Li *et al. Journal of Cloud Computing: Advances, Systems and Applications* (2020) 9:23
https://doi.org/10.1186/s13677-020-00171-0

Journal of Cloud Computing: Advances, Systems and Applications

RESEARCH Open Access

A security event description of intelligent applications in edge-cloud environment

Qianmu Li[1*], Xiaochun Yin[2], Shunmei Meng[1,3], Yaozong Liu[4] and Zijian Ying[1,3,5]

Abstract

In traditional network environment, the attack topology of the network is usually obtained based on a graph traversal algorithm. It uses connection relationships to describe the process of the attack, thus completing the description of network security event. However, in the edge-cloud environment, the control logic and data forwarding of network devices are separated from each other. The control layer is responsible for the centralized management of network edge nodes. After acquiring the entire network topology, it can automatically generate a visualized network structure. This architecture extends traditional cloud computing architecture to the edge of the network, helping to handle some latency-sensitive service requirements, especially for most IoT applications. Therefore, security analysts can grasp the connection status of the devices on the entire network in the control domain. This network topology generation method based on the control layer information is directly and efficiently, which can greatly simplify the description of security events in the edge-cloud environment. At the same time, the separate structure also hides specific details of the underlying network device. Petri-net, as a formal description tool, can be used to describe such structure. Among existing security event description methods, the CORAS modeling tool has the advantages of graphical description, reusability and refinement description. And it also provides analysis guides to guide the operation steps. Based on the edge-cloud environment, this paper combines the advantages of CORAS modeling and analysis with Object-oriented Petri-net theory, and proposes a COP (CORAS-based Object-oriented Petri-net)-based Intelligent Applications security event description method. Experiments verify that this method is suitable for describing the complexity and dynamics of security events in edge cloud environment.

Keywords: Security event description, CORAS modeling, Petri-net

Introduction

Edge-Cloud computing is the product of ICT convergence, which can meet the development needs of future HD video, VR/AR, Industrial Internet and V2X business. In order to better understand the security events of the edge cloud network and evaluate network security from system perspective, a security event description method is needed.

The technology for describing network security events has become one of focus research fields in Edge Cloud. A command-level anomaly detection method with matrix and color is proposed to represent anomaly situation [1]. Such matrix-based description method is also used in fields such as IP address-based traffic description [2]. The description tool developed by Junlong Zhou implements dynamic resource descriptions with fault tolerance for data-intensive meteorological workflows in cloud [3]. There are many similar methods [4, 5] that can help describe large-scale network security faults, such as worm attacks, DDoS attacks, and network scanning attacks. These methods can also describe network traffic on parallel coordinate planes, and realize attack visualization. Ref. [6] designed a method for describing the correlation among network security events. This method provides a security internal correlation through a ring-shaped relationship diagram to help analysts

* Correspondence: qianmu@njust.edu.cn
[1]School of Cyber Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
Full list of author information is available at the end of the article

detect malicious behavior. With the increase of network complexity, rule-based vulnerability analysis technology is difficult to find potential penetration points in the network. In edge cloud networks, Ref. [7] explores a two-stage locality-sensitive hashing-based approach for privacy-preserving mobile service recommendation in cross-platform edge environment. A strategy model is proposed to establish the connection between two devices. Ref [8] used CORAS framework to link prediction in paper citation network to construct paper correlated graph. The CORAS framework has the advantages of graphical description, good reusability, and fine description [9–15]. However, the formal description ability is insufficient. And the lack of dynamic analysis capabilities restricts its application in large-scale edge cloud network security analysis and evaluation.

In addition, most of the network topology are usually obtained based on the graph traversal algorithms, and the connection relationship is used to describe the occurrence of the attack. What's more, the description of the network security event is completed. In the edge cloud network, the control logic and data forwarding of network equipment are separated, and the control layer is responsible for the centralized management of network nodes. The control layer can obtain the entire network topology and use the entire network topology to automatically generate a visualized entire network structure. Security analysts can grasp the connection status of the entire network devices in the control domain. The method of network topology generation based on control layer information is directly and efficiently, which greatly simplifies the description of security events in edge cloud networks. At the same time, such a separated structure also allows the specific details of the underlying network equipment to be hidden, thereby forming an abstract, virtual, flat structure. As a formal description tool, Petri-net can be used to describe such structures [16–20]. Among existing security event description methods, CORAS modeling tools have the advantages of graphical description [21–24], good reusability, and fine-grained description. They also provide analysis guides to guide operation steps.

So, based on the structure of the edge cloud network, this paper combines the advantages of CORAS modeling and analysis ideas with Object-oriented Petri-net theory, and then proposes a COP (CORAS-based Object-oriented Petri-net) security event description method to model the complexity and dynamics of intelligent applications security events.

Security risk assessment methods

(1) Attack Trees Analysis (ATA). ATA is an analytical method for exploiting system weaknesses from the perspective of an attacker [25]. It uses the tree structure to describe the possible attacks on the system. Because most risk assessment methods need to make assumptions based on existing information, the accuracy of the assessment will be limited by the accuracy of the hypothesis. To ensure the best results, the conclusions drawn from the attack tree analysis need to be compared to other analysis results or assessed by experts. However, building a 100% accurate attack tree model is almost impossible. And this step will greatly increase the complexity of the method. The evaluator needs to know the extent of the assessment and make the attack tree model good enough. In order to prevent this step from consuming too many resources, the following three conditions need to be considered:

a. Defender's system has vulnerabilities.
b. Attackers need to have enough ability to exploit these vulnerabilities.
c. The expected benefit is the motivation for the attack, and the attacker can gain benefits by attacking.

The main advantage of ATA is that it can be easily rewritten according to the needs and characteristics of the organization. This method can also conclude which attacks are most likely to occur in terms of the entire system. From a certain perspective, security is not a result but a process, and ATA can form a basic understanding of this process.

(2) Failure Tree Analysis (FTA). FTA is a top-down assessment method. It uses a tree diagram to organically link system security failures to internal failures. In the fault tree, the root node indicates a fault, and the leaf node indicates an event that may cause a fault. Different layers are linked by logic gate symbols and the upper layer probability is calculated according to the underlying probability. However, the fault tree cannot analyze the hazards and risks caused by the fault time, so it can only be used as a method of some parts in the risk analysis.

(1) Failure Mode Effect and Criticality Analysis (FMECA). FMECA is a single component failure mode analysis and hazard analysis tool. Its purpose is to reduce the possibility of failure and improve the reliability of system operation [26]. FMECA is a bottom-up approach that identifies faults in the form of a discussion and records the results in a table. The disadvantage of this approach is that there are too many limitations in a single unit, ignoring the connections and commonalities between the units.

(2) Hazard and Operability Study (HAZOP). HAZOP is a structured inspection method for potential hazards of the system. It uses structured checks to determine the abnormal operation of the system from normal design. And the purpose of this method is to identify threats. The HAZOP analysis is conducted in the form of a discussion, and the analyst uses a variety of analysis techniques to collect system information into the document as an input to the analysis. In the analysis process, some system-related questions are used to form special guidance words to help improve the comprehensiveness of the analysis. This not only ensures the analysis results are consistent with the characteristics of the system, but also adds extra information. The analysis results are saved in a table format.

(3) Petri-net. Petri-net is a graphical modeling tool based on mathematical theory. Petri-net can automatically control the state of the system by changing the state of the token in the system to describe a dynamic complex system. It is commonly used in the field of security analysis to analyze security threats transmitted through the system.

(4) Analytic Hierarchy Process (AHP). AHP uses a hierarchical approach to quantify empirical judgments and form quantitative decision values. However, this method is subject to human factors, and there are fluctuations between various indicators and lack of consistency.

The traditional method lacks comprehensive considerations for security risk technology and management. A single assessment method cannot objectively and accurately reflect the security status of complex information security system engineering. This comes to analysis comprehensive security risk assessment methods. Comprehensive risk assessment methods have a set of implementation steps and theoretical systems, and their solutions for risk assessment are more comprehensive than traditional risk assessment methods. They may contain some traditional analytical methods. However, in addition to these, they generally follow certain security standards and also provide solutions to systemic risks.

(1) CCTA Risk Analysis and Management Method (CRAMM). CRAMM is a security service framework system proposed by the British government. It is an automated qualitative assessment method, but in order to achieve good results, experts need to participate in the assessment. The purpose of this method is to assess the security of related information systems and networks. To achieve the goal, the method focuses on three aspects:
(1) Identify assessment assets.
(2) Identify threats and vulnerabilities and calculate risks.
(3) Identify and give countermeasures according to priority.

(2) Operationally Critical Treat, Asset, and Vulnerability Evaluation (OCTAVE). OCTAVE is a method developed by Carnegie Mellon University to define the security risks of assessing information within a system organization. This approach provides a new approach to information security for large organizations. OCTAVE enables organizations to view security issues from a risk-based perspective and describe the technology in a commercial perspective. OCTAVE Allegro is a new version that was published in 2007. This version is based on the two previous versions, OCTAVE Original (1999) and OCTAVE-S.

OCTAVE Allegro focuses on information assets. One of the advantages of using OCTAVE Allegro is that it can be conducted in the form of a seminar. It provides the required collaborative environment, the necessary guides, work forms and questionnaires. All of the above-mentioned content is free. OCTAVE Allegro consists of four stages and eight steps. The results of each step are recorded by the worksheet and used as input for the next step.

(3) Consultative Objective and Bi-functional Risk Analysis (COBRA). COBRA is a risk analysis method created by C&A. COBRA aims to provide organizations with a way to self-assess their own information technology without additional consultants. COBRA follows the guidance of ISO 17799 and its risk assessment process includes two aspects. One is COBRA Risk Consultant, and the other one is ISO Compliance.

COBRA Risk Consultant is a questionnaire-based computer program that contains a number of standardized questions to gather information about asset types, vulnerabilities, threats, etc. This approach generates appropriate recommendations and solutions by evaluating relevant threats. COBRA Risk Consultant is designed based on self-assessment, which can be used without relevant knowledge and without expert involvement. The reports generated by COBRA Risk Consultant are professional business reports that can be read by security professionals or non-professionals. ISO Compliance contains standard questions related to the broad categories specified in the ISO 17799 standard.

(4) Control Objectives for Information and related Technology (COBIT). COBIT is proposed by ISAKA. It is the most internationally recognized and most authoritative standard for security and information technology management and control. And It has been developed to COBIT 5.

(5) A Platform for Risk Analysis of Security Critical Systems (CORAS). CORAS was formally proposed by Greece, Germany, Norway and the United Kingdom in 2003. It is a qualitative risk assessment method and provides a complete set of graphical language to model threats and risks.

There is no unified evaluation system for security risk assessment methods. This paper presents a simple assessment framework for comparing the various methods described above. The framework evaluates the above methods from the eight aspects: data requirement (DR), tool support (TS), operability(O), application cost (AC), application range (AR), method type (MT), policy assurance (PA) and support organization (SO). This helps relevant organizations to select appropriate security risk assessment methods based on their needs. Table 1 shows the comparison results.

Component-based CORAS and petri-net

CORAS is a modeling analysis description method formed by combining some security analysis technologies (such as HazOp, FTA, FMEA, etc.) and system development technologies (such as UML) [27]. CORAS is a graphical and model-based method that has the following advantages:

(1) CORAS can provide a precise description of the target system. Its syntax and all related security features are easy to use;

(2) The graphical representation of CORAS information enhances the communication and interaction of each participant in the analysis;

(3) CORAS facilitates the documentation of risk assessment assumptions and assessment results.

CORAS can be divided into three different components:

(1) The CORAS Risk Modeling Language: This part includes the graphical grammar, textual grammar of the CORAS icon and related semantics;

(2) The CORAS Method: This part includes a step-by-step description of the safety analysis process and a guide to constructing a CORAS chart;

(3) The CORAS Tool: This part includes tools for documenting, maintaining, and reporting the results of risk analysis.

In addition to including descriptions and analytical methods, the CORAS approach also takes into account international standards for risk management, such as the Australian/New Zealand Standard for Risk Management, AS/NZS 4360:2004, ISO/IEC 17799, ISO/IEC 13335, the ISO Reference Model for Open Distributed Processing, and so on.

CORAS has gradually started to develop toward component-based risk analysis [18]. Component-based refers to a way of thinking or development rather than a specific technology. At its core, for complex system analysis tasks, reusable components should be utilized to reduce the workload, rather than analyzing from scratch. It contains development techniques including syntax, rules, and implementation guidelines for specifying the behavior and system architecture of components. This standardizes the incremental analysis of the system. A simple example is given below to illustrate how component-based CORAS describes and analyzes

Table 1 Assessment to the security risk assessment method

Name	TS	O	AC	AR	MT	PA	SO
ATA	–	easy	low	small	Qualitative	low	–
FTA	–	easy	low	small	Qualitative	low	–
FMECA	–	medium	medium	small	Qualitative	low	–
HAZOP	–	easy	medium	medium	Qualitative	low	–
Petri-net	–	difficult	medium	medium	Quantitative	low	–
AHP	–	easy	low	medium	Comprehensive	high	–
CRAMM	–	difficult	high	wide	Quantitative	low	UK
OCTAVE	Y	difficult	low	wide	Comprehensive	high	CMU
COBRA	Y	medium	medium	wide	Qualitative	high	C&A
COBIT	Y	difficult	medium	wide	Qualitative	high	ISAKA
CORAS	Y	medium	medium	wide	Comprehensive	high	EU

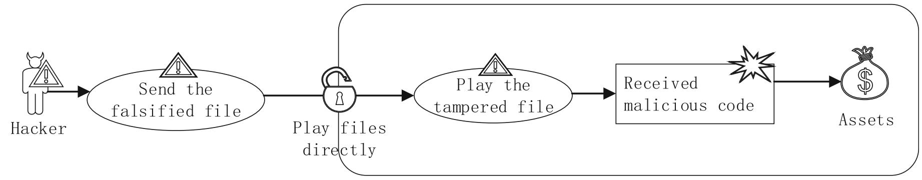

Fig. 1 CORAS modeling analysis of threat scenarios

Security Events. An example of modeling and analysis of a threat scenario is shown in Fig. 1.

Hackers have grasped the fragile point that the player can directly play files. By sending tampered music files, the media player buffer overflow vulnerability is used to threaten user-related media assets. When the receive file operation is invoked, the channel interface calls the tampered music file from the interface of the media player. Once the file is played, it will use a buffer overflow vulnerability to overwrite the pointer address to point to malicious code, threatening the user's assets. In the above threat scenarios, scenarios, risks, and threat assets are defined as individual component objects. The description of the entire Security Event is done by connecting the calling relationships of the interfaces between the objects. The entire description process is very clear and concise, which helps participants involved in the risk analysis and evaluation to understand and communicate the entire event. At the same time, related scenes are also very convenient for documenting preservation. If a new threat scenario is created, the entire modeled part is not necessary to make major changes, so the reusability of the model is also guaranteed. However, from the above examples, CORAS can also be found to have shortcomings such as insufficient formal description ability, excessive subjectivity, and insufficient dynamic analysis capability.

Petri-net is a graphical description method based on mathematical theory. It is a special directed graph consisting of the "place", "transition" and "connection" relationship. And it uses Token to describe the state changes in the graph. The basic Petri-net is defined as following:

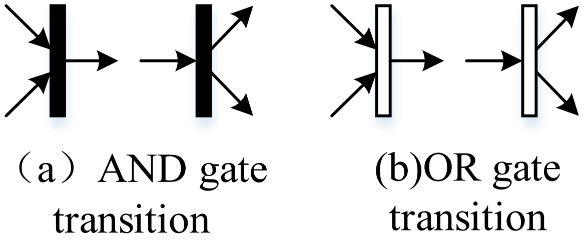
（a）AND gate transition
(b)OR gate transition

Fig. 2 Gate transition symbol

Definition 1 Basic Petri-net is a triple:

$$PN = (P, T, F) \tag{1}$$

Where:

(1) P is a finite set of spaces that represent the state of the system; T is a finite set of transitions that represent changes in behavior;
(2) $P \cup T \neq$, $P \cap T =$;

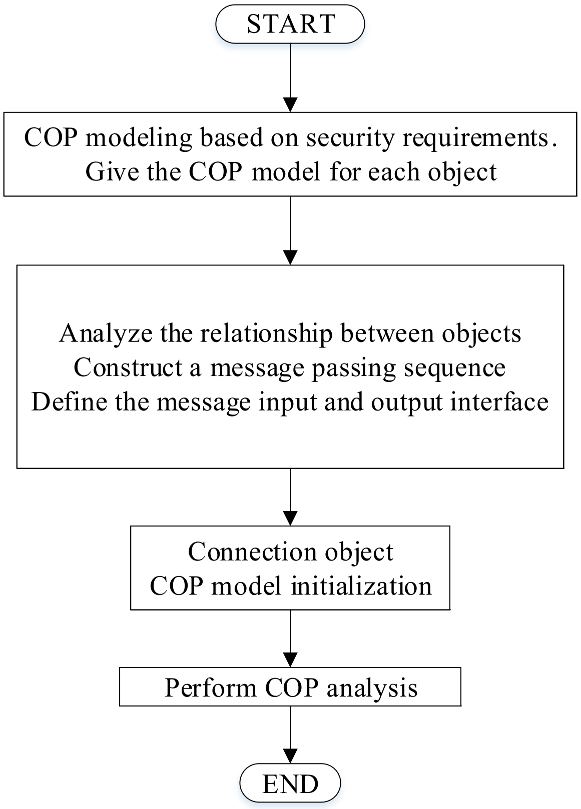
START
COP modeling based on security requirements. Give the COP model for each object
Analyze the relationship between objects
Construct a message passing sequence
Define the message input and output interface
Connection object
COP model initialization
Perform COP analysis
END

Fig. 3 COP modeling steps

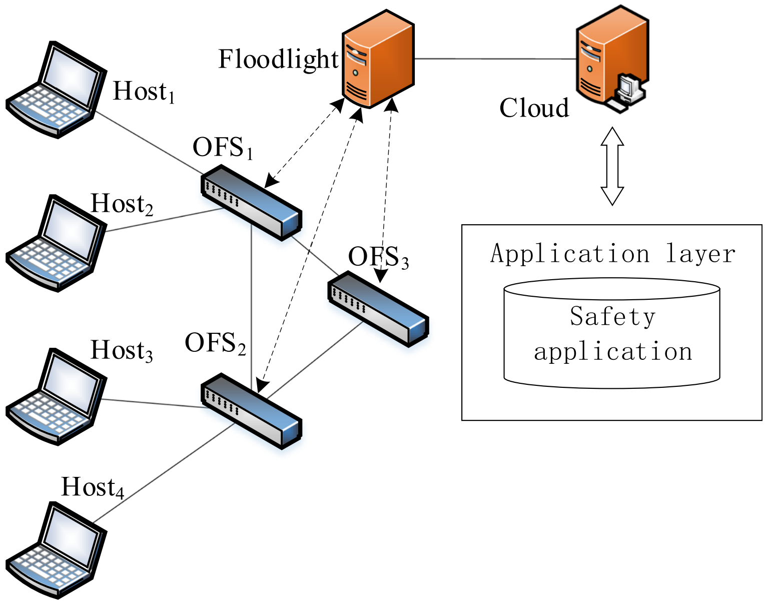

Fig. 4 Experimental edge-cloud environment topology

(3) $F \subseteq (P \times T) \cup (T \times P)$ is a Solitary Sets. It is the "connection" relationship of Petri-net, connecting places and transitions;

(4) $Dom(F) \cup Cod(F) = P \cup T$;

$$Dom(F) = \{x|\exists y : (x, y) \in F\}, Cod(F) \\ = \{x|\exists y : (y, x) \in F\}$$

Place sets and transition sets are the basic building blocks of a petri. Connection relationships are constructed from these two sets. Each place represents a storage location for a resource. Transition is based on voluntary connection and is governed by connection relationships. Therefore, transition can only be directly related to the place:

$$F \subseteq (P \times T) \cup (T \times P)$$

$Dom(F) \cup Cod(F) = P \cup T$. This means that there are no resources that do not participate in any "transitions" and no "transition" that cause resource "connection".

The Petri-net model unifies graphics and semantics. Its expression is intuitive and its content structure is rigorous. It is easy to describe the relationship of system connection. Petr-net is ideally suited to describe the characteristics of various real-time, dynamic cyber-attacks for risk description and analysis.

Combining the advantages of CORAS modeling and analysis ideas with the Object-oriented Petri-net theory, this paper proposes a COP (CORAS-based Object-oriented Petri-net) security event description method in an edge cloud environment. This method is suitable for modeling and describing the complexity and dynamics of network security events.

COP modeling method

Definition 2 COP is a security event description process that defines it as a triple:

$$COP = \{SP, OG; OF\} \tag{2}$$

where,

(1) $SP = \{sp_1, sp_2, \dots, sp_n\}$ is a sub-process of the COP evaluation process, which can be regarded as a special place;

(2) $OG = \{og_1, og_2, \dots, og_n\}$ is a collection of Outer Gate Transitions between sub-processes. In order to comply with the description of COP, this paper extends the transition T to G. G can be regarded as a special gate transition, and this transition has the characteristic of gate. This paper introduces two different gate transitions, as shown in Fig. 2:

Table 2 Packet transmission information in the experiment

Number	Send Content
p_{51}	Edge-host1 sends ICMP packets to Cloud
p_{52}	Edge-host2 sends ICMP packets to Cloud
p_{61}	Edge-host3 sends TCP packets to Cloud
p_{62}	Edge-host4 sends TCP packets to Cloud

Table 3 Attack probability assignment table

Assignment	Identification	Threat frequency	Frequency Range	λ_i
5	Very high	occur frequently	>50 % · *lbor*	1
4	High	Very likely to happen	(20 % ~50%) · *lbor*	0.5
3	Medium	likely to happen	(10 % ~20%) · *lbor*	0.2
2	Low	Less likely to happen	(5 % ~10%) · *lbor*	0.1
1	Very low	Extremely rare	<5 % · *lbor*	0.01

(3) $OF = \{of_1, of_2, \ldots, of_n\}$is a collection of all Outer Flows outside the sub-process, corresponding to the dependencies between the subprocesses.

Definition 3 The COP sub-process sp_i is internally defined as a triple:

$$inner(sp_i) = \{P, IG; IF\} \tag{3}$$

where,

(1) $P = \{p_1, p_2, \ldots, p_n\}$ is a collection of all the places in the sub-process sp_i;
(2) $IG = \{ig_1, ig_2, \ldots, ig_n\}$ is a collection of all Inner Gate Transitions within sub-process sp_i;
(3) $IF = \{if_1, if_2, \ldots, if_n\}$ is a collection of Inner Flows between all the libraries and transitions in sub-process sp_i.

Definition 4 Sub-process sp_i internal and external communication is defined as a four-tuple, defined as follows:

$$outer(sp_i) = \{IM, OM, OG; OF\} \tag{4}$$

Where,

(1) $IM = \{im_1, im_2, \ldots, im_n\}$ is a collection of all In-message queues outside of sub-process sp_i;
(2) $OM = \{om_1, om_2, \ldots, om_n\}$ is a collection of all Out-message queues outside of sub-process sp_i;
(3) The definition of $OG = \{og_1, og_2, \ldots, og_n\}$ and $OF = \{of_1, of_2, \ldots, of_n\}$ is defined in Definition 2;

Message passing between objects is triggered by the transition which is in the connection message.

In the description of modeling using the COP method, the COP model of each object is given first. Secondly, the message input and output interface are defined according to the connection relationship between the objects. Then connect the interfaces according to the connection relationship and initialize the COP model. Finally, a COP analysis is performed. The COP modeling steps are shown in Fig. 3.

The COP model initialization algorithm is as follows:

Conversion rules:

(1) Each method in each class is represented by a pair of places. Given a method M, a token in its input place indicates that M has been called; a token in its output place indicates that M has completed execution.
(2) The state in CORAS is represented by the place in COP, and the transition in the CORAS state diagram are represented by the transition in COP.
(3) The relationship between state and transition in CORAS is represented by the arc between corresponding place and transition in COP.
(4) Events and actions in CORAS correspond to service requests, service completion and confirmation of service completion in COP. A pair of places

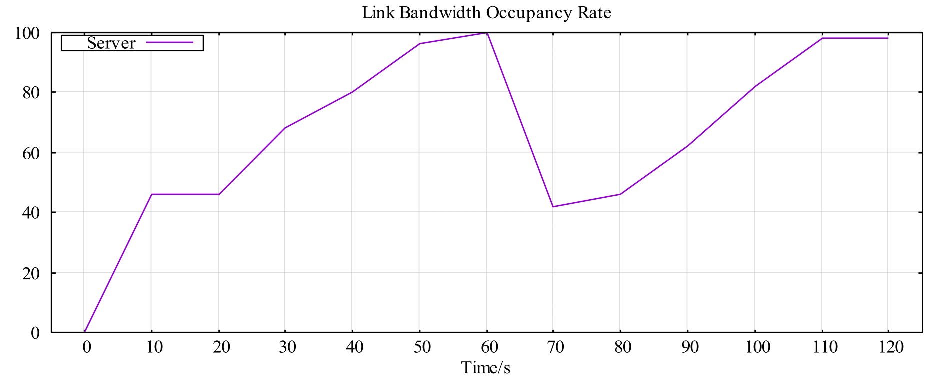

Fig. 5 Cloud link bandwidth occupancy rate

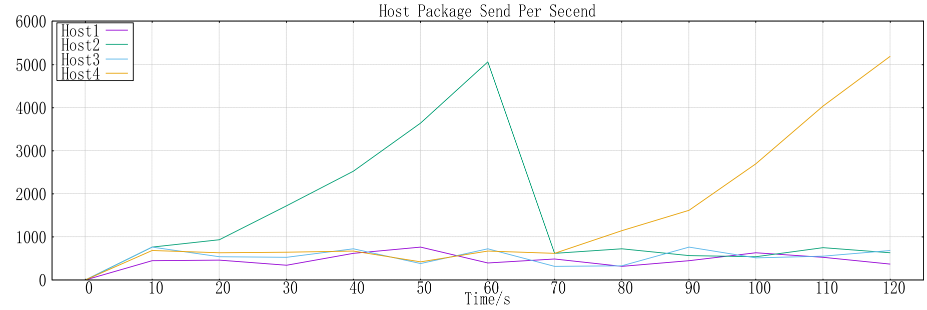

Fig. 6 Edge-host packet transmission frequency

corresponding to the services provided by the class (ie, methods of the class) have been added in the conversion of the class in step (1), and the places corresponding to the requested service must also be added.

(5) The individual can be used to represent the token.
(6) The combination of place pairs uses the following rule: When one of two place pairs with the same name provides services and the other pair requests services, merge them into a pair of places and maintain all previous connections (arcs).

The transformation of the CORAS model to the COP model has changed the deficiencies of traditional methods. The above rules enable the CORAS model to fully describe the concurrency, synchronization, and conflict situations of security events in the edge cloud system.

Experimental cases and analysis

Different from the distributed management of traditional network devices, the unified management of the control layer in the edge cloud network will cause new threats. DDoS attacks against network controllers is an example [5]. In order to verify the feasibility and effectiveness based on COP, this paper uses SDN technology to build a simulation environment as shown in Fig. 4. The paper carried out the DoS attack simulation and described the security events triggered. The device layer includes multiple Edge-hosts, OpenFlow switches, controllers, and application servers. The control layer uses Floodlight as the SDN controller. The application layer runs a security application. The simulation software is MININET.

Common DOS attacks include ICMP packet attacks and TCP request attacks. These two attacks achieve the effect of denial of service attacks by consuming bandwidth resources and link resources in the network. The data packet transmission information in the experiment is shown in Table 2. After the request, the stream data that is not matched by the OFS flow table will be packaged and delivered to Floodlight. After the Floodlight identifies the packet, it passes the packet to the application layer security application for processing. The security app sends the specified protection policy to Floodlight. Floodlight will send the corresponding new flow table and settings to OFS. Finally, the OFS processes the packet according to the new command. The experiment collects the link bandwidth occupancy rate (lbor: link bandwidth occupancy rate), the client packet transmission rate (psps: package send per second), and the server-side packet reception rate (prps: package received per second) as statistical indicators.

The statistical indicator includes the Cloud packet reception frequency prps. This frequency response corresponds to the attack strength and credibility of the attack. The greater the number of attacks, the more

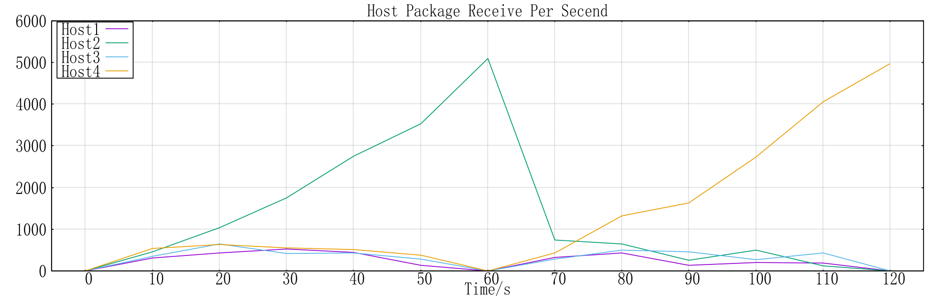

Fig. 7 Cloud packet receiving frequency

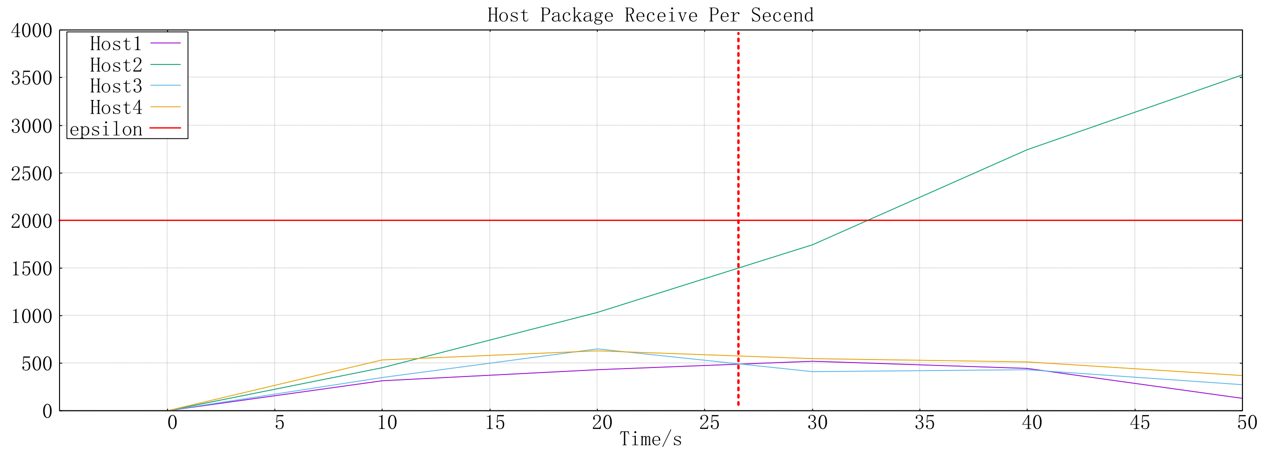

Fig. 8 Cloud packet receiving frequency fragment

likely the attack is to be a real intrusion. A gate threshold value ε, which is dynamically adjusted by the application layer security application, can be set as a reference value for the number of alarms, whereby the probability λ of occurrence of a certain attack can be calculated.

$$\lambda_i = \begin{cases} \frac{prps_i}{\varepsilon_i} & if\,(n_i < \varepsilon_i) \\ 1 & otherwise \end{cases} \tag{5}$$

For an attack, when the data is less than the set gate threshold ε_i, the probability value λ_i of the attack is represented by $\frac{prps_i}{\varepsilon_i}$. When the threshold ε_i is exceeded, the probability value λ_i of the attack is considered to be 1.

It is also possible to divide the transmission frequency $prps_i$ into different intervals according to the provisions of GB20984–2007 as the basis for the attack threat assignment. The division between intervals can be divided into non-equal divisions, as shown in Table 3. In this way, the probability λ of an attack occurring is calculated.

The experiment uses the first attack probability calculation method as the evaluation basis. First, Edge-host$_1$ sends ICMP packets at a lower frequency. Edge-host$_3$ and Edge-host$_4$ send TCP packets at a lower frequency. Edge-host$_2$ sends ICMP packets at increasing frequency until it occupies all of the link bandwidth and then drops to normal. Edge-host$_4$ then sends TCP packets with increasing frequency until it occupies all of the link bandwidth and then drops to normal. The Cloud link bandwidth occupancy, Edge-host packet transmission frequency, and Cloud packet reception frequency in the experiment are shown in Fig.5, Fig.6 and Fig.7.

It can be seen that as the two DoS attacks progress, the bandwidth is heavily occupied, normal traffic cannot be sent, and the connection cannot be established. We choose the 27th second, as shown in Fig. 8, as the time point to analyze the experiment result. In the figure, the red horizontal line is the gate threshold value ε, and the red vertical line is the 27th second of the experiment. Assume that both the ICMP gate threshold ε_1 and the ε_2 of TCP are 40% of the bandwidth occupied by the Cloud packet.

Table 4 lists the data on the likelihood of an attack occurring at the red vertical dashed line.

Attack scene COP modeling definition

The moment is modeled and analyzed according to the COP modeling step. The process is as follows:

The moment contains five sub-processes, in which Edge-host$_1$ ~ Edge-host$_4$ are recorded as potential attack initiators as sub-process sp_1~sp_4. Two different potential attack behaviors ICMP and TCP belong to two different

Table 4 The possibility of an attack at this moment

Number	Packet acceptance frequency *psps*	Gate threshold value ε	Attack possibility λ
p_{51}	489	2000	0.244
p_{52}	1507	2000	0.753
p_{61}	502	2000	0.251
p_{62}	587	2000	0.294
p_{51} p_{52}	total: 1996	total: 3085	Proportion: 0.647
p_{61} p_{62}	total: 1089		Proportion: 0.353

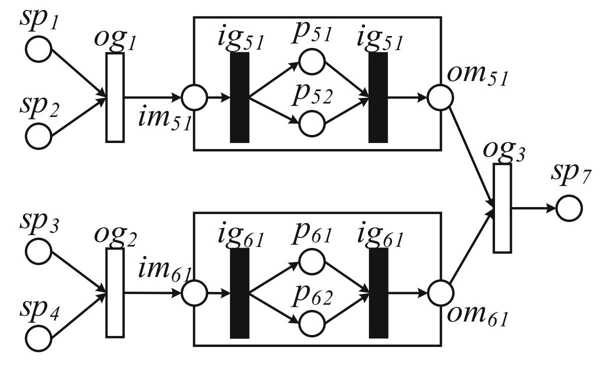

Fig. 9 COP model generated based on attack scenario information

sub-processes sp_5 and sp_6. The attacked server is the target Recorded as sub-process sp_7.

(1) Initialize the COP network, assign Φ;
(2) New a sub-process sp_1. sp_1 does not have a library and transitions that need to be described in detail. And add sp_1 to the COP network. Similarly, new a sub-process $sp_2 \sim sp_4$. $sp_2 \sim sp_4$ does not have a library and transitions that need to be described in detail. $sp_2 \sim sp_4$ is added to the COP network;
(3) Create a new subprocess sp_5. The behavior im_{51} that initiates the attack within the A sub-process is taken as the input of sp_5.It can be seen from Table 1 that sp_5 includes p_{51}, p_{52} suspected of initiating an ICMP (ig_{51}) attack. Since p_{51}, p_{52} belong to the same ICMP attack ig_{51}, they conform to the "AND" relationship. So, add the AND transition ig_{51} to sp_5. Finally, the consequences of the attack are taken as the output om_{51} of sp_5 and added to sp_5. Calculate the internal IF of sp_5. Add the internal flow relationship IF to sp_5. Add sp_5 to the COP network. Similarly, modeling can get sp_6 and add sp_6 to the COP network.
(4) It can be seen from Table 2 that $sp_1 \sim sp_4$ randomly initiate attacks can make an affection of sp_7. So, there is a logical OR relationship between the attack behaviors. Add OR gate transitions og_1, og_2 and og_3 to the COP. Calculate OF based on the relationship between the elements and add to the COP.
(5) Improve the COP network;

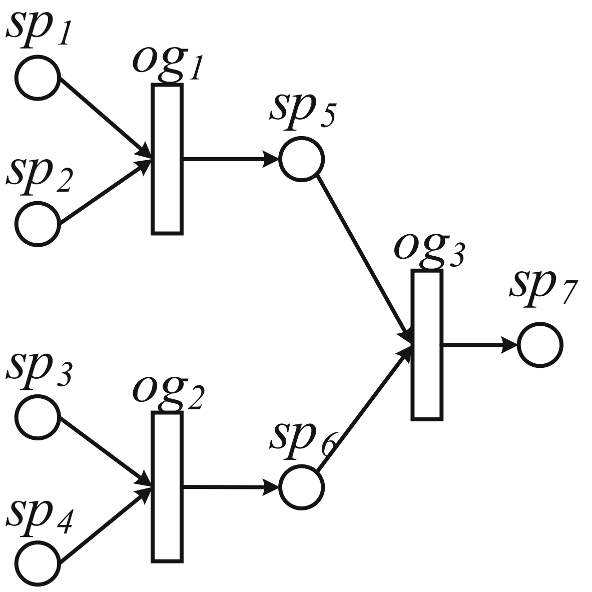

Fig. 10 Dynamically scaled COP model

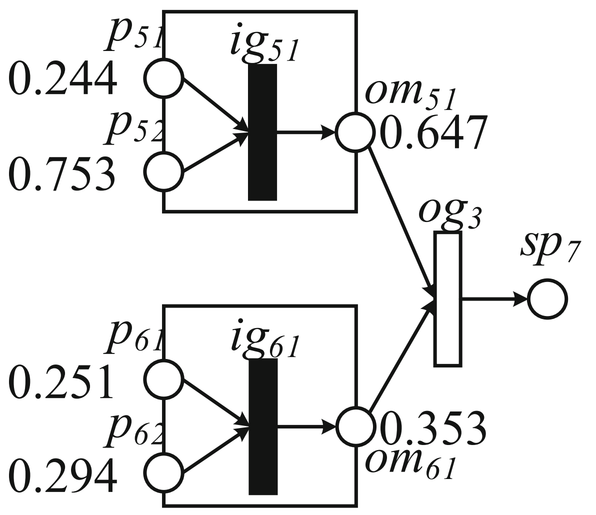

Fig. 11 COP model with attack probability

The modeling results are shown in Fig.9:

COP method analysis

In the qualitative description, we want to know the type of attack, rather than the specific attack details, so we can compress the sub-process. This kind of sub-flow is independently scaled. The describing way of the details like packing up and opening is completed. The description of different refinement levels is realized. The sub-processes that have completed the analysis at the same time can be saved independently as the analysis results. Portions of the same analysis content encountered in

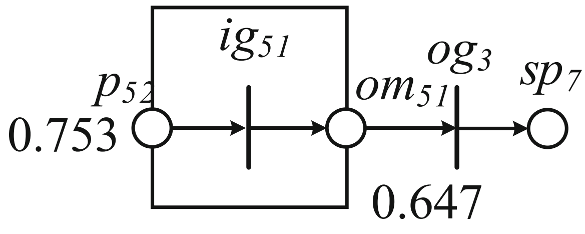

Fig. 12 COP model with an attack gate threshold of 0.6

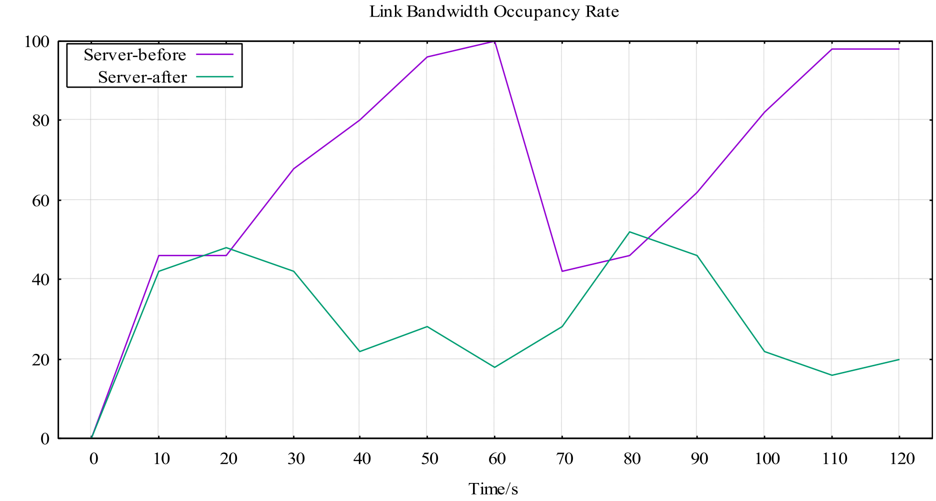

Fig. 13 Cloud link bandwidth usage

other analyses can be directly replaced to achieve reuse of the model.

The qualitative results are shown in Fig.10. It can be clearly seen that $sp_1 \sim sp_4$ initiates two different attacks sp_5 and sp_6 against sp_7. The results of each attack analysis can be saved separately to implement model reuse. The attack process can be scaled independently to achieve a different level of description.

In the quantitative description, the analysis can be performed based on the connection relationship in the COP network. Suppose that the risk of an object being attacked is F. From the definition of COP, it can be seen that in the case of AND gate transition, the value of F is determined by the sum of the possibility of initiating attack precondition. In the case of an OR gate transition, the value of F is determined by the maximum probability of initiating an attack precondition. Bring the possibility of potential attack at this moment in Table 3 to Fig.9. The possibility of each attack content and attack type is shown in Fig.11.

According to the definition, the risk value of the possible attack node sp_7 is calculated as follows:

$$\begin{aligned} F(sp_7) = \; & MAX[0.647 \cdot SUM(0.244, 0.753), 0.3 \cdot SUM(0.251, 0.294)] \\ = \; & MAX[0.997, 0.545] \\ = \; & 0.997 \end{aligned} \tag{6}$$

The overall risk value of node sp_7 is 0.997. In this way, the risk value of the attacked party can be calculated, and the dynamic quantitative analysis can be further implemented by modifying the set gate threshold.

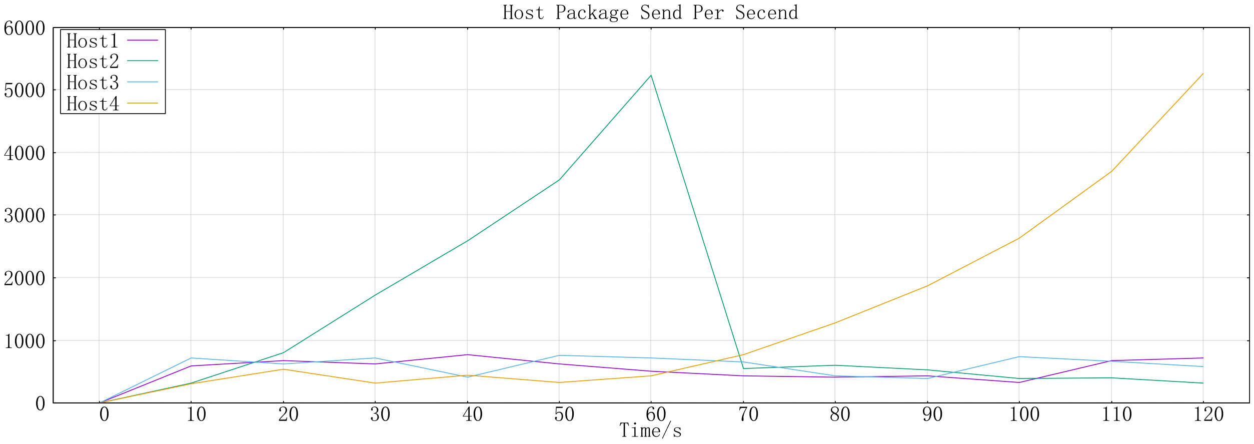

Fig. 14 Edge-host packet transmission frequency

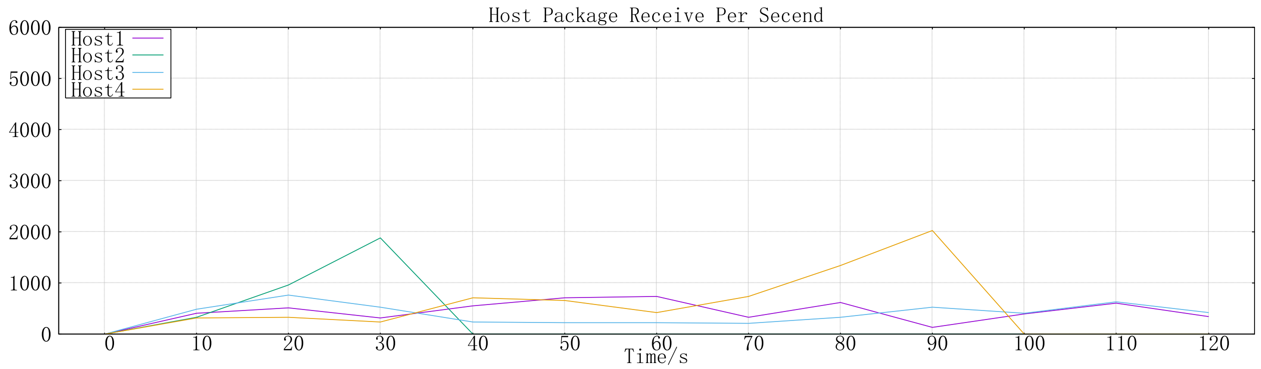

Fig. 15 Cloud packet receiving frequency

Assuming that the probability of attack to be analyzed exceeds 0.6, the new COP model is shown in Fig.12:

Among them, ig_{51}, og_3 degenerates into a normal gate transition. At this time, the risk value of sp_7 is:

$$F(sp_7) = MAX[0.647 \cdot SUM(0.753)] = 0.487 \tag{7}$$

Once it is detected that the actual risk value of the relevant asset exceeds the acceptable risk value (assumed to be 0.5), the application-level security application performs the flow table update according to the set rules. Then, depending on the magnitude of the risk value, a new forwarding path can be set to offload, limit or block certain stream data. In the experiment, if the gate threshold is exceeded, the forwarding request of the relevant network segment is discarded, and the stream data is discarded. After setting the rules, the Cloud link bandwidth occupancy, Edge-host packet transmission frequency, and Cloud packet reception frequency are shown in Fig.13, Fig.14, and Fig.15. It can be seen that in the case where the transmission packet law is unchanged in the simulation network, the transmission source with the attack intention is blocked, the link occupancy rate of the Cloud end is significantly reduced, and the normal service is guaranteed.

Brændeland G et al. used the EBNF paradigm to describe CORAS and use the paradigm to calculate the probability of security risk based on the description results [28]. However, this method fails to take advantage of the graphical description of CORAS. And the safety risk probability calculation method is mainly through static evaluation by experts, so the evaluation results are not objective enough. COP inherits CORAS's graphical description, reusability and refined description of the advantages, and uses Object-oriented Petri-net to increase the advantages of formal description, scalability and dynamic verification. At the same time, the data source of CORAS quantitative analysis is transformed from subjective expert evaluation into objective scanning analysis, which reduces the human factors in the analysis process and makes the results more reliable.

Conclusion

This paper combines a model-based static Security Event modeling description method CORAS and Object-oriented Petri-net, and proposes a COP-based security risk modeling method. Compared with the existing model-based methods, the proposed COP model not only inherits the existing model's extensibility, reusability, and refinement description, but also enhances the formal description and dynamic analysis capabilities. In the edge-cloud environment, the COP description of the entire network in the control domain can be directly generated based on the control layer information, and its efficiency is far superior to the topology discovery technology in the traditional network. The attack simulation experiment proves that COP can effectively describe the cloud environment security incidents, and can further carry out risk strategy response based on the description results.

Abbreviation

COP: CORAS-based Object-oriented Petri-net

Acknowledgements

We want to thank the authors of the literature cited in this paper for contributing useful ideas to this study.

Authors' contributions

Qianmu Li, Xiaochun Yin, Shumei Meng, Yaozong Liu and Zijian Ying have written this paper and have done the research which supports it. Qianmu Li, Xiaochun Yin, Shumei Meng has collaborated in the conception, research and design of the paper. The authors read and approved the final manuscript.

Authors' information

Qianmu Li received the BSc and PhD degrees from Nanjing University of Science and Technology, China, in 2001 and 2005, respectively. He is currently a full professor with the School of Cyber Science and Engineering, Nanjing University of Science and Technology, China. His research interests include information security and data mining. He received the China Network and Information Security Outstanding Talent Award in 2016, and Education Ministry Science and Technology Awards in 2012.

Xiaochun Yin received the B.S. degree in education and technology from Qufu Normal University, Qufu, China in 2004, and received the M.S. degree in education and technology from Nanjing Normal University, Nanjing, China in 2007, and received the Ph.D. from Dongseo University, Korea in 2015. She is now working as an associate professor in Weifang University of Science &Technology China. Her research interests include network security, IoT security, authentication protocol and agricultural intelligence systems. She has published over 20research papers in international journals and international conferences.

Shunmei Meng received her PhD degree in Department of Computer Science and Technology from Nanjing University, China, in 2016. Now, she is an assistant professor of School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing, China. She has published papers in international journals and international conferences such as TPDS, ICWS, and ICSOC. Her research interests include recommender systems, service computing, and cloud computing.

Yaozong Liu received the Ph.D. degree from the Nanjing University of Science and Technology, China, in 2016. He is currently a Lecturer with the Intelligent Manufacturing Department, Wuyi University, China. His research interests include data mining and network security.

Zijian Ying received the BSc. degree from Nanjing University of Science and Technology, China, in 2019. He is now a graduate student at Nanjing University of Science and Technology. His research interests include data mining and network security.

Funding
This work was supported in part by the 2019 Industrial Internet Innovation and Development Project from Ministry of Industry and Information Technology of China, 2018 Jiangsu Province Major Technical Research Project "Information Security Simulation System", Fundamental Research Funds for the Central Universities (30918012204), Military Common Information System Equipment Pre-research Special Technology Project (315075701).

Availability of data and materials
Not applicable.

Competing interests
The authors declare that there is no conflict of interest regarding the publication of this manuscript.

Author details
[1]School of Cyber Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China. [2]Facility Horticulture Laboratory of Universities in Shandong, WeiFang University of Science & Technology, ShouGuang 262700, China. [3]Nanjing Liancheng Technology Development Co., Ltd., Jiangsu Graduate Workstation of Nanjing University of Science and Technology, Nanjing 210008, China. [4]Intelligent Manufacturing Department, Wuyi University, Jiangmen 529020, China. [5]Jiangsu Zhongtian Technology Co. Ltd., Nantong 226009, China.

Received: 10 February 2020 Accepted: 13 April 2020
Published online: 01 May 2020

References
1. Li Q, Meng S, Wang S, Zhang J, Hou J (2019) CAD:command-level anomaly detection for vehicle-road collaborative charging network. IEEE Access 7: 34910–34924
2. Zhou J, Hu XS, Ma Y, Sun J, Wei T, Hu S (2019) Improving availability of multicore real-time systems suffering both permanent and transient faults. IEEE Transact Comp (TC) 68(12):1785–1801
3. Xu X, Mo R, Dai F, Lin W, Wan S, Dou W (2019) Dynamic resource provisioning with fault tolerance for data-intensive meteorological workflows in cloud. IEEE Transact Indust Informatics. https://doi.org/10.1109/TII.2019.2959258
4. Li Q, Wang Y, Ziyuan P, Wang S, Zhang W (2019) A time series association state analysis method in smart internet of electric vehicle charging network attack. Transp Res Rec 2673:217–228
5. Zhou J, Sun J, Zhou X, Wei T, Chen M, Hu S, Hu XS (2019) Resource Management for Improving Soft-Error and Lifetime Reliability of real-time MPSoCs. IEEE Transact Comp-Aided Design Integr Circuits Syst (TCAD) 38(12):2215–2228
6. Li Q, Meng S, Zhang S, Hou J, Qi L (2019) Complex attack linkage decision-making in edge computing networks. IEEE Access 7:12058–12072
7. Qi L, Zhang X, Dou W, Hu C, Yang C, Chen J (2018) A two-stage locality-sensitive hashing based approach for privacy-preserving Mobile service recommendation in cross-platform edge environment. Futur Gener Comput Syst 88:636–643
8. Liu H, Kou H, Yan C, Qi L (2019) Link prediction in paper citation network to construct paper correlated graph. EURASIP J Wirel Commun Netw. https://doi.org/10.1186/s13638-019-1561-7
9. Liu Y, Wang S, Khan MS, He J (2018) A novel deep hybrid recommender system based on auto-encoder with neural collaborative filtering. Big Data Mining Analytics 1(3):211–221
10. Zhang C, Yang M, Lv J, Yang W (2018) An improved hybrid collaborative filtering algorithm based on tags and time factor. Big Data Mining Analytics 1(2):128–136
11. Ramlatchan M, Yang Q, Liu M, Li J, Wang YL (2018) A survey of matrix completion methods for recommendation systems. Big Data Mining Analytics 1(4):308–323
12. Xu X, Liu X, Xu Z, Wang C, Wan S, Yang X (2019) Joint optimization of resource utilization and load balance with privacy preservation for edge services in 5G networks. Mobile Netw Appl. https://doi.org/10.1007/s11036-019-01448-8
13. Qi L, Zhang X, Dou W, Ni Q (2017) A distributed locality-sensitive hashing based approach for cloud service recommendation from multi-source data. IEEE J Selected Areas Commun 35(11):2616–2624
14. Xu X, Cai Q, Zhang G, Zhang J, Tian W, Zhang X, Liu AX (2018) An incentive mechanism for crowdsourcing markets with social welfare maximization in cloud-edge computing. In: Concurrency and computation: practice and experience
15. Qi L, Dou W, Wang W, Li G, Yu H, Wan S (2018) Dynamic Mobile crowdsourcing selection for electricity load forecasting. IEEE Access 6:46926–46937
16. Hong JE, Bae DH (2000) Software modeling and analysis using a hierarchical object-oriented petri net. Inform Sci Int J 130:131–164
17. Kong C, Luo G, Tian L, Cao X (2019) Disseminating authorized content via data analysis in opportunistic social networks. Big Data Mining Analytics 2(1):12–24
18. Kumar S, Singh M (2019) Big data analytics for healthcare industry: impact, applications, and tools. Big Data Mining Analytics 2(1):48–57
19. Zhang Y, Wang K, He Q et al (2019) Covering-based web service quality prediction via neighborhood-aware matrix factorization. IEEE Trans Serv Comput. https://doi.org/10.1109/TSC.2019.2891517
20. Zhang Y, Cui G, Deng S et al (2018) Efficient query of quality correlation for service composition. IEEE Trans Serv Comput. https://doi.org/10.1109/TSC.2018.2830773
21. Li Q, Meng S, Zhang S, Wu M, Zhang J, Ahvanooey MT, Aslam MS (2019) Safety risk monitoring of cyber-physical power systems based on ensemble learning algorithm. IEEE Access 7:24788–24805
22. Hou J, Li Q, Meng S, Ni Z, Chen Y, Liu Y (2019) DPRF: a differential privacy protection random Forest. IEEE Access 7:130707–130720. https://doi.org/10.1109/ACCESS.2019.2939891
23. Hou J, Li Q, Cui S et al (2020) Low-cohesion differential privacy protection for industrial internet. J Supercomput 7:1–23. https://doi.org/10.1007/s11227-019-03122-y
24. Hou J, Li Q, Tan R, Meng S, Zhang H, Zhang S (2019) An intrusion tracking watermarking scheme. IEEE Access 7:141438–141455. https://doi.org/10.1109/ACCESS.2019.2943493
25. Li Q, Song Y, Zhang J, Sheng VS (2020) Multiclass imbalanced learning with one-versus-one decomposition and spectral clustering. Expert Syst Appl 147:113152. https://doi.org/10.1016/j.eswa.2019.113152
26. Li Q, Hou J, Meng S, Long H (2020) GLIDE: a game theory and data-driven mimicking linkage intrusion detection for edge computing networks. Complexity 2020:713616018 pages, 2020. https://doi.org/10.1155/2020/7136160
27. Hou J, Li Q, Chen Y, Meng S, Long H, Sun Z (2019) Intelligent system security event description method. In: 9th EAI international conference on cloud computing. Springer, Sydney Dec 4 - Dec 5, 2019
28. Brændeland G, Dahl HEI, Engan I et al (2007) Using dependent CORAS diagrams to analyse mutual dependency. In: Critical information infrastructures security. Springer, Berlin Heidelberg, pp 135–148

Publisher's Note