Journal of Communications and Information Networks
Vol.1, No.4, Dec. 2016
DOI: 10.11959/j.issn.2096-1081.2016.040

Correspondence letter

Game strategies for distributed denial of service defense in the Cloud of Things

WANG Yichuan, ZHANG Yefei, HEI Xinhong, JI Wenjiang, MA Weigang

Faculty of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China

Abstract: Integration of the IoT (Internet of Things) with Cloud Computing, termed as the CoT (Cloud of Things) can help achieve the goals of the envisioned IoT and future Internet. In a typical CoT infrastructure, the data collected from wireless sensor networks and IoTs is transmitted through a SG (Smart Gateway) to the cloud. The bandwidth between an IoT access point and SG becomes a bottleneck for information transmission between the IoT and the cloud. We propose a novel game theory model to describe the CoT attacker, who expects to use minimum set and energy consumption of IoT attack devices to occupy as many bandwidth resources as possible in a given time period; and the defender, who expects to minimize false alarms. By analyzing this model, we have found that the game theory model is a non-cooperative and repeated incomplete information game, and Nash equilibrium is existent, perfected by the subgame. The best strategy for each stage of the attack is to adjust the attack link number dynamically based on the comparison results of value ϵ and turning point ϵ_0 for each time period. At the same time, the defender adjusts the threshold value β dynamically, based on the comparison results of the *Load* value and expected value of α for each time period. The simulation result shows that our strategy can significantly mitigate the harm of a distributed denial of service attack.

Key words: Cloud of Things, network security, DDoS attack, smart gateway, energy consumption

Citation: WANG Y C, ZHANG Y F, HEI X H, et al. Game strategies for distributed denial of service defense in the Cloud of Things[J]. Journal of communications and information networks, 2016, 1(4): 143-155.

1 Introduction

The IoT paradigm, which manages its own configuring nodes (things) with high intelligence, is dynamic and global net worked infrastructure oriented. It generally contains small objects (things) with limited memory storage and computing capacity, and is characterized by the real world with consequential issues regarding privacy, performance, scalability and reliability[1]. Conversely, cloud computing is vast with virtually unlimited capabilities regard to global storage and computation power. This technology has partially solved most IoT issues. The IoT and cloud are two comparatively challenging technologies and they have been merged together to change the current and future Internet working services[2,3]. Most papers proposed the cloud and IoT separately, and have shown great interest in this trend since 2008,

Manuscript received Aug. 17, 2016; accepted Nov. 22, 2016
This paper is supported by the Natural Science Founds of China (Nos. 61602376, U1334211, U1534208), Shaanxi Science and Technology Innovation Project (No. 2015KTZDGY01-04), Science Technology Project of Shaanxi Education Department (No. 16JK1573), Ph.D. Research Startup Funds of Xi'an University of Technology (No. 112-256081504), College Research Funds of Xi'an University of Technology (No.112-451016007).

Moreover, there were more publications between 2008 and 2013 regarding the proposed integration of cloud and IoT in our review. Currently, the upcoming trend is the integration of cloud and CoT. This new model is called as the CoT.

It is known that many IoT devices are vulnerable to simple intrusion attempts, for example, using weak or even default passwords[4]. In 2012, the Carna botnet revealed that there were more than 1.2 million open devices that allowed login with empty or default credentials. In January, 2014, an Internet-connected refrigerator was discovered as part of a botnet sending over 750 000 spam e-mails. In December, 2014, an online DDoS (Distributed Denial of Service) attack (i.e., booter) knocked down Sony and Microsoft Corporation's gaming networks, presumably powered by thousands of compromised IoT devices such as home routers[5].

From an attacker's point of view, IoT devices have their own advantages, as opposed to PCs. They are online 24/7, have no anti-virus installed, and have weak login passwords, giving attackers an easy access to powerful shells[5]. However, they also have their own disadvantages because they rely primarily on the battery as a power source. If the total energy consumed by the infected IoT devices is too much, their lifetime is sharply reduced[6]. Hence, the attacker has to consume more time and incur a greater cost to infect other IoT devices. Thus, the goal of a smart attacker is to control multiple infected IoT devices to launch DDoS attacks and slow, or takedown, the ability of the targeted domain, network infrastructure, web site, or application, to accept legitimate requests. The bandwidth between an IoT AP (Access Point) and a smart gateway SG becomes a bottleneck for information transmission between the IoT and the cloud.

The CoT attacker expects to use both a minimum number of IoT attack devices and minimal energy consumption to occupy the most band-width resources in a given time period, whereas the defender expects to minimize the amount of false alarms. In this paper, we propose a novel game theory model to describe the scenario. In our model, we consider that: 1) both the attacker and defender are rational, and 2) their strategies are dynamic.

The remainder of this paper is organized as follows: Section 2 introduces related studies and gives an overview of our research. Section 3 explains our game model and Section 4 analyzes the model. The simulation experiments using NS-3 are discussed in Section 5, and conclusions are provided in Section 6.

2 Related work

In this section, we discuss the basics of IoT, Cloud, CoT, and DDoS attacks, and overview their essential characteristics.

2.1 Cloud of Things

The core idea of the IoT can be summarized in a sentence: "A worldwide network of interconnected entities"[7-10]. With the popularity of the wireless communication system, IoT has been increasingly employed as a technology driver for crucial smart monitoring and control applications[11-13]. An IoT system can be depicted as a collection of smart devices that interact with each other to achieve a common goal[14]. IoT works on the basis of M2M (Machine-To-Machine) communication, which refers to the communication between two machines without human intervention. In a centralized approach, application platforms located in the Internet (e.g. cloud services) acquire information from entities located in the data acquisition network, and provide raw data and services to other entities.

The sensor is a typical intelligent device in IoT. Most sensors utilize limited battery energy to provide power. Owing to the conditional restriction in many

cases such as the difficult geographical environment they are located, the batteries are difficult to replace by maintenance personnel, who hope the batteries can survive for months or even years in the network. The battery subsystem capacity determines the systems operational life span. As Ref.[15] reported, sensors operating at high frequencies consumed more power than those operating at low (base-band) frequencies. Therefore, energy is sharply reduced when the sensor frequently sends a large number of data. Thus, the IoT system must consider the aspect of energy management.

In recent years, cloud computing has brought great convenience and improved resources sharing over the Internet[16]. It is a model for enabling ubiquitous, convenient, and on-demand network access to a shared pool with configurable computing resources (e.g., networks, servers, storage, applications, and services). Thereby, cloud computing can provide significant convenience to its customers, and performance improvement via resource sharing[17].

The cloud can benefit from IoT, extending its limits to real world things in a more dynamic and distributed manner, and deliver a massive number of services in real time[18-20]. The cloud will act as an intermediate layer between the application and the things, concealing all the functionalities and complexities required for later processing[18]. Fig.1 presents an overall communication pattern of CoT, which helps manage IoT resources and provides more effective cost and efficient means to produce services. CoT creates a new and extended portfolio of services.

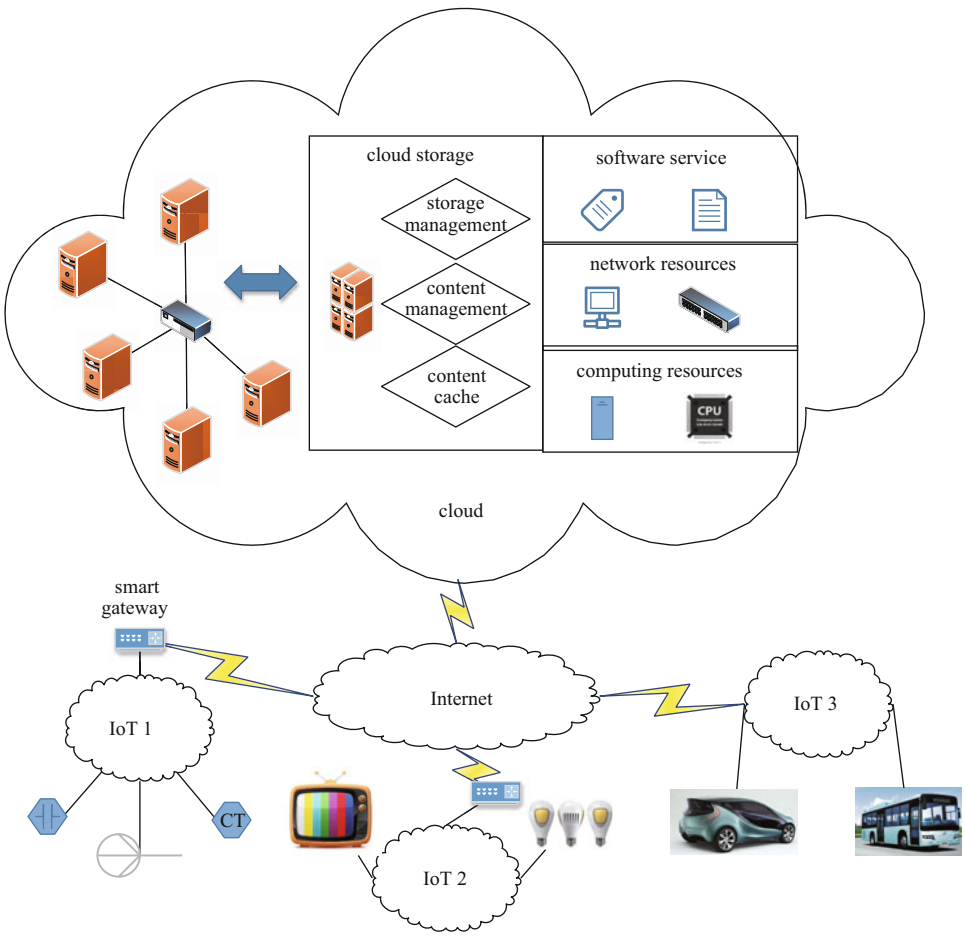

Figure 1 Cloud of Things

With CoT, the services are provided in the cloud, and ubiquitous access is given to users, extending the scope of service usage, as well as improving accessibility. CoT affects future application development, where the information gathering process and transmission will deliver new challenges to be addressed in a multi-cloud environment[21].

An SG would provide better help in the utilization of network and cloud resources. The data collected from wireless sensor networks would be transmitted through a gateway to the cloud. Received data are then stored in the cloud and provided as a service to users through the cloud. SG must manage various aspects of underlying IoTs and perform a number of tasks, such as data collection, preprocessing, filtering data and reconstructing it into a more useful form, uploading only necessary data to the cloud, tracking IoT objects and sensor activities, power energy consumption, security and privacy of the data, and overall service monitoring and management. It is possible that the data gathered from the IoT is transmitted directly to the SG, or that multiple IoTs are connected with base station(s), which in turn transmit data to the SG.

2.2 Botnet and DDoS attack defense

Botnet studies typically focus on four aspects, including detecting, analyzing, resisting and counterattack. Botnet detection and analysis receive more additional attention.

A light-weight mechanism was proposed to detect botnets by using their fundamental characteristics in Ref.[22]. It referred to a BotGAD, which requires a small amount of data from DNS (Domain Name System) traffic to detect a botnet. The BotGAD can automatically detect botnets while providing real-time monitoring in large-scale networks. Ref.[23] used fuzzy pattern recognition techniques based on frequency to observe bot behavior. Meanwhile, several researchers have focused on new botnet technologies for better botnets analysis and development trend prediction. Ref.[24] analyzed a new form of P2P (Peer-to-peer Computing) botnets called AntBot, which aimed to spread C&C (Command and Control) information to individual bots even though an adversary persistently polluted keys used by seized bots to search the C&C information.

For current DDoS attack and defense studies, as Ref.[25] shows, not only is there an alarming increase in the number of DDoS attack incidents, but also the attack technique, botnet size, and attack traffic, have attained new heights. Effective defense measures to mitigate attacks are imminent.

Ref.[26] demonstrated the exploitation pattern of an inherent weakness in LHAC (Local-Host Alert Correlation) based methods and asserted that current LHAC implementations could allow pockets of cooperative bots to hide in an enterprise-level network scale. Ref.[27] proposed a graph-based representation of infected computers, allowing us to use graph-partitioning algorithms to separate out different botnets, even in a network infected with varieties of zombie viruses at the same time. Ref.[28] proposed a method of detecting DDoS attacks through data mining.

In the new CoT network environment, two problems should be noticed: 1) A CoT botnet can dynamically adjust attack strategy to launch a larger scale DDoS attack using infected smart devices. 2) A dynamic defense mechanism should be deployed in the SG against the IoT DDoS attack.

3 Game model

A DDoS attack towards the SG is regarded as effective, if the adversary can consume the network resources between AP and SG sharply and massively. We present our game models for CoT DDoS attacks and their possible countermeasures. We consider the interaction between DDoS attacking device master

(AM) and DM (Defense Mechanism) in a SG as a two-player game. It is a non-cooperative incomplete information game.

The DM knows the network connection numbers and loads. It can determine whether to stay connected, or disconnect the link, depending on the suspicious value for each connection α using IDS(Intrusion Detection Systems). Such strategy is adjusted according to a threshold value represented by β and Neyman-Pearson criterion for hypothesis testing.

When $\alpha>\beta$, the DM considers the device in IoT as an attacker and disconnects the link to free bandwidth resources. The DM also tries to avoid bandwidth overload, which will cause the links of valid things to be disconnected. Thus, the attack recognition rate γ must be improved. The degree is only a theoretical value that DM is able to improve by the evaluation of the known attacks. Thus the DM attempts to discover an optimal strategy for the threshold value β of the disconnection request to improve the detection rate and reduce the false alarm rate of the current knowledge base.

The AM knows all the statuses of the attack nodes. This includes the total number of attack nodes and links, network infrastructure and whether the attack link has been disconnected. The information is gathered via various methods (e.g., network detection), and includes the bandwidth resource consumption of normal and attack nodes, as well as the current load. The AM is able to prepare an attack strategy, such as the number attack nodes (represented by function N_A) needed, amount of electrical energy (represented by function N_ρ) consumed, and which attack mode should be applied in each attack. The AM must avoid the DM to detect all attack nodes, represented by N_B, because for each attraction, N_B is a constant. The strategy must ensure an efficient DDOS attack, and, at the same time, hide attack nodes in proportion to the AM to the highest degree.

3.1 The DM Strategy

The DM is unable to determine whether network traffic is from an attack node. In actual networks, the DM responds or rejects a request according to the network access control rules. Therefore, the DM measures the network connection through a defined suspicious value.

Definition 1 (Suspicious) The suspicious of a network connection is defined by the malicious degree of the connection determined by the DM's supervisor.

The DM handles each connection by consulting $\alpha(X)$ and its rule. Hereby, we define γ to denote the accuracy of the judgment towards current attacks. Thus, γ is an objective theoretical value. Hereby, we define the conception malicious intent (Υ) to describe whether a connection is malicious.

Definition 2 (Malicious intent) Malicious intent Υ of a network connection, defined by the malicious possibility of the connection. It is a function of γ and α. We denote it as $\Upsilon(\gamma, \alpha)$.

When $\gamma = 0$, the probability of whether a connection has malicious intent can be denoted by 1/2, which is independent of the subjective suspicious of the DM. When $\gamma = 1$, the probability of whether a connection has malicious intent can be denoted by α. How the DM improves its recognition rate through self-learning is beyond the scope of this paper. Here, we simply assume $\Upsilon(\gamma, \alpha)$ is a linear function with respect to γ. Then we get the expression:

$$\Upsilon(\gamma,\alpha)=\left(\alpha-\frac{1}{2}\right)\gamma+\frac{1}{2}, \qquad (1)$$

Therefore, whether a connection is malicious can be denoted by probability function $1-\Upsilon(\gamma, \alpha)$.

There are four cases that will happen when IDS judges the link property and processes the link connection status. 1) A link to be preserved, while the link is an attack connection. 2) A link to be

disconnected, and the link is an attack connection. 3) A link to be preserved, and the link is a normal connection. 4) A link to be disconnected, while the link is a normal connection. For each connection, we get the strategies of the DM distribution as shown in Tab.1.

Table 1 Strategies distribution of the DM

property\strategies	reserve	disconnect	probability
attack	P_{AR}	P_{AD}	$\gamma(\gamma, \alpha)$
normal	P_{NR}	P_{ND}	$1-\gamma(\gamma, \alpha)$
probability	$P\{\alpha \leqslant \beta\}$	$P\{\alpha>\beta\}$	

Since variables γ and β are independent of each other, we assume that the joint distribution is independent. Thus, we get

$$\begin{cases} P_{AR} = P\{\alpha \leqslant \beta\}\left(\left(\alpha-\frac{1}{2}\right)\gamma+\frac{1}{2}\right), \\ P_{AD} = P\{\alpha > \beta\}\left(\left(\alpha-\frac{1}{2}\right)\gamma+\frac{1}{2}\right), \\ P_{NR} = P\{\alpha \leqslant \beta\}\left(\frac{1}{2}-\left(\alpha-\frac{1}{2}\right)\gamma\right), \\ P_{ND} = P\{\alpha > \beta\}\left(\frac{1}{2}-\left(\alpha-\frac{1}{2}\right)\gamma\right). \end{cases}$$

We get the strategies expectation of DM for connection i.

$$E_D^i = P_{AR}^i W_{AR} + P_{AD}^i W_{AD} + P_{NR}^i W_{NR} + P_{ND}^i W_{ND} . \quad (2)$$

W_{AR}, W_{AD}, W_{NR} and W_{ND} respectively represent the weight of the above four different cases, Case 1 and 4 represent the incorrect judgment of IDS, while case 2 and 3 represent the correct judgment of IDS.

Attacking nodes can imitate the normal devices (e.g., sensors) to access the SG. $\overline{R_C}$ represents the total bandwidth resources of the SG. R_C represents the resource occupation of one connection. Thus, $N_R = \frac{R}{R_C}$ depicts the capacity of all connections of the server. Moreover, the current total number of connections is $N_C (N_C \leqslant N_R)$.

Thus, we get the utility function of the DM.

$$U_D = \frac{N_C}{N_R} W_O + \left(1-\frac{N_C}{N_R}\right) W_D \sum_{i=1}^{N_C} E_D^i, (N_C \leqslant N_R) . \quad (3)$$

W_O denotes the costs of the crashed network between AP and SG caused by the DM strategy failure. W_D denotes the weight of the network not crashed yet with N_C connections maintained in the meantime.

3.2 The AM strategy

Similarly, the AM is unsure whether the DM has been equipped with a sandbox or honeypot for series detection and measurements towards the AM. The probability that the current attack node is rejected is the basis for the AM's next strategy decision. Assume N_A is the current total active attack nodes keeping a connection with the smart gateway. N_T represents the total active attack nodes in time T. It is a non-decreasing function related to time t and has a minimum value 0, and maximum value N_B to denote the total attack nodes. While, the function itself depends on the strategy that AM decided. $\frac{N_A}{N_T}$ depicts the proportion of the DM forwarding attack traffic from attack nodes. We use $1-\frac{N_A}{N_T}$ to denote the denying probability of next the attack.

The AM can apply strategies to launch attacks or keep the hidden state for each infected node. Obviously, the connections of the attack nodes, which have already been launched, shall be considered as exposure regardless of whether the AM continues to take the attack strategy. Meanwhile, we assume that the AM decides whether to start attack traffic based on the comparison between the current time t and T, denoted as the actual initial attack time of the current launching attack nodes. That is to say, if $t > T$, then the attack link starts. If $t \leqslant T$ then the link hides and does not start attacks.

The same applies for the DM strategy. There are four cases that will happen for each infected node connection as follows: 1) An infected node launches an attack, while the defender deploys defense mechanisms, such as IDS, and honeynet. 2) An infected node hides, and the defender deploys defense mechanisms. 3) An infected node launches attack, and the defender does not deploy defense mechanisms. 4) An infected node hides, and the defender does not deploy defense mechanisms. We derive the strategies distribution for the AM, which is shown in Tab.2.

Table 2 Strategies distribution of the AM

property\strategies	attack	hide	probability
detected	P_{DA}	P_{DH}	$1-\frac{N_A}{N_T}$
not detected	P_{NA}	P_{NH}	$\frac{N_A}{N_T}$
probability	$P\{t>T\}$	$P\{t \leqslant T\}$	

Because variables N_A and T are independent of each other, we assume the joint distribution is independent. Thus, we get

$$\begin{cases} P_{DA} = P\{t > T\}\left(1-\dfrac{N_A}{N_T}\right), \\ P_{DH} = P\{t \leqslant T\}\left(1-\dfrac{N_A}{N_T}\right), \\ P_{NA} = P\{t > T\}\left(\dfrac{N_A}{N_T}\right), \\ P_{NH} = P\{t \leqslant T\}\left(\dfrac{N_A}{N_T}\right). \end{cases}$$

We get the strategies expectation of AM for connection i.

$$E_A^i = P_{DA}^i W_{DA} + P_{DH}^i W_{DH} + P_{NA}^i W_{NA} + P_{NH}^i W_{NH} . \quad (4)$$

W_{DA}, W_{DH}, W_{NA} and W_{NH} respectively represent the weight of the above four different cases, Case 1 and 4 represent the AM does not take an effective attack strategy, while case 2 and 3 represent that AM does take an effective attack strategy.

The sensors operating at high frequencies burn more power than those operating at low (baseband) frequencies. As Ref.[6] reported, the author analyzes the relationship between the transmitted data volume and the energy consumption in different transfer protocols and finds a certain proportional relationship between them as presented in Fig.2. It shows that the energy consumption almost closes to a linear function with the number of sent data.

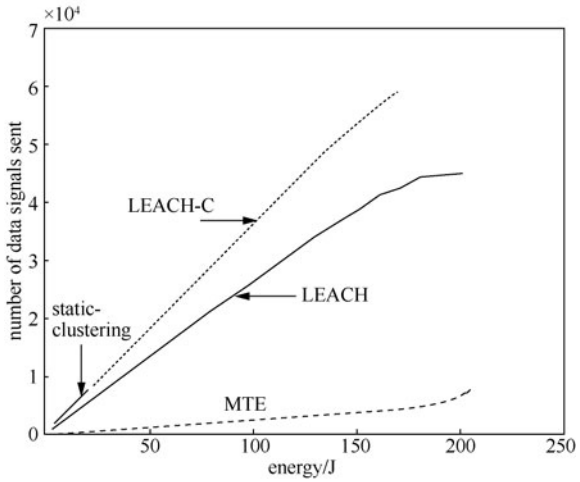

Figure 2 Total amount of data sent at the BS per given amount of energy

Definition 3 (Energy consumption) Ξ_i denotes the energy consumption of an attack link, from the ith infected device in IoT.

$$\Xi_i = l_i \cdot \rho_i , \quad (5)$$

where l_i denotes the number of bits transmitted through the ith attack link in a short time slot (ΔT), ρ_i denotes the energy consumption for transmitting one bit.

It is shown that in Ref.[6], $\rho_i \approx 50$ nj average, which can be denoted as ρ, for a wireless IoT node. For convenience, we assume that communication devices are similar in terms of energy consumption in the IoT, and nearly equal data transmission l for each attack node. Thus, we get the total energy consumption of all the infected nodes that launched attack, and denote

it as Ξ :

$$\Xi = N_A l\rho \ . \tag{6}$$

The utility function of AM is

$$U_A = \frac{N_T}{N_B} W_E + \left(1 - \frac{N_T}{N_B}\right) W_A \sum_{i=1}^{N_A} E_A^i + \frac{N_C}{N_R} W_S + \Xi W_\rho, (N_T \leqslant N_B), \tag{7}$$

$$U_A = \frac{N_T}{N_B} W_E + \left(1 - \frac{N_T}{N_B}\right) W_A \sum_{i=1}^{N_A} E_A^i + \frac{N_C}{N_R} W_S + N_A l \rho W_\rho, (N_T \leqslant N_B). \tag{8}$$

W_E denotes the weight of costs if AM exposes all attack nodes. W_A denotes the weight if AM does not expose all attack nodes. W_S denotes the income if AM finishes the DDoS attack to crash the SG successfully. $W\rho$ represents the weight of the importance of the energy consumption.

4 Model analysis

We use MATLAB as the platform for numerical computation. Just like the example in the DM, let us consider the scenario $W_{AR} = W_{ND} = -1$ and $W_{AD} = W_{NR} = 1$. Fig.3 illustrates the pay off of DM U_D for each different distribution of α.

Experiment 1 parameters are: N_R=1 000, γ= 0.5, $W_O = -100$; $U_D(Load, \beta)$ when $W_D = 1$. We consider that the value $E(\alpha)$ is in compliance with a Poisson distribution. Thus we denote the cumulative distribution function (CDF) as $e^{-\lambda}\sum_{i=0}^{\lfloor k \rfloor} \frac{\lambda^i}{i!}$, where λ is the expected value and k is the number of occurrences. Therefore, we first select α_1'~*Poisson* (40) and α_2'~*Poisson* (70) to approximate a practical situation. Then we distribute $\frac{\alpha_1'}{100}$ and $\frac{\alpha_2'}{100}$ as the distribution of α_1 and α_2.

Here, $Load = \frac{N_C}{N_R}$, and $E(\alpha)$ are the mathe-matical expectation of α. Experiment 1 shows the best strategy for DM by which the value β is increased if $Load > E(\alpha)$, and decreased if $Load > E(\alpha)$. This strategy is able to obtain higher benefits for the DM according to the utility function.

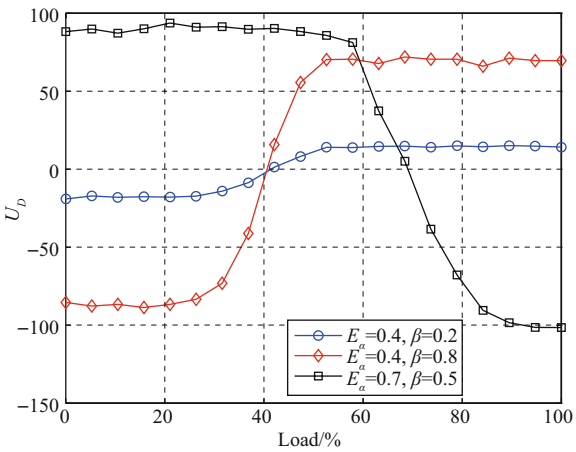

Figure 3 Effect of suspicious value to the DM

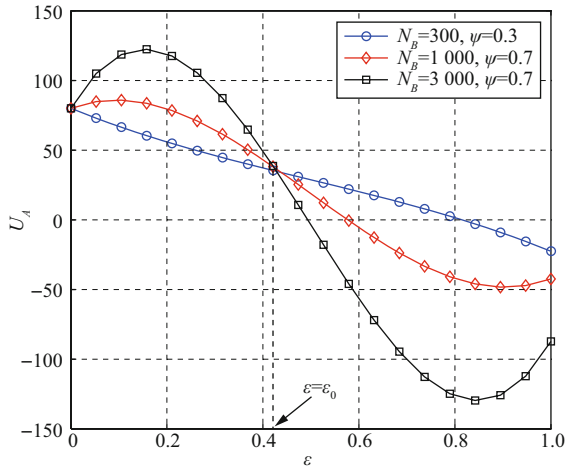

Figure 4 Effect of suspicious value to the AM

For the AM, we consider the scenario $W_{DA} = W_{NH} = -1$ and $W_{DH} = W_{NA} = 1$. Fig.4 illustrates the payoff of AM U_A for each different distribution of N_B. Experiment 2 parameters: N_R = 10 000; W_E = −100; W_A =1; W_S = 100; W_ρ=−1; l = 800×1 024×8; ρ =0.000 000 005; $Load$ = 0.8, N_B has the value 300, 1 000, and 3 000, respectively. Where $\psi = \frac{N_T}{N_B}, \epsilon = \frac{N_A}{N_T}$.

Fig.4 shows the U_A curve graph when variables N_B, ψ are changed in three conditions, while other variables are fixed. we can find the three curves are focused on one point, denoted as Turning Point, represented by ϵ_0 . By analyzing the curve change process, we can determine the best strategy for the AM is hiding attack nodes to avoid detection and counterattack, if the number of effective attack connections is over ϵ_0, that is $\epsilon \geqslant \epsilon_0$. Conversely, if the AM has less than ϵ_0 of effective attack connections, that is, $\epsilon < \epsilon_0$, the best strategy is to increase the number of current launching attack nodes. It can be concluded that value ϵ_0 has a deep influence on the AM strategies.

Table 3 Turning point

W_ρ	ϵ_0	W_ρ	ϵ_0	W_ρ	ϵ_0	W_ρ	ϵ_0
−0	0.5	−3	0.32	−6	0.19	−9	0.05
−1	0.42	−4	0.26	−7	0.16	−10	0.02
−2	0.37	−5	0.21	−8	0.11	−11	0.008

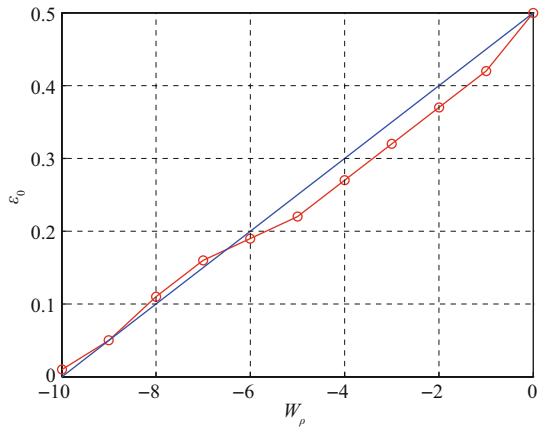

Figure 5 Turnning points of AM

Tab.3 represents value ϵ_0 as it corresponds to different W_ρ values. The relationship diagram is shown in Fig.5.

The dashed curve shows the actual relationship diagram. To simplify, we use a linear function (shown as a solid line) to describe it, and then, find that they are similar. ϵ_0 is shown as.

$$\epsilon_0 = = \frac{1}{20} W_\rho + \frac{1}{2} . \tag{9}$$

From the above analysis, the strategies of the DM are "increase β" and "decrease β", while the strategies of AM are "increase new traffic" and "decrease new traffic". In connection with a DDoS attack, the two-party-game, between the AM and DM, is a non-cooperative and repeated with incomplete information game.

Theorem 1 If the DM and the AM are rational, there exists a unique Nash equilibrium point in stage strategies.

Proof 1 (Proof of theorem 1) :

Before the given time T, the DM is uncertain, not only of the ratio ε of disconnected attack nodes traffic to all attack nodes, but whether the AM intends to increase or decrease the number of current launching attack nodes. Hence, based on the known *Load* and $E(\alpha)$, if $E(\alpha) > Load$, the DM chooses to "increase β", otherwise, to "decrease β".

Similarly, before the given time T, the AM is uncertain not only of the suspicious value α of the DM for each connection, but of $E(\alpha)$. Hence, based on the known ϵ , if $\epsilon < \epsilon_0$, the AM chooses to "increase new flows", otherwise, to "decrease new flows".

At the given time T, both the DM and AM need to decide their strategies according to the $E(\alpha)$ and ϵ previously determined. The game reaches the Nash equilibrium. We represent the Nash equilibrium point as $s^*(X^*, Y^*)$ fulfilling $U_A(\epsilon, X^*) \geqslant U_A(\epsilon, X)$. The X represents whether to increase or decrease the new random attack connection numbers, $U_D(E(\alpha), Y^*) \geqslant U_D(E(\alpha), Y)$. The Y represents whether to increase or decrease the value of β. The game parties can then determine their strategies for each connection and traffic with reference to this result. The DM may decide to reserve or disconnect each connection based on the new β value for obtaining the

vector y^* of the strategies matrix. In a similar manner, the AM may adjust new connection operations via controlled attack nodes based on the strategy X^* for obtaining strategy x^* vector.

$$U_A(\epsilon, x_i^*) \geqslant U_A(\epsilon, x_i), 1 \leqslant i \leqslant N_C, \forall x_i ,$$
$$U_D(E(\alpha), y_j^*) \geqslant U_D(E(\alpha), y_i), 1 \leqslant i \leqslant N_B, \forall y_j .$$

In each stage of the game, if the DM and AM are rational, they both choose the strategy using their respective utility function to reach maximization. $(E(\alpha) > load) \cap (E(\alpha) \leqslant load) = \phi$ and $(\epsilon < \epsilon_0) \cap (\epsilon \geqslant \epsilon_0) = \phi$. Thus, the strategies' choices are clear in each stage of the game, and Nash Equilibrium is the strictly dominant strategy for the DM and AM. Therefore, the Nash Equilibrium point is unique.

5 Simulation

We use an NS-3 network simulation tool as the platform to validate our model. The network topology is shown in Fig.6.

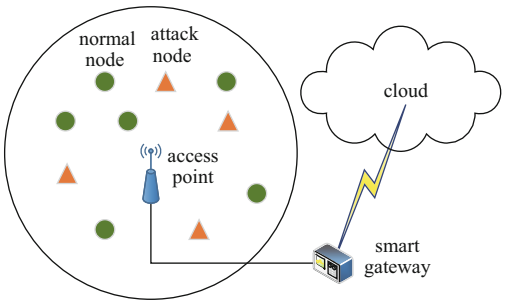

Figure 6 Simulation network topology

There are some normal IoT nodes and attack IoT nodes in the AP radio coverage. The data collected from wireless sensor networks and IoTs' flows will be transmitted through SG to the cloud. The simulation configuration of the NS-3 platform is shown in Tab.4.

As described in the previous model description, we assume that both attacker and defender are rational. If $\epsilon \geqslant \epsilon_0$ the AM hides three more attack nodes, to avoid detection. Otherwise, it adds three additional attack nodes for the current launching attack. The DM set $\beta = \beta + 0.1$, if $Load > E(\alpha)$, and $\beta = \beta - 0.1$, if $Load > E(\alpha)$. The suspicious value for each connection α depends on its arrival time. According to a threshold value, represented by β and Neyman-Pearson criterion for hypothesis testing, if an IoT node $\alpha > \beta$, the SG controlling the AP should disconnect the link between it and the AP.

Table 4 NS-3 simulation configuration

name	configuration
NS-3 version	V 3.25
server CPU	INTEL XEON X5650 12 M cache, 2.66 GHz, 6.40 GT/s
server OS	fedora 21 Linux System
bandwidth	10 Mbit/s IoT node ⟷ AP 100 Mbit/s AP ⟷ SG
delay	2 ms IoT Nodes ⟷ AP 1msAP ⟷ SG
WiFi channel model	YANS[29]
traffic type	TCP socket
port	8080
interarrival time(ms)	normal~uniform[15,45] attack~uniform[0,40]
IoT nodes number	normal: 30 attack: 20
access start time	normal: 0th second attack: 1th second
packet size	800 KB
energy consumption per Bit	50 nJ
simulation end time	5.5 th second

Fig.7 shows the comparison among the 3 cases: 1) Neither the AM nor DM adopts our strategies. 2) Both the AM and DM adopt our strategies. 3) Only the DM adopts our strategies. The X-axis is the time of simulation, and Y-axis is the bandwidth occupancy rate of the line between AP and SG.

We have noted that: In case 1), neither the AM nor DM adopts our strategies. When the DDoS attack

started, all 20 attack nodes sent socket message requests to SG through AP. We can see the bandwidth occupancy rate increasing sharply to above 90% keeping at a high level.

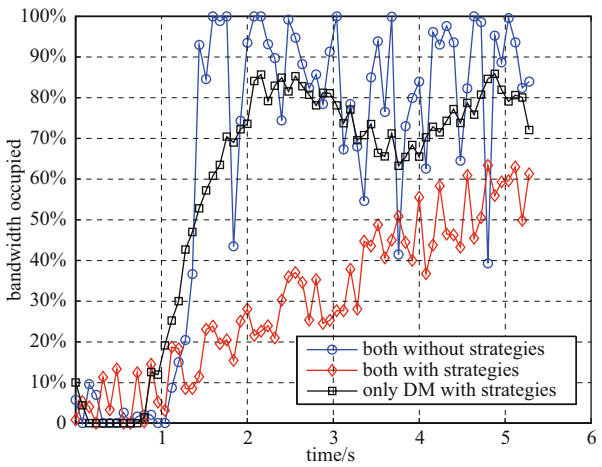

Figure 7 Effect comparison of our strategies

In case 2), both the AM and DM adopt our strategies. We can see the bandwidth occupancy rate increasing slower than the other two cases.

This is because the AM dynamically adjusts the number of attacking nodes for fewer exposure, and the DM dynamically adjusts threshold value β for better network utilization according to our strategies.

In case 3), only DM adopts our strategies, and AM does not adopt our strategies. When the DDoS attack started, all 20 attack nodes launch an attack, and we can see the bandwidth occupancy rate reducing after increasing significantly. This is because that the DM decreases threshold value β, such that some attack requests are disconnected.

Therefore, the simulation experiment shows that our strategies do indeed significantly mitigate the harm of the DDoS attack.

6 Conclusion and future work

In this paper, we propose a novel game theory model to describe our scenario as follows: the CoT attacker expects to use a minimum number of IoT attack devices to occupy the most bandwidth resources in a given time period, and the defender expects to minimize the false alarm rate. In our model, we consider that: 1) both the attacker and defender are rational, and 2) their strategies are dynamic.

By analyzing the model, we find that the confrontation relationship between the attacker and defender can be described as a non-cooperative game model. We have proved it to be a repeated incomplete information game, with an existent Nash equilibrium is existent perfected by the subgame. The best strategy for each stage of the SG defender is to reduce the threshold value β when estimating that the mathematical expectation of the suspicious value is greater than the load rate of server resources. We use an NS-3 network to validate our model and its effectiveness. The result shows that our strategy can significantly mitigate the treat posed by a DDoS attack. Our planned future work is the analysis of new DDoS attacks in the CoT network, and how to recognize and trace them.

References

[1] BABU S M, LAKSHMI A J, RAO B T. A study on cloud based Internet of Things: CloudIoT[C]// Global Conference on Communication Technologies (GCCT), Thuckalay, India, 2015, 2015: 60-65.

[2] CHANG K D, CHEN C Y, CHEN J L, et al. Internet of Things and cloud computing for future internet[M]. Security-enriched urban computing and smart grid, 2011:1-10.

[3] ZHOU J, LEPPANEN T, HARJULA E, et al. CloudThings: A common architecture for integrating the internet of things with cloud computing[C]//IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Whistler, Canada, 2013: 651-657.

[4] CUI A, STOLFO S J. A quantitative analysis of the insecurity of embedded network devices: results of a wide-area scan[C]//The 26th Annual Computer Security Applications Conference, Austin, USA, 2010: 97-106.

[5] PA Y M P, SUZUKI S, YOSHIOKA K, et al. IoTPOT: analysing the rise of IoT compromises[C]//The 9th USENIX Conference on Offensive Technologies, Washington, USA, 2015: 9.

[6] HEINZELMAN W, CHANDRAKASAN A P, BALAKRISHNAN H. An application specific protocol architecture for wireless microsensor networks[J]. IEEE transactions on wireless communications, 2002, 1(4): 660-670.

[7] BASSI A, HORN G. Internet of Things in 2020[C]//Joint European Commission/EPoSS Expert Workshop on RFID/Internet-of-Things, Brussels, Belgium, 2008.

[8] VILMOS A, MEDAGLIA C, MORONI A, et al. Vision and challenges for realising the Internet of Things[J]. Hot working technology, 2010, 35(2): 59-60.

[9] VERMESAN O, FRIESS P, GUILLEMIN P, et al. Internet of things strategic research roadmap[J]. Information security & technology, 2009, 29(16): 300-304.

[10] ROMAN R, ZHOU J, LOPEZ J. On the features and challenges of security and privacy in distributed internet of things[J]. Computer networks, 2013, 7(10): 2266-2279.

[11] ATZORI L, IERA A, MORABITO G. The internet of things: a survey[J]. Computer networks, 2010, 54(15): 2787-2805.

[12] MIORANDI D, SICARI S, de Pellegrini F, et al. Internet of Things: vision, applications and research challenges[J]. Ad hoc networks, 2012, 10(7): 1497-1516.

[13] PALATTELLA M R, ACCETTURA N, VILAJOSANA X, et al. Standardized protocol stack for the internet of (important) things[J]. IEEE communications surveys & tutorials, 2013, 15(3): 1389-1406.

[14] SICARI S, RIZZARDI A, GRIECO L A, et al. Security, privacy and trust in Internet of Things: the road ahead[J]. Computer networks, 2015, 76: 146-164.

[15] POTTIE G J. Wireless sensor networks[C]//Information Theory Workshop, Killarney, Ireland, 1998.

[16] WANG Y, MA J, LU D, et al. From high-availability to collapse: quantitative analysis of "cloud-droplets-freezing" attack threats to virtual machine migration in cloud computing[J]. Cluster computing, 2014, 17(4): 1369-1381.

[17] WANG Y, CHANDRASEKHAR S, SINGHAL M, et al. A limited-trust capacity model for mitigating threats of internal malicious services in cloud computing[J]. Cluster computing, 2016, 19(2): 647-662.

[18] BOTTA A, DE DONATO W, PERSICO V, et al. On the integration of cloud computing and internet of things[C]// International Conference on Future Internet of Things and Cloud (FiCloud), Barcelona, Spain, 2014: 23-30.

[19] AAZAM M, KHAN I, ALSAFFAR A A, et al. Cloud of Things: Integrating Internet of Things and cloud computing and the issues involved[C]//The 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 2014, 2014:414-419.

[20] AAZAM M, HUNG P P, HUH E N. Smart gateway based communication for cloud of things[C]//IEEE 9th International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Singapore, 2014, 2014: 1-6.

[21] AGUZZI S, BRADSHAW D, CANNING M, et al. Definition of a research and innovation policy leveraging cloud computing and IoT Combination[EB/OL]. European commission, directorate-general of communications networks, content & technology, 2014, https://ec.europa.eu/digital-single-market/news/call-tenders-study-definition-research-and-innovation-policy-leveraging-cloud-computing-and-iot.

[22] CHOI H, LEE H. Identifying botnets by capturing group activities in DNS Traffic[J]. Computer networks, 2012, 56(1): 20-33.

[23] WANG K, HUANG C Y, LIN S J, et al. A fuzzy pattern-based filtering algorithm for botnet detection.[J]. Computer networks, 2011, 55(15): 3275-3286.

[24] YAN G, HA D T, EIDENBENZ S. AntBot: Anti-pollution peer-to-peer botnets[J]. Computer networks, 2011, 55(8):1941-1956.

[25] ARORA K, KUMAR K, SACHDEVA M. Impact Analysis of Recent DDoS Attacks[J]. International journal on computer science & engineering, 2011, 3(2): 4-5.

[26] SHIRLEY B, BABU L, MANO C. Bot detection evasion: a case study on localhost alert correlation bot detection methods[J]. Security and communication networks, 2012, 5(12): 1277-1295.

[27] JAIKUMAR P, KAK A C. A graphtheoretic framework for isolating botnets in a network[J]. Security and communication networks, 2012, 5(16):2605-2623.

[28] GARG K, CHAWLA R. Detection of DDoS attacks using data mining[J]. International journal of computing and business research, 2011, 2(1): 2229-6166.

[29] LACAGE M, HENDERSON T R. Yet another network simulator[R]. Institut National De Recherche En Informatique Et En Automatique Research Report: RR-5927, INRIA. 2006.

About the authors

WANG Yichuan [corresponding author] was born in Chengdu, China. He received his Ph.D. in computer system architecture from Xidian University of China in 2014. Now he is a Lecturer at Xi'an University of Technology, and with Shaanxi Key Laboratory of Network Computing and Security Technology. His research areas include cloud computing, trusted computing, and network security. (Email: chuan@xaut.edu.cn)

ZHANG Yefei was born in Datong, China. In 2013, she received her B.S. degree from Yuncheng Institute. Currently, she is a master's candidate in the Faculty of Computer Science and Engineering, Xi'an University of Technology, Xi'an, China. Her research interests include cloud computing and network security. (Email: xiaxuena1@163.com)

HEI Xinhong was born in Yanan, China. He received his B.S. and M.S. degrees in computer science and technology from Xi'an University of Technology, Xi'an, China, in1998 and 2003, respectively, and his Ph.D. degree from Nihon University, Tokyo, Japan, in 2008. He is currently a professor with the Faculty of Computer Science and Engineering, Xian University of Technology, Xi'an, China. His current research interests include intelligent systems, safety-critical systems, and train control systems. (Email: heixinhong@xaut.edu.cn)

JI Wenjiang was born in Yanan, China. He obtained his B.S. and Ph.D from Xidian University in 2006 and2013, respectively. He is currently a lecturer in Xi'an University of Technology. His research interests include information and network security in VANET, privacy preserving in VANET and network simulation. (Email: wjj@xaut.edu.cn)

MA Weigang was born in Lanzhou, China. He received his Ph.D. degrees in computer system architecture from Xidian University of China in 2015. Currently, he is a Lecturer at Xi'an University of Technology and with Shaanxi Key Laboratory of Network Computing and Security Technology. His research areas include cloud computing, trusted computing and software reliability. (Email: mwg arkey@xaut.edu.cn)