each management server can also construct a security system specialized for each system and network by building a security model optimized for the sub-network. Such a system can provide higher performance than a general security system covering a wide area against advanced cyberattacks.

The central CTI server in the cloud layer generates a CTI by synthesizing Indicators of Compromises (IoCs) collected in the AMI layer and data related to cyberattacks analyzed in the station layer. The central CTI server also collects various types of data collected from lower layers and data related to cyber threats collected from external Open-Source

Intelligence (OSINT) [7]. The central server analyzes all types of collected data to generate high-level CTIs and then shares these CTIs with the station layer servers so that the entire system can respond to zero-day attacks.

The CTI-based security framework for the energy cloud environment makes the following contributions.

- Representation and creation of energy cloud-specific CTI data: The proposed framework identifies and collects IoC data specialized for the AMI layer's energy cloud environment. Through this process, data can be selected to suit the characteristics of the AMI layer in which various devices are included, and this forms the basis of practical security functions for responding to cyberattacks targeting AMIs.
- Applying the optimal security model through a hierarchical CTI architecture: The central CTI server creates a CTI to counter common and general cyber threats. On the other hand, the station layer's CTI server generates a CTI suitable for the lower AMI network using the threat-related data provided from the cloud layer to counter targeted and advanced threats. This hierarchical CTI structure copes with both cyberattacks that attack a wide area and those that attack a specific target.

The structure of this paper is as follows. Section 2 describes cyber threats and CTI-related studies in the energy cloud environment. Section 3 describes the specific contents and operation of the proposed CTI framework, and Section 4 presents a plan for implementing the proposed CTI architecture. Section 5 presents experimental results that demonstrate the effectiveness of the proposed CTI framework. Section 6 discusses the meaning and contribution of the proposed framework and concludes the paper.

2. Related Work

This section introduces previous studies related to the energy cloud, AMI, and cyber threat intelligence. In the meantime, several frameworks and architectures related to smart grids and energy cloud infrastructure have been researched, and many of them have been proposed to enhance security. Some studies that considered cybersecurity on AMI mainly discussed countermeasures to cyber threats that can occur in AMI. Many research topics in the area of cyber threat intelligence have been studied, from data representation to threat modeling methodologies, but research on building a CTI specialized for specific networks is still in its infancy [8].

2.1. Security on Advanced Metering Infrastructure

Existing studies that considered cybersecurity in the AMI environment are divided into those that define the types of cyber threats and related vulnerability factors in the AMI environment and those focusing on countermeasures for each threat factor. Foreman [9] and Hansen [10] analyzed AMI from a cybersecurity perspective. They suggested possible cyberattack surfaces in each component, such as smart meters, data collectors, home networks, and related cyber threat types such as Denial of Power (DoP), Theft of Power (ToP), Distribution of Grid (DoG), and Distributed Denial of Service (DDoS) attacks. Haider [11] also identified possible cyberattacks and attack types targeting AMI wireless networks (especially home networks) and presented a cyber threat model using these types.

Research on countermeasures against cyber threats that may arise from AMI has also been conducted from various perspectives. Kamal [12] proposed a method by which to detect abnormal phenomena such as adversarial node access and meter tempering through pattern analyses of the Received Signal Strength Indication (RSSI). Park [13] proposed a technique that analyzes energy consumption patterns based on the data streams of smart meters and that detects energy theft through Anomaly Pattern Detection based on Hypothesis Testing (APD-HT). Chekired [14] proposed a Hierarchical and Distributed Intrusion Detection System (HD-IDS) that incorporates a distributed type of fog architecture. This research used electricity consumption by smart meters and a stochastic Markov chain to identify normal states and false data injection attacks.

Many studies have also sought to detect cyberattacks in AMI environments using artificial intelligence and deep learning technologies. Marbet [15] proposed a method that determines intrusions by analyzing network packet data with deep learning in AMI's Host Intrusion Detection System (HIDS) and Network Intrusion Detection System (NIDS). Zhang [16] proposed a system that detects intrusions by learning network packet data with an Extreme Learning Machine (ELM). Furthermore, Souza [17] proposed a method by which to detect energy theft by modeling residential and commercial consumers using Self-Organizing Maps (SOM) and Multiplayer Perceptron (MP)-ANN and learning the consumers' energy consumption rates per hour. Bendiab [18] proposed a framework that visualizes AMI network traffic in images using the Binivis technique and detects malicious codes through classification algorithms.

Considering that the AMI environment is a distributed network, user behavior can be analyzed in terms of the demand response on a distributed energy infrastructure. This method can effectively detect abnormal user behavior patterns. Yao et al. [19] proposed a method that detects abnormal behavior in smart meters by analyzing long-term energy data patterns using a Convolutional Neural Network (CNN). In particular, Irtija et al. [20] proposed a method that classifies user types through the energy demand patterns of users in an AMI environment and proposed a modeling methodology for user type classification through contract theory from an economic perspective. This approach has the advantage of modeling a cost function for specific behaviors in a distributed environment and considering the problem of detection in the area of optimization through optimization of the function.

2.2. *Threat Modeling and Cyber Threat Intelligence*

Analyzing cyber threats using the traces of cyberattacks to identify the starting point, progress, and attacker is the basis of a proactive response to a cyber threat. With an in-depth analysis of data related to security events, high-quality and large-scale security event data are required for this basis. A typical IT environment is composed of various, complex interfaces, and these interfaces generate complex and heterogeneous security event data [21]. Several studies have investigated how to define and express cyber threats, i.e., threat modeling methodology to cover these complex and heterogeneous characteristics of security events.

The Incident Object Description Exchange Format (IODEF) [22] and Open Indicators of Compromise (OpenIOC) [23] are pioneering results in the area of threat modeling to express data related to cyberattacks. These results were later extended to research on a data-sharing system combined with a data expression method for a more advanced CTI system [24]. Structured Threat Information Expression (STIX) [25] and Trusted Automated Exchange of Intelligence Information (TAXII) [26] are de facto standard technologies that established a data-sharing system and an expression system for indicators of compromises related to cyber threats. They are currently being used as data management systems for many commercial CTI systems.

CTI can be implemented as a framework for the effective collecting, refining, analyzing, and sharing of cyberattack data. ATIS (Automated Threat Intelligence fuSion framework) [27] proposed an implementation method of CTI through an architecture composed of five planes corresponding to analysis, collection, control, data, and application. Each plane of ATIS plays an explicit role and simultaneously analyzes heterogeneous data effectively through the interaction between each adjacent plane. Gascon et al. [28] proposed a framework referred to as Mantis that integrates cyber threat information using different standards and identifies the relationships among threat data through an attributed graph-based similarity determination algorithm. Mantis showed that 14,000 CyBOX objects could be used to track related threat information with a mean average accuracy of 80% on a single set of cyber threat data. In terms of automated threat-related data collection and analysis, other researchers proposed iACE (IOC Automatic Extractor) [24], which uses natural language processing technology to extract IoC data effectively from documents and uses graph mining techniques for an accurate analysis of the extracted IoC data. iACE recognizes grammatical connections among tokens to extract cyberattack patterns related to IoC in documents, converting these data into the OpenIOC format. iACE extracted 900,000 OpenIOC data objects from 71,000 technical blog articles and showed an accuracy rate of approximately 95% for the classification of the extracted IoC data.

Threat modeling technology has also progressed toward the modeling of the cost and efficiency of each choice made in the relationship between the attacker and the defender through fusion efforts with game theory. Wang et al. [29] proposed a Moving Target Defense (MTD) technique that modeled the interaction between a general device and an attacker using a Markov game in a distributed network environment, in this case the IoT environment, and determined the optimal behavior based on a zero-determinant factor. Sengupta et al. [30] proposed a method for deriving an optimal MTD technique by modeling a strategy based on a general sum Markov game for Advanced Persistent Threat (APT) attacks in a cloud computing environment. These studies are the results of deriving the optimal countermeasures in each situation by optimizing game theory with regard to specific problems and network environments for the modeling and optimization of the cost of security countermeasures.

3. Proposed Cyber Threat Intelligence Framework for Energy Cloud Environments

This section proposes a cyber threat intelligence-based security framework that can improve the cyber threat response capabilities of the existing energy cloud and smart grid environment. Figure 1 shows the architecture of the security framework proposed in this paper. The proposed framework consists of nodes that perform essential functions for the energy cloud demand response and nodes that generate cyber threat intelligence based on IoCs observed in the demand response process. The proposed framework also uses a strategy to respond to cyber threats of various types and patterns through a hierarchical structure.

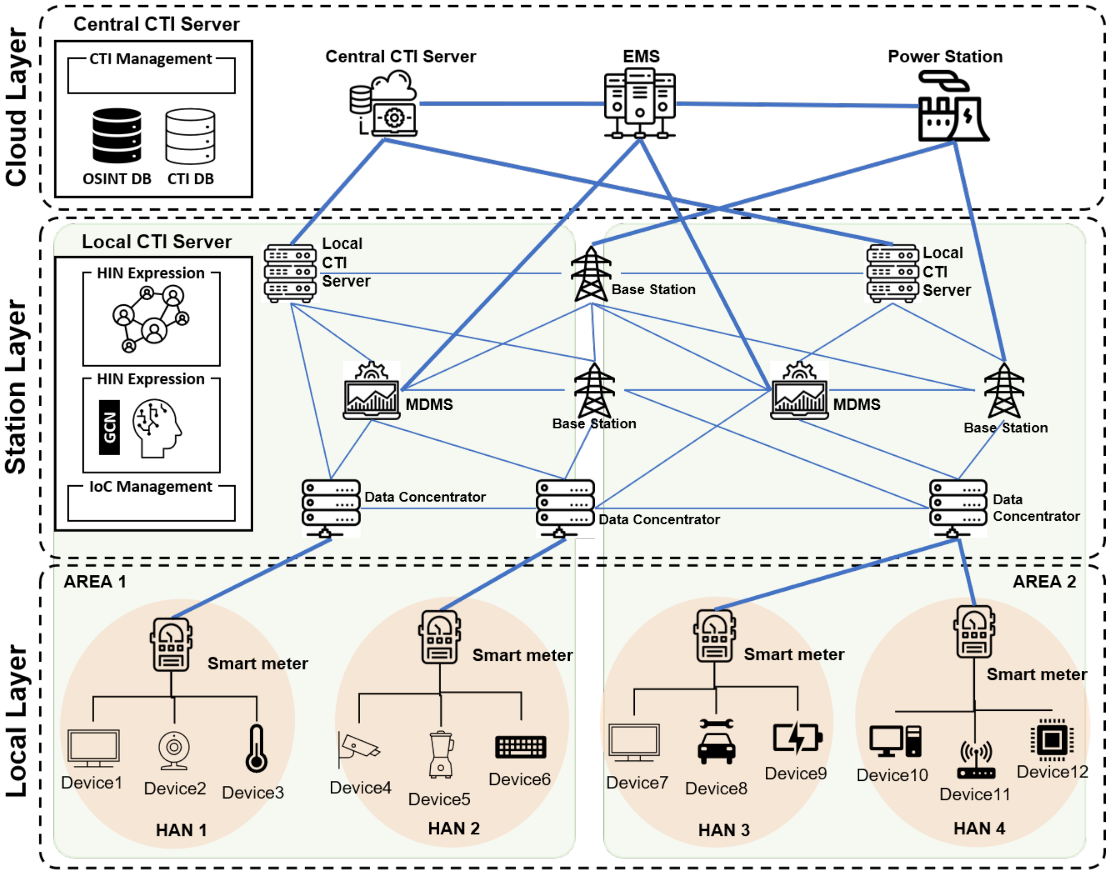


Figure 1. Architecture of the cyber threat intelligence framework for an energy cloud platform. HAN, Home Area Network.

3.1. Architecture of the Proposed Framework

The proposed framework is divided into three layers: the local layer, the station layer, and the backbone layer.

The first layer, the local layer, is where the prosumers responsible for the actual supply and demand of energy are located. Prosumers are devices located in each home, building, and factory that consume energy. Examples include computers, servers, lighting devices, and devices that generate energy, such as solar panels and wind turbines. These prosumer devices fluidly consume and supply energy according to the state of energy required. Smart meters existing in the local layer constitute an AMI environment and manage the energy demand responses of prosumer devices. Each device transmits energy object data related to energy consumption and outputs data to smart meters in real time. Moreover, prosumer devices transmit different status information to smart meters according to device resources and capabilities. Status information includes simple indicators such as timestamps, firmware hash values that identify whether the firmware installed on a device has been tampered with, CPU utilization status indications, and complex indicators such as a list of processes currently running and a list of ports and IP addresses engaged in communication. Prosumer devices periodically transmit device state information to provide data for CTI generation of the local layer. At this time, the amount of state information provided and the transmission period are set to suit the device's resources.

The second layer, the station layer, comprises the Meter Data Management System (MDMS) that manages smart meters of the local layer, the base station in charge of actual energy transmission, and the local CTI server. The MDMS calculates the power demand of the Home Area Network (HAN) based on the energy object data received from the smart meter. The MDMS delivers the calculated power demand data to the base station, and the base station delivers energy to places where energy is needed based on the received data. The essential functions of the MDMS and base stations operate as fundamental elements of the smart grid. The MDMS and base station in each case serve a geographical area, and each area can contain devices with a specific purpose and general prosumer devices. For example, an area containing devices related to wind power or nuclear power has different characteristics from those of typical residential areas containing devices located in such an area. Different characteristics lead to different behavior patterns for each prosumer device. Cyberattacks that focus on energy cloud infrastructure generally target specific Industrial Control Systems (ICSs), Supervisory Control And Data Acquisition (SCADA) systems, or specific Programmable Logic Controllers (PLCs). Therefore, in cyberattacks targeting such environments, the observed indicators and patterns differ from those of traditional cyberattacks. The role of the local CTI server in the station layer is to create cyber threat intelligence specialized for prosumer devices located in each region. Cyber threat intelligence uses a large amount of IoCs to build contextual knowledge about cyberattacks. This method enables comprehensive detection and prediction of a cyberattack through deep inference about the attack, but if the attack pattern is fixed, the odds of false positives and false positives may increase. The type of cyberattack targeting the energy cloud environment depends on the target environment and device. Therefore, it is possible to reduce the ratio of false positives and false positives for cyberattack detection in the energy cloud environment by using a security model trained on an AMI-specific intrusion accident index on the local CTI server.

The third layer, the backbone layer, refers to the site that responds to energy demand requests in a wide area. It consists of a high-voltage transmission center, an energy management system (EMS), and a central CTI server. The EMS analyzes the energy demand flow of the entire energy cloud by synthesizing the energy demand generated by analyzing the energy object data in the MDMS of the station layer. The demand flow analyzed by the EMS leads to the policy to respond to the physical energy demand level, and the high-voltage transmission center transmits energy to each region according to the policy created by the EMS. The EMS and high-voltage transmission center are responsible for the energy cloud's essential functions at the wide-area level. The central CTI server calculates a comprehensive cyber threat response plan for the entire energy cloud environment. Countermeasures against cyber threats generated by the central CTI server include specific IP and domain blocking policies, firewalls, and antivirus software update policies to identify malicious codes quickly. The central CTI server generates cyber threat intelligence by collecting all IoCs collected by general IT systems and AMIs in the local layer and all cyber threat-related data collected by OSINT. The central CTI server abstracts and classifies cyber threat types based on metadata pertaining to various types of IoCs. Based on the classified results, the central CTI server trains a deep learning-based security model that can detect zero-day attacks and that predicts possible future attack types based on the observed attack patterns. Furthermore, the metadata used here serve as data for the classification and identification of attacker groups. This method comprehensively improves the security capabilities of the entire energy cloud environment composed of various systems, from the general computing environment to prosumer devices.

3.2. CTI Generation and Sharing Process

This section describes the collection of the IoC data for the generation of a CTI for each layer of the proposed framework. Figure 2 describes the overall process of the CTI generation and sharing mechanism.

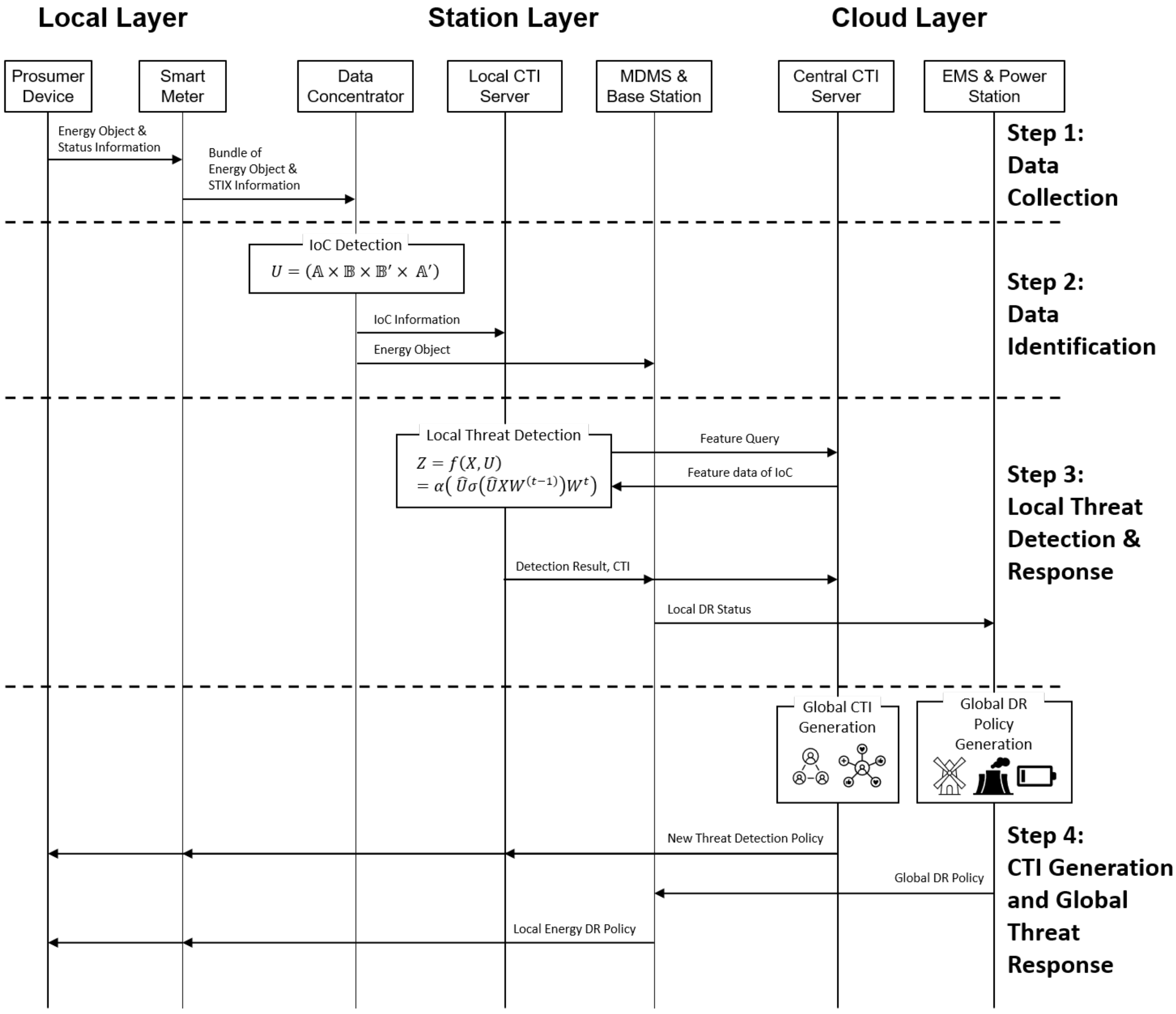

Local Layer
Station Layer
Cloud Layer
Prosumer Device
Smart Meter
Data Concentrator
Local CTI Server
MDMS & Base Station
Central CTI Server
EMS & Power Station
Energy Object & Status Information
Bundle of Energy Object & STIX Information
Step 1: Data Collection
IoC Detection
$U = (\mathbb{A} \times \mathbb{B} \times \mathbb{B}' \times \mathbb{A}')$
IoC Information
Energy Object
Step 2: Data Identification
Local Threat Detection
$Z = f(X, U)$
$= \alpha\left(\hat{U}\sigma\left(\hat{U}XW^{(t-1)}\right)W^{t}\right)$
Feature Query
Feature data of IoC
Detection Result, CTI
Local DR Status
Step 3: Local Threat Detection & Response
Global CTI Generation
Global DR Policy Generation
New Threat Detection Policy
Global DR Policy
Local Energy DR Policy
Step 4: CTI Generation and Global Threat Response

Figure 2. The dissemination process of policies on energy demand-response and CTI-based threat response in an energy cloud environment.

Step 1 (data collection from prosumer devices): Prosumer devices in the local layer's HAN transmit their operating status information to smart meters along with energy object data, including information about energy consumption and production status. The smart meter collects energy object data and status information of various prosumer devices in the HAN and transmits them to the connected data concentrator. For communication between prosumer devices, smart meters, and data concentrators, short-range communication based on wired or wireless communication is used.

Step 2 (energy object data and IoC identification): Data concentrators that receive data from smart meters separate each device's energy object data and state information from the received data. The separated energy object data are transmitted to the MDMS of the station layer to respond to the energy demands of prosumer devices. Simultaneously, the data concentrators identify the network information and device operation statuses included in the device status information received from the smart meter and convert these data to CTI data based on the STIX expression method. The converted CTI data are transmitted to the local CTI server along with the energy objects.

Step 3 (local threat detection and response): The local CTI server learns a deep learning model to detect cyber threats in the area based on the energy object data and CTI data received from the data concentrator. First, the local CTI server identifies the IoC from the CTI data received from the data concentrator. Subsequently, the local CTI server queries the central CTI server's database to obtain feature data for the identified IoC data. After securing the IoC feature data, the local CTI server converts the CTI data into the HIN (Heterogeneous Information Network) format and evaluates whether the CTI data indicate the existence of a cyber threat using a pre-learned Graph Convolutional Network (GCN) model. If it is found to be a cyber threat, the local CTI server transmits the result and CTI data to the central CTI server while simultaneously transmitting the corresponding CTI data to the MDMS. The MDMS then identifies the information on smart meters and prosumer devices included in the CTI data to respond to abnormal behaviors in the energy demand response process.

Step 4 (CTI generation and global threat response): The central CTI server aggregates all CTI data received from multiple local CTI servers. The central CTI server also collects additional data from OSINT using the CTI's IoC resources delivered to collect various cyber threat-related data. Data collection through OSINT improves the quantity and quality of metadata related to IoC resources, which means an improvement in the quality of feature data required to learn GCN-based detection models in local CTI servers. The central CTI server also generates an action policy in a form that can be directly used by smart meters and prosumer devices by synthesizing the cyber threat analysis results received from the local CTI server. These behavioral policies are delivered in the form of a blacklist of IPs and domains or updates of antivirus programs. The CTI generated by the central CTI server is delivered to each local CTI server, smart meter, and prosumer device. At this stage as well, information related to the energy demand response from the MDMS is delivered to the EMS and the highest energy management station. The EMS synthesizes energy demand-related information and the reported CTI and creates a demand response policy for the entire energy cloud system reflecting the response to the cyber threat. This demand response policy is delivered to the MDMS of the station layer and acts as a practical response to the energy demand of smart meters and prosumer devices.

4. Implementation of the Proposed Framework

This section describes the implementation plan of the proposed security framework. The proposed framework is divided into the process of collecting data from a prosumer device, the method by which to generate and learn the local CTI, and the process of generating the global CTI.

4.1. IoC Data Collection from Prosumer Devices

The CTI system's performance when responding to cyber threats is determined by the type and quality of the collected IoC data. Therefore, during the data collection process, data in a form that can express the state of the system to be analyzed in detail must be collected, and similarly, data in a form that can be associated with a cyberattack must be collected. The proposed framework collects data, as shown in Table 1 below, from prosumer devices. These data are collected at regular time intervals considering the resources and performance capabilities of each device.

4.2. Local Threat Detection

Among the data collected in Table 1, process information is used as the primary data to indicate the device status information. Processes running on the system are related to the user who executed the process. If the user is a specific user other than the administrator, the user communicates with the prosumer device using a communication protocol and port number. That is, the user information is related to the protocol and port information. Additionally, external users' actions on prosumer devices leave traces of IP address information about external users. Inspired by the results of HinCTI's research [31], this paper uses the CTI data model shown in Figure 3 for regional threat detection.

Table 1. Device status information and IoC data collected from prosumer devices. PPID, Parent Process Identification Number.

Data Type	Description
Process information list	Process name and PID, PPID information
User information list	User information and UID information for each process
Port information	Port information currently performing communication
Communicated protocols	Communication protocol information currently in operation
Process resource information	Average, variance, maximum, minimum value of the consumed resources (CPU, memory) of each process in a specific time period
Energy object statistics	Average, variance, maximum, and minimum values of the generated and consumed amount of energy in a specific time period (energy object)
Communicated IP address	IP address information (IoC) of the other party currently communicating

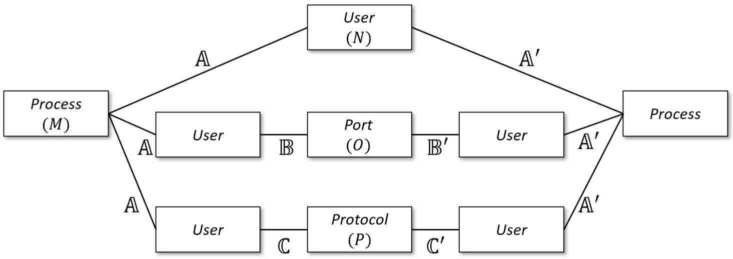


Figure 3. Information network using local threat detection.

Data types having a connection relationship can be expressed as a matrix of the connection relationship between each data instance. For example, the connection relationship between M processes and N users can be expressed as an adjacency matrix having a size of $M \times N$, and each element of the adjacency matrix indicates whether the two data instances are connected (1) or not connected (0). For each type of data collected by the prosumer device, an adjacency matrix indicating the relationship between the data instances can be expressed as follows.

- Adjacent matrix representing the relationship between M processes and N users: $\mathbb{A}$;
- Adjacent matrix representing the relationship between N users and O port information: $\mathbb{B}$;
- Adjacent matrix representing the relationship between N users and P protocol information: $\mathbb{C}$.

Using the relationship data in Figure 3, three types of relationships can be inferred using process information, user information, port information, and protocol information. The relationship L_1 can be expressed through the diagonal matrices of $\mathbb{A}$ and $\mathbb{A}'$, and the resulting $M \times M$ matrix can represent the relationship between the process and the process owner. The relationship L_2 can be derived from the calculation result of $\mathbb{A} \times \mathbb{B} \times \mathbb{B}' \times \mathbb{A}'$, and this result indicates the relationship between the process and port information used in the prosumer device. Furthermore, the relationship L_3 can be derived as a calculation result of $\mathbb{A} \times \mathbb{C} \times \mathbb{C}' \times \mathbb{A}'$, and this result may indicate the current communication state of the prosumer device. Inspired by earlier work [32], given that the calculation results of the three associations above are all matrices of the same form, the corresponding Hadamard product (•) yields a matrix U including all relationships that can be considered in a prosumer device. The equation for expressing all of the relationships of the data of the prosumer devices as an adjacency matrix U is shown below.

$$U = (\mathbb{A} \times \mathbb{A}') \bullet (\mathbb{A} \times \mathbb{B} \times \mathbb{B}' \times \mathbb{A}') \bullet (\mathbb{A} \times \mathbb{C} \times \mathbb{C}' \times \mathbb{A}') \tag{1}$$

The process resource information delivered from the prosumer device is information on the CPU and memory resource consumption of each process running on the device. This information can be viewed as information of the physical layer of the device's operating state, and through this information, the normal operating state of the device can be inferred. The proposed framework uses the device's process resource information as feature matrix X for the process. X is a matrix with a size of $M \times w$, and w indicates the dimension of information in the process of gathering the resource.

The state information of the device observed at each specific period in the prosumer device constructs the feature matrix X. The following is a description of the feature matrix using device-related data mentioned in Table 1.

- The process information includes the PID (Process Identification Number) and the PPID (Parent Process Identification Number) of the process running in the prosumer device. Because a prosumer device dedicated to the energy cloud environment continuously performs a predetermined operation, it can be expected that the process's state will not change significantly. Accordingly, it is possible to determine a white list of expected processes in a normal device state. In the Raspberry Pi-based prosumer device configured for this research, on average, eighty-six processes operate. We limit all possible types of processes to 100, including unknown processes, and use the process list as an element of the feature matrix through one-hot encoding.
- The prosumer device can use the white list to determine the legitimacy of the User Identifier (UID) that runs the process. A previously known user may be recognized as a legitimate user, and an unknown user identifier may be recognized as not being legitimate. In this research, for a given process list, one binary data instance, indicated as one when all UIDs are registered on the white list, and zero otherwise, was used as an element of the feature matrix.
- Port information is expressed depending on whether or not some well-known ports are used among the device's ports. In this research, eleven features expressing 11 ports with a high frequency of use (20, 21, 22, 23, 69, 80, 161, 443, 990, 992, and others) were used as elements of the feature matrix through one-hot encoding.
- Communication protocol information is expressed as binary data pertaining to whether TCP communication is utilized or whether UDP communication is utilized. In this research, two data instances on two protocols were used as elements of the feature matrix.
- Process resource information is expressed as the average, variance, maximum, and minimum values of the CPU and memory usage of each specific time interval. In this research, eight numerical data instances for these data were used as elements of the feature matrix.
- Energy object statistics are expressed as the average, variance, maximum, and minimum values of the energy produced for each specific time interval. In this re-

search, four numerical data instances for these data were used as elements of the feature matrix.

- Communication IP address information can be expressed depending on whether the corresponding device exists on the white list for other devices currently communicating. In this research, one binary data instance expressing whether all communication exponents exist on the white list was used as an element of the feature matrix.

A total of 127 features were used to construct feature matrix X. The local CTI server trains the GCN model using the state information U of the prosumer device received from the data concentrator and the feature matrix X, which is process resource information. First, the state information U is calculated as $\tilde{U} = U + I$ with a self-loop added and then normalized with $\hat{U} = \hat{D}^{-\frac{1}{2}}\tilde{U}\hat{D}^{-\frac{1}{2}}$. The normalized $\hat{U}$ is used to train the GCN model, as shown below in Equation (2).

$$Z = f(X, U) = softmax(\hat{U}\ ReLU(\hat{U}XW^{(t-1)})W^{(t)}), \tag{2}$$

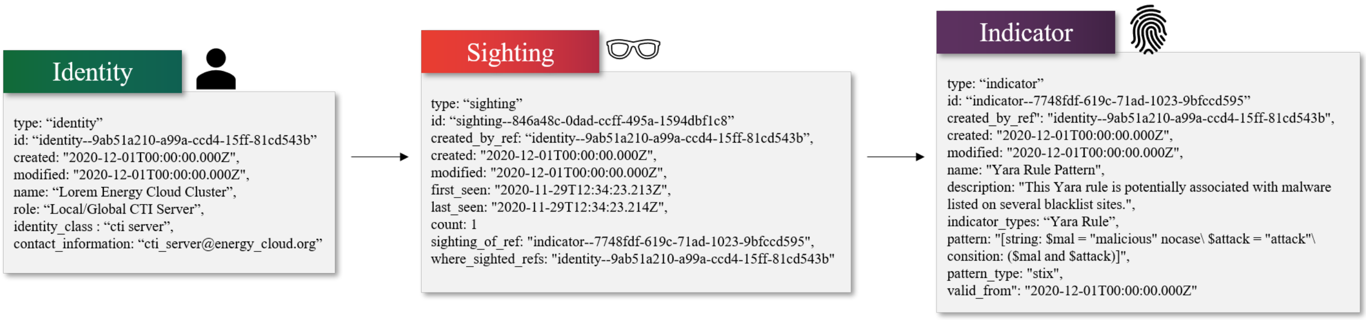
$\tilde{D}$ refers to the degree matrix of $\tilde{U}$. The feature matrix X is input as W^0 of the deep learning model and trains the weight matrix W. ReLU as used in the above model is an activation function defined as $ReLU = max(0, \cdot)$, and softmax is an activation function defined as $softmax(x_i) = e^{x_i} / \sum_j e^{x_j}$. This model receives the state information matrix of the device and the feature matrix for resource consumption and performs multiple classifications for various device states. For multiple classifications, a categorical cross-entropy function is used as a cost function, and the Adam optimizer is used.

4.3. Generation and Dissemination of CTI

The local CTI server at the station layer detects and identifies the local AMI layer's cyber threats by conducting local threat detection using IoC data from the prosumer devices. The local CTI server performs a direct security function for the prosumer device based on the threat analysis result. For this security function, a communication blocking policy or device isolation policy is used to quarantine the attacked prosumer device or block the node identified as an attacker. The communication blocking policy protects the device from cyberattacks by delivering commands and security policies that can control the behavior of prosumer devices, such as the Yara rule or Snort rule. Additionally, device isolation blocks the operation of malicious processes by temporarily stopping or rebooting the attacked device.

For example, feature matrix X, which is the core element of local CTI generation, uses a whitelist of IP information of the opposite device communicating with the prosumer device as a feature during the process of local threat detection. In this process, the opponent's (or attacker's) IP address is transmitted to the local CTI server, and the server creates a CTI related to the Snort rule that blocks the IP according to the local threat detection result. Likewise, process information such as the PID used as an element of the feature matrix can be used for the CTI to form a command to control the prosumer device.

Furthermore, the threat analysis result of the local CTI server is transmitted to the global CTI server. The global CTI server analyzes vulnerabilities and attack types related to the cyberattack in the National Vulnerability Database (NVD) provided by the National Institute of Standards and Technology (NIST) using the transmitted threat-related data. The NVD provides threat-related information through databases, such as Common Vulnerabilities and Exposures (CVE), Common Weakness Enumeration data (CWE), and Common Attack Pattern Enumeration and Classification (CAPEC) data. As shown in the Figures 4 and 5, this knowledge information is used to create STIX-based security policies, and these security policies are disseminated to local CTI servers and prosumer devices to be used as practical guidelines for countering cyberattacks.



type: “identity”
id: “identity--9ab51a210-a99a-ccd4-15ff-81cd543b”
created: "2020-12-01T00:00:00.000Z",
modified: "2020-12-01T00:00:00.000Z",
name: “Lorem Energy Cloud Cluster”,
role: “Local/Global CTI Server”,
identity_class : “cti server”,
contact_information: “cti_server@energy_cloud.org”

type: “sighting”
id: “sighting--846a48c-0dad-ccff-495a-1594dbf1c8”
created_by_ref: “identity--9ab51a210-a99a-ccd4-15ff-81cd543b”,
created: "2020-12-01T00:00:00.000Z",
modified: "2020-12-01T00:00:00.000Z",
first_seen: "2020-11-29T12:34:23.213Z",
last_seen: "2020-11-29T12:34:23.214Z",
count: 1
sighting_of_ref: "indicator--7748fdf-619c-71ad-1023-9bfccd595",
where_sighted_refs: "identity--9ab51a210-a99a-ccd4-15ff-81cd543b"

type: “indicator”
id: “indicator--7748fdf-619c-71ad-1023-9bfccd595”
created_by_ref": "identity--9ab51a210-a99a-ccd4-15ff-81cd543b",
created: "2020-12-01T00:00:00.000Z",
modified: "2020-12-01T00:00:00.000Z",
name: "Yara Rule Pattern",
description: "This Yara rule is potentially associated with malware listed on several blacklist sites.",
indicator_types: “Yara Rule”,
pattern: "[string: $mal = "malicious" nocase\ $attack = "attack"\ consition: ($mal and $attack)]",
pattern_type: "stix",
valid_from": "2020-12-01T00:00:00.000Z"

Figure 4. Examples of scenarios for forming CTI security policy.

```
{
    "type": "bundle", "id": "bundle--8d211d6-1a81-8a26-99ac-70badc948621",
    "objects": [
        {
            "type": "identity",
            "spec_version": "2.1",
            "id": "identity--9ab51a210-a99a-ccd4-15ff-81cd543b",
            "created": "2020-12-01T00:00:00.000Z",
            "modified": "2020-12-01T00:00:00.000Z",
            "name": "Lorem Energy Cloud Cluster.",
            "roles": [
                "Local/Global CTI Server"
            ],
            "identity_class": "cti server",
            "contact_information": "cti_server@energy_cloud.org"
        },
        {
            "type": "sighting",
            "spec_version": "2.1",
            "id": "sighting--846a48c-0dad-ccff-495a-1594dbf1c8",
            "created_by_ref": "identity--9ab51a210-a99a-ccd4-15ff-81cd543b",
            "created": "2020-12-01T00:00:00.000Z",
            "modified": "2020-12-01T00:00:00.000Z",
            "first_seen": "2020-11-29T12:34:23.213Z",
            "last_seen": "2020-11-29T12:34:23.214Z",
            "count": 1,
            "sighting_of_ref": "indicator--7748fdf-619c-71ad-1023-9bfccd595",
            "where_sighted_refs": [
                "identity--9ab51a210-a99a-ccd4-15ff-81cd543b"
            ]
        },
        {
            "type": "indicator",
            "spec_version": "2.1",
            "id": "indicator--7748fdf-619c-71ad-1023-9bfccd595",
            "created_by_ref": "identity--9ab51a210-a99a-ccd4-15ff-81cd543b",
            "created": "2020-12-01T00:00:00.000Z",
            "modified": "2020-12-01T00:00:00.000Z",
            "name": "Yara Rule Pattern",
            "description": "This Yara rule is potentially associated with malware listed on several blacklist sites.",
            "indicator_types": [
                "Yara-rule"
            ],
            "pattern": "[string: $mal = "malicious" nocase\ $attack = "attack"\ consition: ($mal and $attack)]",
            "pattern_type": "stix",
            "valid_from": "2020-12-01T00:00:00.000Z"
        }
    ]
}
```

Figure 5. Example of STIX-based CTI security policy.

4.4. Computational Complexity and Implementation Cost

The proposed framework should consider computational complexity and the data traffic cost when it undertakes anomaly detection. During the process of training the GCN model for local threat detection, the adjacency matrix A is the result of multiplying the adjacency matrices for IoC data. The size of the input data is greatly influenced by the size of the IoC data handled by the GCN model. Because the process of outputting and transmitting information for forming feature matrix X in a prosumer device is not a large operation compared to the device's performance, the time complexity for data operation is determined by the size of IoC data to be observed. The IoC data's complexity as described in Section 4.2 and Figure 3 is determined by the number of processes expected to run on a prosumer device. The number of processes is an indicator affected by the prosumer device's role and performance. As more prosumer devices are connected to a complex network and play various roles, more processes are expected to operate, leading to an exponential load of multiplication operations in the adjacency matrix for IoC data. The time complexity of the general matrix multiplication operation is $O(n^{2.807})$ [33], which is the time complexity of Strassen's algorithm, and the time complexity of the fastest algorithm known to date is the method proposed by Josh and Virginia: $O(n^{2.3728596})$ [34]. During the GCN model's learning process, matrix multiplication is used in the graph convolutional layer and the fully connected layer, but the order of the matrix is lower than that of the adjacency matrix for IoT data, which does not affect the time complexity.

Another consideration with regard to the implementation cost is network traffic. The proposed framework requires near real-time threat detection to respond to cyberattacks in the energy cloud environment. To this end, each prosumer device shares its device status information in a short period. If the number of prosumer devices constituting the energy cloud environment is very large, the energy cloud network must accurately accommodate a large amount of device status information transmitted every second.

5. Experiments

In this section, simulations and results are described to prove the proposed framework's effectiveness and performance. The proposed framework protects the system from cyberattacks against energy demand responses by generating CTI based on the prosumer device's state information. This experiment models attackers' attacks that can occur during the energy demand response process and evaluates how accurately the proposed framework can detect and respond to cyberattacks through simulation results.

5.1. Experimental Setup

In this experiment, a prosumer device and a smart meter were configured with the Raspberry Pi 4B model, and the simulation environment was configured with a local CTI server and a central CTI server separately. The specifications of the configured devices and servers are shown in Table 2. Prosumer devices use solar panels and batteries connected to the Raspberry Pi device as energy production modules, and the motors and lighting devices are the energy consumption modules. Each prosumer device measures the energy produced and consumed in 0.2 seconds and transmits the average energy production and consumption information and the process status information to the CTI server every second. This experiment simulates the process of generating the CTI and detecting cyber threats using energy object data and 20,480 IoC data instances transmitted from a prosumer device every second.

Table 2. Information on devices, servers, and libraries used in the experiment.

Spec	Prosumer Device & Smart Meter	Local CTI Server	Central CTI Server
CPU	ARM Cortex-A72 1.5 GHz	Intel i7-9700KF 3.60 GHz	Intel Xeon E5-2697 v4 2.30 GHz
Memory	4 GB	64 GB	64 GB
GPU	Broadcom VideoCore VI	NVIDIA GeForce RTX 2080 SUPER	NVIDIA TESLA P100
ML/DL Library	-	Tensorflow 2.3.1	Tensorflow 2.2.0

5.2. Threat Model

To simulate an attacker's cyberattack on the energy cloud's demand response process, this experiment sets up an attacker who injects abnormal data in the energy demand state. The attacker abnormally manipulates the amounts in the energy measurements consumed and produced by the prosumer device. The attacker induces a malfunction of the sensing function that measures energy consumption and production amounts through malicious code and can lower, increase, or fix the target prosumer's energy production and consumption levels. This experiment simulated an attacker attacking the energy demand response process through the following three attack types. Each attack type involves an attack of an abnormal value injected into the GPIO interface of the Raspberry Pi, a prosumer device.

- Attack Type 1: Inject a high value into the GPIO interface of the energy measurement module to disguise energy as over-produced and over-consumed.
- Attack Type 2: Inject a low value into the GPIO interface of the energy measurement module to disguise as if no energy is produced or consumed.
- Attack Type 3: By injecting a fixed value into the GPIO interface of the energy measurement module, it disguises as if energy was abnormally produced and consumed.

Regarding the data transmitted from the prosumer device, the ratio of the actual attacked data is shown in Table 3.

Table 3. Dataset used in the experiment.

Device Status	Number of Data	Ratio
Normal	15,823	77.26%
Attack Type 1	2101	10.26%
Attack Type 2	1978	9.66%
Attack Type 3	578	2.82%
Total	20,480	100%

5.3. Performance Evaluation

In this experiment, multi-class classification was conducted for 20,480 data instances and four classes. Data learning was performed through 10-fold cross-validation. The confusion matrix presented in Figure 6 shows the classification accuracy for each class. Figure 7a–c shows the power amount data generated by the prosumer device and the point in time when three attack patterns were conducted in the entire dataset. In each graph, the blue line represents the measured power production and the orange graph represents the time point at which the corresponding type of attack was performed, expressed as zero and one, respectively. Each type of attack was performed when the orange graph had a high value. The confusion matrix in Figure 6 is the sum of the confusion matrices derived from each verification phase of the cross-validation. In the dataset used in this experiment, the ratio of data is relatively concentrated on the normal case. For this reason, the accuracy for a normal label shows relatively high performance. Looking at the results for Type 1 and Type 2, the detection accuracy for attacks that manipulate the amount of power production, causing it to be undervalued, exceeds that for relapses that manipulate the amount of power production and cause it to be overvalued. This occurs because the pattern of the amount of power calculated in the experimental environment did not fluctuate significantly from the maximum value; thus, an attack that underestimates the amount of power is better detected. Furthermore, in attack Type 3, which is an extreme case, the amount of data is very small, and the attack was intensively performed at the end of the experiment. At the time of the Type 3 attack, the generated amount of power shows two patterns. Therefore, when the amount of attack-related data is extremely small, it can be confirmed that the false detection rate is similar to the distributed pattern on power production at the time the attack is performed. Additionally, this experiment uses the macro-F1 score and micro-F1 score as metrics to measure the results comprehensively when performing multiple classifications for normal patterns and attack types. This metric can effectively measure the performance of a multi-class classification problem of an unbalanced dataset. Let the label group be classified as $\mathcal{S}$ and any label in $\mathcal{S}$ as s. Let the true-positive, true-negative, false-positive, and false-negative rates for an arbitrary label s correspondingly be TP_s, TN_s, FP_s, and TN_s, after which the macro-F1 score and the micro-F1 score are calculated using Equations (3)–(6), as shown below.

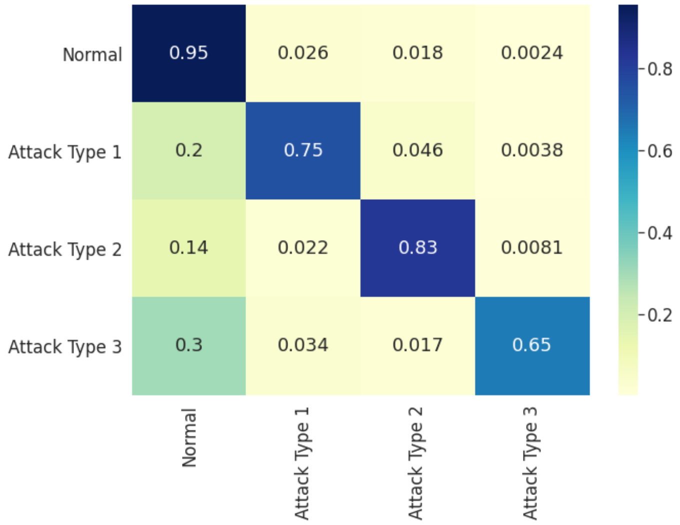
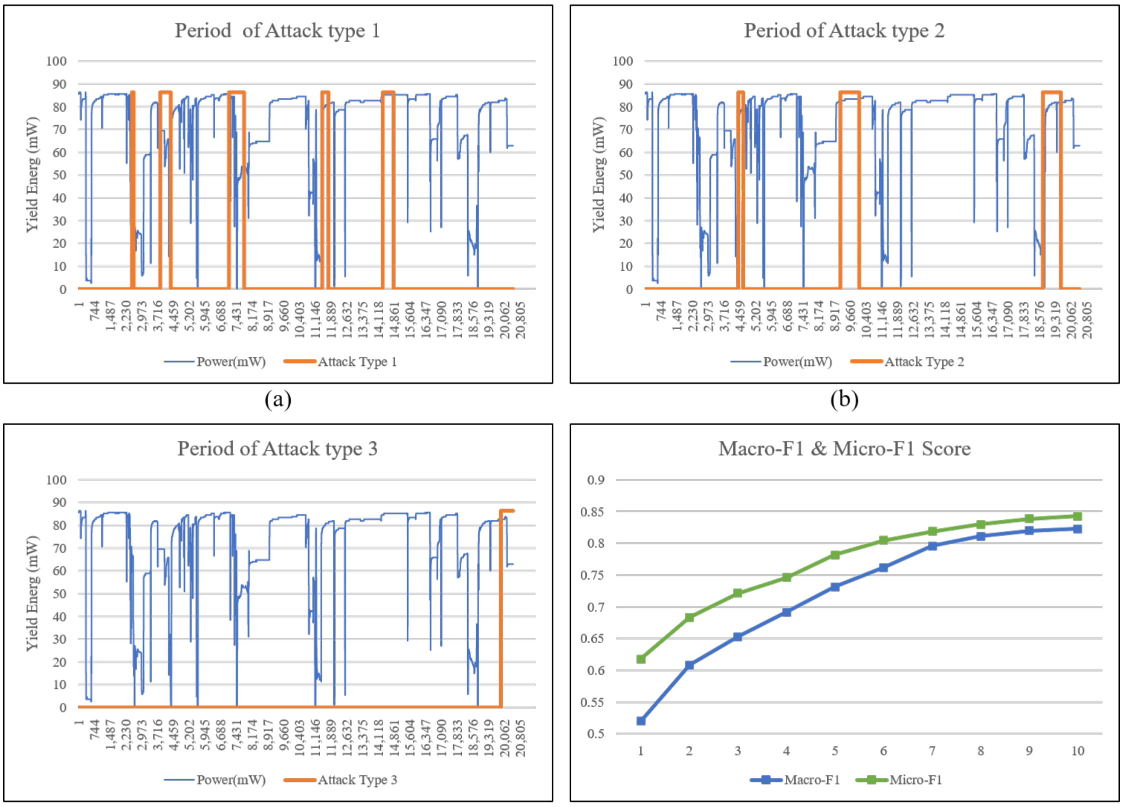


Figure 6. Confusion matrix on the experiment result.



(a) (b)

(c) (d)

Figure 7. (**a**–**c**) Graph of the total energy production observed by the prosumer device and graph indicating the time when the attack was conducted (and the type). The blue line denotes the power production measured by the device, and the orange line indicates when the attack was conducted.(**d**) Macro-F1 score and micro-F1 score value for the multi-class classification performance of the proposed model.

$$Precision_s = \frac{TP_s}{TP_s + FP_s}, Recall_s = \frac{TP_s}{TP_s + FN_s} \tag{3}$$

$$Macro - F1 = \frac{1}{|S|} \sum_{s \in S} \frac{2 \times Precision_s \times Recall_s}{Precision_s + Recall_s} \tag{4}$$

$$Precision_{\mathcal{S}} = \frac{\sum_{s \in \mathcal{S}} TP_s}{\sum_{s \in \mathcal{S}} TP_s + \sum_{s \in \mathcal{S}} FP_s}, Recall_{\mathcal{S}} = \frac{\sum_{s \in \mathcal{S}} TP_s}{\sum_{s \in \mathcal{S}} TP_s + \sum_{s \in \mathcal{S}} FN_s} \tag{5}$$

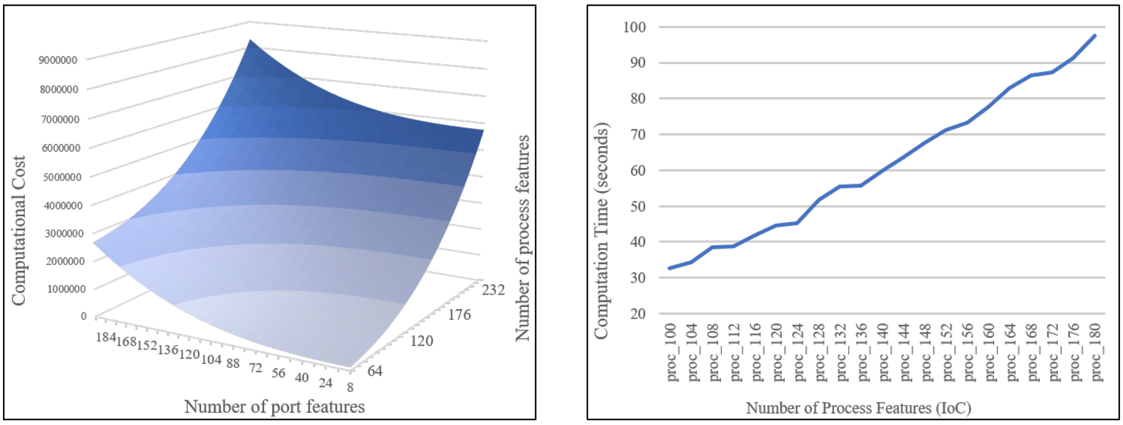
$$Micro - F1 = \frac{2 \times Precision_{\mathcal{S}} \times Recall_s}{Precision_{\mathcal{S}} + Recall_{\mathcal{S}}} \tag{6}$$

Figure 7d shows the results after measuring the multi-class classification performance with the macro-F1 score and the micro-F1 score for normal operation and the three attack types. In Figure 7d, the blue line represents the macro-F1 result and the green line represents the micro-F1 result.

5.4. Evaluation of Computational Complexity and Implementation Cost

In this experiment, when constructing a node feature matrix, data such as the IP and protocol were expressed as a single instance of binary data using a whitelist. This feature information has a relatively weak effect on the computational complexity, whereas feature information subjected to the one-hot encoding of categorical data, in this case process and port information, has an immense impact on the computational complexity. Figure 8a shows the computational complexity of the matrix multiplication operation of the IoC adjacency matrix. The left axis represents the number of floating-point multiplication operations for matrix multiplication, the lower axis the number of ports in the node feature matrix, and the right axis the number of processes in the node feature matrix. This surface graph shows the proportionality of the computational complexity of the two sets of primary feature information.

Figure 8b shows the GCN model's training time for 20,480 data instances performed in the local CTI server environment (Intel i7-9700KF 3.6 GHz with 64 GB RAM). The experiment was performed while increasing the amount of process feature information to analyze the effect of the time complexity on the expansion of IoC data. Figure 8b shows that according to the number of processes, the time it takes to learn the entire dataset increases in the form of an exponential curve proportional to the time complexity $O(n^{2.8})$. In the initial experiment using 100 processes, it took approximately 32.6 s to learn the entire dataset, and in the experiment where 180 processes were used, it took about 97.4 s to learn the dataset. Due to the constraints of the configured environment, there were no significant differences in local threat detection accuracy level according to the number of processes.



(a) (b)

Figure 8. (**a**) Graph on the computational complexity ratio for two major features (number of processes, number of the port) that have the most significant impact on the complexity of the adjacent matrix. (**b**) Total data learning time for the number of processes of the node feature matrix X used in the experiment.

5.5. Discussions and Limitations

The proposed model generates the CTI, detects and responds to cyber threats, and collects state information and the IoC from prosumer devices. The CTI generated in the local area learns a security model optimized for a specific environment by learning the normal behavior patterns of devices using the state information of the devices in the local layer. In this experiment, a macro-F1 score of 0.822 and a micro-F1 score of 0.843 demonstrate that the proposed hierarchical threat response method is practically effective. This result shows that the proposed framework can effectively generate a CTI through data observed in the device domain and can spread a security model through a protocol configured to share it. However, the proposed framework does not consider the time series characteristics of the data. These limitations can lead to somewhat lower threat response performance outcomes for long-term targeted attacks against specific devices. Moreover, due to the nature of IoC data, in which data types are organically connected to each other, a methodology that can dynamically analyze the characteristics from IoC data and convert them to a CTI is required to learn the security code continuously.

6. Conclusions

In this paper, we proposed a CTI framework suitable for energy cloud environments. The proposed framework collects state information and IoC data from prosumer devices in the local layer where the AMI is configured. The collected data are used as a dataset for a deep learning-based threat detection technique in the middle station layer, and the detection result is fused with IoC data to create a security policy specialized for the local environment. In this process, a heterogeneous information network and a graph convolutional network generate contextual knowledge about cyber threats based on various types of data collected from various devices. This method detects cyber threats through contextual knowledge about the device by learning the relationship between the prosumer device's state information and the surrounding environment. Moreover, the detection result of the intermediate station layer is used to generate a global CTI in the cloud layer, and the generated CTI data are used to improve the threat response capability of the entire system. This hierarchical CTI technology forms the ability to respond to both targeted attacks targeting a specific environment and general attacks targeting a wide range of areas. Simultaneously, learning the security model at the local layer can reduce the possibility of overfitting that can occur when simultaneously detecting various types of cyberattacks. This feature is a beneficial security response method in an energy infrastructure environment where system constancy is important. Furthermore, through experiments, we showed that the proposed security framework could realistically ensure the capability to respond to cyber threats.

However, the process of deriving a countermeasure and a security policy for cyberattacks using CTI should be performed only after a cost-perspective analysis. By modeling the behavior of attackers and general users in a given environment and calculating the cost associated with the behavior of each entity and predicting each outcome, the resulting security policy can be used and evaluated from a quantitative perspective.

In future research, we intend to study how the security framework can cope with more long-term and large-scale targeted attacks, model the interaction between attackers and defenders, and derive optimal solutions for each situation. Through this subsequent study, we will analyze the time series characteristics of cyberattacks and the costs of countermeasures and study an advanced framework that can simultaneously consider user privacy during the process of collecting and sharing data.

Author Contributions: The contribution of authors are as follows: conceptualization: S.G. and C.L.; methodology: S.G.; software for experiments: S.G.; validation: S.G.; formal analysis: S.G.; investigation: S.G.; resource: S.G.; data curation: S.G.; writing, original draft preparation: S.G.; writing, review and editing: S.G.; visualization: S.G.; supervision: C.L.; project administration: C.L.; funding acquisition: C.L. All authors read and agreed to the published version of the manuscript.

Funding: This work was supported by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Trade, Industry & Energy(MOTIE) of the Republic of Korea (No. 20171510102090).

Acknowledgments: We thank Seunghee Seo (Seoul National University of Science and Technology) for the discussions. We thank Inyeong Kim (Seoul National University of Science and Technology) for providing the experimental environments.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AMI	Advanced Metering Infrastructure
CTI	Cyber Threat Intelligence
CyBOX	Cyber Observable eXpression
DDoS	Distributed Denial of Service
DoG	Distribution of Grid
DoP	Denial of Power
EMS	Energy Management System
GCN	Graph Convolutional Network
ICS	Industrial Control Systems
IDS	Intrusion Detection System
IoC	Indicator of Compromise
IODEF	Incident Object Description Exchange Format
IoT	Internet of Things
IP	Internet Protocol
MDMS	Meter Data Management System
OSINT	Open-Source Intelligence
PID	Process Identification Number
PLC	Programmable Logic Controller
PPIF	Parent Process Identification Number
SCADA	Supervisory Control And Data Acquisition
STIX	Structured Threat Information Expression
TAXII	Trusted Automated Exchange of Intelligence Information
ToP	Theft of Power
UID	User Identifier

References

1. Khan, R.; Maynard, P.; McLaughlin, K.; Laverty, D.; Sezer, S. Threat analysis of blackenergy malware for synchrophasor based real-time control and monitoring in smart grid. In Proceedings of the 4th International Symposium for ICS & SCADA Cyber Security Research, Belfast, UK, 23–25 August 2016; pp. 53–63.
2. Mavroeidis, V.; Bromander, S. Cyber threat intelligence model: An evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. In Proceedings of the 2017 European Intelligence and Security Informatics Conference (EISIC), Athens, Greece, 11–13 September 2017; pp. 91–98.
3. Sun, C.C.; Liu, C.C.; Xie, J. Cyber-physical system security of a power grid: State-of-the-art. *Electronics* **2016**, *5*, 40. [CrossRef]
4. Ramsdale, A.; Shiaeles, S.; Kolokotronis, N. A comparative analysis of cyber threat intelligence sources, formats and languages. *Electronics* **2020**, *9*, 824. [CrossRef]
5. Serketzis, N.; Katos, V.; Ilioudis, C.; Baltatzis, D.; Pangalos, G. Improving Forensic Triage Efficiency through Cyber Threat Intelligence. *Future Internet* **2019**, *11*, 162. [CrossRef]
6. Griffioen, H.; Booij, T.; Doerr, C. Quality Evaluation of Cyber Threat Intelligence Feeds. In Proceedings of the International Conference on Applied Cryptography and Network Security, Rome, Italy, 22–25 June 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 277–296.
7. Gong, S.; Cho, J.; Lee, C. A reliability comparison method for OSINT validity analysis. *IEEE Trans. Ind. Inform.* **2018**, *14*, 5428–5435. [CrossRef]
8. Wagner, T.D.; Mahbub, K.; Palomar, E.; Abdallah, A.E. Cyber threat intelligence sharing: Survey and research directions. *Comput. Secur.* **2019**, *87*, 101589. [CrossRef]
9. Foreman, J.C.; Gurugubelli, D. Identifying the cyber attack surface of the advanced metering infrastructure. *Electr. J.* **2015**, *28*, 94–103.

10. Hansen, A.; Staggs, J.; Shenoi, S. Security analysis of an advanced metering infrastructure. *Int. J. Crit. Infrastruct. Prot.* **2017**, *18*, 3–19. [CrossRef]
11. Haider, M.H.; Saleem, S.B.; Rafaqat, J.; Sabahat, N. Threat Modeling of Wireless Attacks on Advanced Metering Infrastructure. In Proceedings of the 2019 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS), Karachi, Pakistan, 14–15 December 2019; pp. 1–6.
12. Kamal, M.; Tariq, M. Light-weight security and blockchain based provenance for advanced metering infrastructure. *IEEE Access* **2019**, *7*, 87345–87356. [CrossRef]
13. Park, C.H.; Kim, T. Energy Theft Detection in Advanced Metering Infrastructure Based on Anomaly Pattern Detection. *Energies* **2020**, *13*, 3832. [CrossRef]
14. Chekired, D.A.; Khoukhi, L.; Mouftah, H.T. Fog-Based Distributed Intrusion Detection System Against False Metering Attacks in Smart Grid. In Proceedings of the ICC 2019-2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019; pp. 1–6.
15. El Mrabet, Z.; Ezzari, M.; Elghazi, H.; El Majd, B.A. Deep Learning-Based Intrusion Detection System for Advanced Metering Infrastructure. In Proceedings of the 2nd International Conference on Networking, Information Systems & Security, Rabat, Morocco, 27–28 March 2019; pp. 1–7.
16. Zhang, K.; Hu, Z.; Zhan, Y.; Wang, X.; Guo, K. A Smart Grid AMI Intrusion Detection Strategy Based on Extreme Learning Machine. *Energies* **2020**, *13*, 4907. [CrossRef]
17. de Souza, M.A.; Pereira, J.L.; Alves, G.D.O.; de Oliveira, B.C.; Melo, I.D.; Garcia, P.A. Detection and identification of energy theft in advanced metering infrastructures. *Electr. Power Syst. Res.* **2020**, *182*, 106258. [CrossRef]
18. Bendiab, G.; Grammatikakis, K.P.; Koufos, I.; Kolokotronis, N.; Shiaeles, S. Advanced metering infrastructures: Security risks and mitigation. In Proceedings of the 15th International Conference on Availability, Reliability and Security, Dublin, Ireland, 25–28 August 2020; pp. 1–8.
19. Yao, D.; Wen, M.; Liang, X.; Fu, Z.; Zhang, K.; Yang, B. Energy theft detection with energy privacy preservation in the smart grid. *IEEE Internet Things J.* **2019**, *6*, 7659–7669. [CrossRef]
20. Irtija, N.; Sangoleye, F.; Tsiropoulou, E.E. Contract-Theoretic Demand Response Management in Smart Grid Systems. *IEEE Access* **2020**, *8*, 184976–184987. [CrossRef]
21. Bhatt, S.; Manadhata, P.K.; Zomlot, L. The operational role of security information and event management systems. *IEEE Secur. Priv.* **2014**, *12*, 35–41. [CrossRef]
22. Takahashi, T.; Landfield, K.; Millar, T.; Kadobayashi, Y. IODEF-Extension to Support Structured Cybersecurity Information. 2012. Available online: https://tools.ietf.org/id/draft-takahashi-mile-sci-02.html (accessed on 29 November 2020).
23. Mandiant, O. An Open Framework for Sharing Threat Intelligence. Alexandria, Virginia. 2014. Available online: www.openioc.org (accessed on 29 November 2020).
24. Liao, X.; Yuan, K.; Wang, X.; Li, Z.; Xing, L.; Beyah, R. Acing the ioc game: Toward automatic discovery and analysis of open-source cyber threat intelligence. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 755–766.
25. Barnum, S. Standardizing cyber threat intelligence information with the Structured Threat Information eXpression (STIX). *Mitre Corp.* **2012**, *11*, 1–22.
26. Connolly, J.; Davidson, M.; Schmidt, C. *The Trusted Automated Exchange of Indicator Information (taxii)*; The MITRE Corporation: Bedford, MA, USA, 2014; pp. 1–20.
27. Modi, A.; Sun, Z.; Panwar, A.; Khairnar, T.; Zhao, Z.; Doupe', A.; Ahn, G.J.; Black, P. Towards automated threat intelligence fusion. In Proceedings of the 2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC), Pittsburgh, PA, USA, 1–3 November 2016; pp. 408–416.
28. Gascon, H.; Grobauer, B.; Schreck, T.; Rist, L.; Arp, D.; Rieck, K. Mining attributed graphs for threat intelligence. In Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy, Scottsdale, AZ, USA, 22–24 March 2017; pp. 15–22.
29. Wang, S.; Shi, H.; Hu, Q.; Lin, B.; Cheng, X. Moving Target Defense for Internet of Things Based on the Zero-Determinant Theory. *IEEE Internet Things J.* **2019**, *7*, 661–668. [CrossRef]
30. Sengupta, S.; Chowdhary, A.; Huang, D.; Kambhampati, S. General sum Markov games for strategic detection of advanced persistent threats using moving target defense in cloud networks. In Proceedings of the International Conference on Decision and Game Theory for Security, Stockholm, Sweden, 30 October–1 November 2019; pp. 492–512.
31. Gao, Y.; Xiaoyong, L.; Hao, P.; Fang, B.; Yu, P. HinCTI: A Cyber Threat Intelligence Modeling and Identification System Based on Heterogeneous Information Network. *IEEE Trans. Knowl. Data Eng.* **2020**. [CrossRef]
32. Zhao, H.; Yao, Q.; Li, J.; Song, Y.; Lee, D.L. Meta-graph based recommendation fusion over heterogeneous information networks. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 635–644.
33. Li, J.; Ranka, S.; Sahni, S. Strassen's matrix multiplication on GPUs. In Proceedings of the 2011 IEEE 17th International Conference on Parallel and Distributed Systems, Tainan, Taiwan, 7–9 December 2011; pp. 157–164.
34. Alman, J.; Williams, V.V. A refined laser method and faster matrix multiplication. *arXiv* **2020**, arXiv:2010.05846.