ELSEVIER

Expert Systems with Applications 27 (2004) 379–390

Expert Systems with Applications

www.elsevier.com/locate/eswa

Constructing detection knowledge for DDoS intrusion tolerance

Shun-Chieh Lin, Shian-Shyong Tseng*

Department of Computer and Information Science, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 300, Taiwan, ROC

Abstract

Intrusion tolerance is the ability of a system to continue providing (possibly degraded but) adequate services after a penetration. With the rapid development of network technology, distributed denial of service (DDoS) attacks become one of the most important issues today. In this paper, we propose a DDoS ontology to provide a common terminology for describing the DDoS models consisting of the Profile model (the representation of the behaviors of system and users) and the Defense model (the descriptions of Detection and Filter methodologies). Also, the Evaluation strategy based upon current statuses of users' behaviors is used to evaluate the degree of the intrusion tolerance of the proposed models during DDoS attacks. Based upon the ontology, four KCs (Profile model, Evaluation strategy, Detection methodology, and Filter methodology Knowledge Classes) and their relationships are then proposed, where each KC may contain a set of sub-KCs or knowledge represented as a natural rule format. For an arbitrarily given network environment, the default knowledge in the Profile KC and the Evaluation KC, the appropriate detection features in the Detection KC, and the suitable access control list policies in the Filter KC can be easily extracted and adopted by our proposed integrated knowledge acquisition framework. We are now implementing a NORM-based DDoS intrusion tolerance system for DDoS attacks to evaluate the proposed models.

Keywords: Distributed denial of service (DDoS); Intrusion tolerance; Ontology; Knowledge acquisition; NORM

1. Introduction

Intrusion tolerance is the ability of a system to continue providing (possibly degraded but) adequate services after a penetration (Stavridou, 2001). With the rapid development of network technology, the network security becomes one of the most important issues today, especially for distributed denial of service (DDoS) attacks. A DDoS attack is a simple but ferocious attack since it could be launched by someone who has DDoS attacking tools. In recent years, many e-commerce enterprises have been attacked by DDoS attacks, such as Yahoo, eBay, Amazon, Key Internet Computers and so on, which cause great damage on these enterprises (Bridis, 2002; CERT/CC, 2003; Xu & Lee, 2003). Various approaches and monitoring policies have then been developed to detect and filter the malicious traffic of DDoS attacks. Unfortunately, it is very difficult to keep up with the rapidly growing expertise or knowledge on their domains due to lack of the common terminology. Therefore, an ontology, which could be acquired from domain experts, is needed for providing a sharing vocabulary to integrate and categorize the increasing growth of knowledge of DDoS intrusion tolerance.

Due to the difficulty of handling the complicated DDoS characteristics, maintaining the huge amount of knowledge for detecting and preventing DDoS attacks becomes more and more important. Accordingly, we will propose a knowledge base to store the characteristics of DDoS attacks, which may be obtained by analyzing the traffic behaviors of the previously discovered DDoS attacking tools, to help mitigate the malicious traffic caused by the DDoS attacks. As we know, the knowledge of DDoS intrusion tolerance could be divided into several knowledge objects including various kinds of DDoS attacking tools/methods and defending heuristics. The relationships between attacking characteristics of DDoS attacking tools have not been clearly defined and the defending heuristics may be too complicated to maintain; moreover, the different characteristics and the relationships of DDoS attacks in DDoS intrusion tolerance may incrementally increase. A knowledge based system (KBS) with object-oriented model and rich relationships is required to achieve the modularity, sharability, and reusability of the dramatically increasing DDoS characteristics.

Therefore, a DDoS ontology according to the experience of the observations for DDoS intrusion tolerance will be

* Corresponding author. Tel.: +886-3-5712121x56658; fax: +886-3-5721490.

E-mail address: sstseng@cis.nctu.edu.tw (S.-S. Tseng).

0957-4174/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2004.05.016

proposed to describe the DDoS models and the Evaluation strategy to evaluate the models. Each knowledge class (KC) could be represented as a node of the ontology to achieve the modularity of DDoS knowledge containing several sub-KCs and the relationships between KCs could be represented as edges. The DDoS models consist of Profile KC (including system state and role state diagrams to model the behaviors of system and users, respectively) and Defense models (including Detection and Filter KCs to collect the knowledge for defending DDoS). To evaluate the tolerance degree during DDoS attacks, the Evaluation KC consisting of the system capacity and network bandwidth utilization criteria could be acquired from domain experts. In order to obtain the more complicated accumulating expertise and speed up the collection of expertise for DDoS intrusion tolerance, an integrated knowledge acquisition (KA) framework including interviewing, training process, and learning process will be proposed based upon the ontology.

The Profile KC could be easily acquired from domain experts by interviewing with domain experts. Traditionally, NORMAL and DEAD states are usually enough to describe the system status because the victim would be crashed simultaneously when the DDoS attacks. By applying the various filtering polices including white list and black list for DDoS attacks according to the role state diagram, the survival time of the system can then be extended and a SURVIVAL state is further added to represent such situation. Since the only way we can stop a DDoS attack once it starts is to identify the addresses of all agents (zombies) sending DDoS packets and shut off traffic from them (Garber, 2000), the system will tend to move SURVIVAL state quickly backed to NORMAL state with the heuristic of setting more restricted filtering policies in the Filter methodology KC. On the other hand, the behaviors of the DDoS intrusion tolerance system will be expected to be the NORMAL-SURVIVAL-NORMAL pattern instead of the traditional NORMAL-DEAD pattern. To determine the new useful filter policy for mitigating the damage causing by DDoS attacks, the role state diagram including CANDIDATE, TRUSTED, SUSPECTED, and UNTRUSTED states will also be proposed to represent the behaviors of each user. The Detection methodology KC, which is responsible for detecting and predicting the occurrence of the DDoS attacks, including attack predicting, attack detecting, and attack tracebacking rules could be obtained by analyzing the behaviors of DDoS attacking tools. Besides, a NORM-based intrusion tolerance system will be implemented to evaluate the proposed models.

2. Related work

As mentioned above, many different DDoS attacking tools and defending methods to help mitigate the malicious traffic developed result in the rapid growth of complicated characteristics of DDoS intrusion tolerance in recent years.

2.1. Basic of DDoS

The DDoS appeared in June of 1998 firstly means that many attackers launch malicious traffic to the same victims together and make the victims too busy to respond all the traffic including legitimate requests. Several different DDoS attacks which were developed (Barlow & Thrower, 2001; Cabreraa et al., 2001; Dittrich, 2000) from 1998 to 2002 can be divided into several categories including UDP flood, ICMP flood, TCP flood, and Smurf as shown in Table 1, so the common characteristics of each category of DDoS attacks could be easily observed and extracted. For example, the TFN2K, discovered in November 1999, is a kind of DDoS attacking tools. It can launch UDP flood, ICMP flood, Smurf, or TCP-SYN flood attacking method to attack victims in one time. All of the observed knowledge can be represented as a natural rule format and stored into a knowledge base.

The general topology of DDoS attack shown in Fig. 1 could be divided into control stage and attack stage. In control stage, a scan is performed in large ranges of network to find the list of vulnerable hosts. Generally speaking, the vulnerable hosts consist of handlers and agents, where the handlers (the first level vulnerable hosts) are controlled

Table 1
The DDoS attacks developed from 1998 to 2002

Date	Attack	Attack method						
		UDP flood	ICMP flood	Smurf	TCP SYN	TCP ACK	TCP RST	TCP SYN/ACK
1998/5	FAPI	○	○					○
1999/6–1999/7	Trin00	○						
1999/8–1999/9	TFN	○	○	○	○			○
1999/9–1999/10	Stacheldracht	○	○	○	○			
1999/11	TFN 2K	○	○	○	○			
2000/1–2000/2	Shaft	○	○		○			
2000/2	Trinity	○			○	○	○	
2002/1	DRDoS							○
2002/4	Mstream					○		

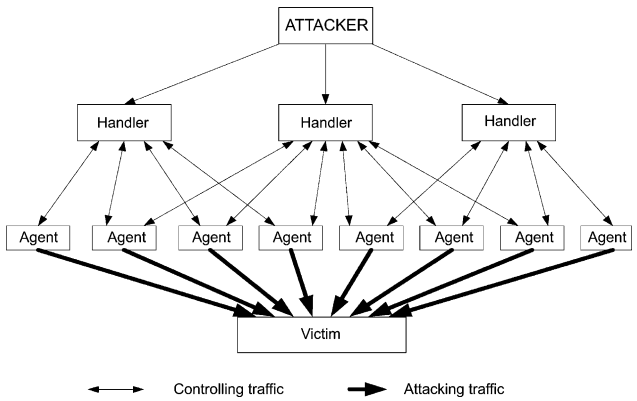

Fig. 1. The general topology of DDoS attacks.

by attackers and agents (the second level vulnerable hosts) are also controlled by attackers through handlers. Most of the controlling traffic, the traffic of communication in control stage, is signal direction between attacker and handler but is bi-direction between handlers and agents. The two level topology results in the locations of attackers can be hidden. After the control stage, the list of vulnerable hosts is then used to launch the distributed attacking traffic in attack stage. The attacking traffic including UDP flood, ICMP flood, Smurf, TCP SYN, TCP ACK, TCP RST, and TCP SYN/ACK as shown in Table 1 could overwhelm the victim.

As we know, there are two different types of attack technique in DDoS attacks: bandwidth consumption and resource consumption. The bandwidth consumption means that the attacking traffic launched by the compromised hosts, which are controlled by attackers, is aggregated to a single huge flood and overwhelms the victim. The resource consumption means that attackers make use of the leak of the network protocol or the system security such as the techniques of SYN flood, land and Teardrop, resulting in the starvation of system resources (CERT/CC, 2003).

As the DDoS attack tools have become more complicated in recent years, the maintenance of the characteristics of DDoS attacks is becoming more difficult despite the previously known common characteristics of each category of discovered DDoS attacks. Therefore, we will propose a knowledge base to store the characteristics of DDoS attacks, which may be obtained by analyzing the traffic behaviors of the DDoS attacking tools, for DDoS intrusion tolerance. Besides, two criteria considering the difference between two types of DDoS attacks will be proposed to evaluate the degree of intrusion tolerance.

2.2. Intrusion tolerance of DDoS

Intrusion tolerance is the ability of a system to continue providing (possibly degraded but) adequate services after a penetration (Stavridou, 2001) As mentioned above, it is very hard to detect and prevent the DDoS attacks. Therefore, the intrusion tolerance of DDoS attacks is an important issue to mitigate the damage during DDoS attacks for providing the critical services continuously on Internet. Park and Lee (2001) suggested a method to install packet filters at different autonomous system on the Internet to filter out attacking traffic traveling between them. The method is very effective but not practical to defend DDoS attacks because of requiring the cooperation of thousands of autonomous systems on Internet.

Chang (2002) also introduced the concept of Internet firewall and four detecting and filtering approaches consisting of ubiquitous ingress packet filtering, router-based packet filtering, local attack detecting, and distributed attack detecting for defending flooding-based DDoS attacks. He also indicated that more effective detect-and-filter approaches, such as distributed attack detecting, should be developed for DDoS intrusion tolerance. However, all of them lack a systematic approach to integrate the knowledge of DDoS intrusion tolerance.

Xu and Lee (2003) proposed a DDoS intrusion tolerance system to sustain the availability of web service under DDoS attacks. The main idea is to isolate and protect legitimate traffic from huge volumes of DDoS traffic when an attack occurs. Unfortunately, it only aims on protecting web service instead of protecting the network.

Although a variety of methods have been proposed to mitigate the damage during DDoS attacks for providing the critical services continuously, it is still very difficult to keep up with the rapid growth of DDoS expertise in their studies. To solve this problem, a DDoS ontology is proposed to provide a common vocabulary among domain experts and an integrated KA framework is then proposed to assist in quickly accumulating their expertise. We will also use the behaviors of access control list (ACL) to evaluate the performance of the DDoS models.

2.3. Knowledge based system and NORM

Although expert systems simulating the decision-making ability of a human expert have been widely applied in many domains, only little research focuses on the DDoS intrusion tolerance. As we know, the knowledge of DDoS intrusion tolerance could be divided into several knowledge objects including various kinds of DDoS attacking tools and defending heuristics. The relationships between attacking characteristics of DDoS attacking tools and the defending heuristics may be too complicated to maintain; moreover, the different characteristics and the relationships of DDoS attacks in DDoS intrusion tolerance may be incrementally increased. A KBS with object-oriented model and rich relationships is required to achieve the modularity, sharability, and reusability of the rapidly increasing DDoS characteristics.

In Lin et al. (2003), a new object-oriented rule model (NORM) was proposed based on the concept of learning and thinking behaviors of human to provide high maintainability

and reusability through object-oriented concept. There are four basic relations between knowledge concepts defined in NORM: Reference, Extension-of, Trigger and Acquire. The Reference relation represents the association of two different KCs if the KCs have common piece of knowledge, which is useful for using original knowledge to construct new knowledge. Extension-of relation is used to extend or modify the KC constructed by other people, which is useful for knowledge sharing and exchanging. The Trigger and Acquire relations are used to represent the interaction of different KCs. Drama, a NORM knowledge modeled rule base system platform implemented using pure Java language, includes Drama Server, Console, Knowledge Extractor, and Rule Editor. Also, it provides Application Programming Interface (API) to access Drama server in Drama integrated systems. In our prototype system, the relations of NORM are used to represent the interaction of different KCs of DDoS intrusion tolerance.

3. Ontology and model of DDoS

As we know, the traditional methods for detecting and filtering DDoS attacks are monitoring the status of network and system, specifying the alert thresholds, defining detection rules, and setting filter policies by domain experts. Based upon interviewing with domain experts, the DDoS ontology proposed to models the behaviors of system and users, the methodologies of defense, and the strategies of evaluation are described as follows.

3.1. Ontology of DDoS

Before understanding the more complicated DDoS knowledge, an ontology, which is needed for sharing knowledge with a common terminology among numerous experts of the DDoS domain, could be divided into three parts: Profile model, Defense model, and Evaluation strategy as shown in Fig. 2. The Profile model is proposed to describe the behaviors of system and users according to the state and user state diagrams. The Defense model consisting of Detection and Filter methodologies is then used to resist the DDoS attacking. Finally, the Evaluation strategy including system and network evaluations is proposed to evaluate the performance of our proposed DDoS model. Hence, the ontology includes Profile model, Detection methodology, Filter methodology and Evaluation strategy knowledge classes (KCs), each may include several sub-KCs and may be obtained by interviewing with domain experts, e.g. Attack predicting and Attack detecting are sub-KCs of the Detection KC.

The Profile model KC (herein the Profile KC) and Evaluation strategy KC (herein the Evaluation KC) mentioned above could be easily acquired from domain experts.

The Detection methodology KC (herein the Detection KC) including attack preventing, attack detecting, and attack tracebacking technologies to defend DDoS attacks describes some useful features which could be selected by analyzing the characteristics/behaviors of attacking tools and the profiles defined in Profile KC. The predicting features including default communication behaviors, encryption behaviors, and encoding behaviors to predict the occurrence of DDoS attacks could be obtained by analyzing the behaviors of controlling traffic in DDoS attack tools. As for detecting DDoS, the behaviors of DDoS attacking traffic become most useful recently; e.g. three kinds of detecting features including TCP, UDP and ICMP are often used to detect the occurrence of DDoS attacks. Due to the massive of forge IP addresses and port numbers used by the spoofing techniques, both users and port numbers are divided into Trusted and Untrusted. Hence, the behaviors of Untrusted source IP could be used to detect the DDoS attacks. Similarly, the randomly generated port numbers are usually used in most DDoS attacking tools;

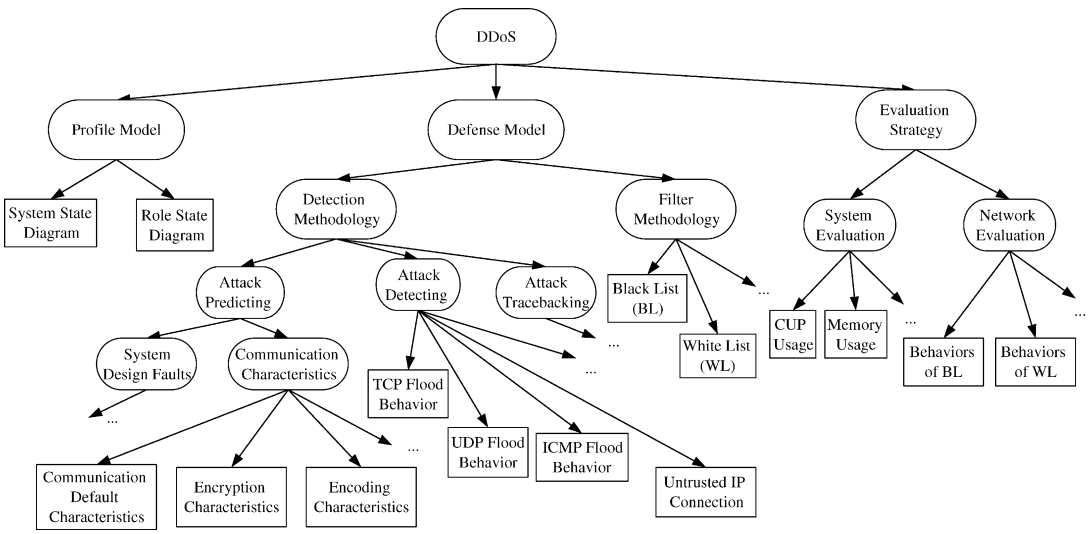

Fig. 2. The ontology of DDoS.

the behaviors of Untrusted destination port numbers could also be used to detect the DDoS attacks. In order to systematically select the suitable features for DDoS intrusion tolerance, a Characteristics Trainer is then proposed to obtain the significant characteristics, e.g. the specific attacking behaviors and the communication behaviors of DDoS attacks, which may be found by comparing the attacking traffic with the normal traffic according to the expertise. The attack tracebacking technology is still evolving and out of the scope of this paper.

The Filter methodology KC (herein the Filter KC) including black list and white list policies provides an efficient way to mitigate the malicious traffic during the DDoS attack. Various filtering policies, which are responsible for filtering out the malicious traffic, could be chosen in Filter KC according to the current system or network environments. To help domain experts adaptively tune the policies for filtering out the malicious traffic according to the response from other KCs, a recommendation of black list and white list policies can be provided by a User's Behavior Learner which is proposed using the role state diagram in Profile KC.

3.2. The relationships between knowledge classes

Four basic relations between KCs have been defined in Drama/NORM (Lin et al., 2003): Acquire, Trigger, Reference, and Extension-of relation. These relations are helpful in describing the relationships among KCs. Trigger relation triggers another KC with current facts as knowledge transfer. In other words, the remnant knowledge in original KC should not be necessarily considered. Acquire represents the acquirement relation. After Acquire process, the original inference process will continue and only facts predefined in the acquired KC will be carried back in Acquire relationship. Reference is used to represent the associations between different KCs. Through the Reference relation, the knowledge contained in referred KC is regarded as the base knowledge and will be taken into consideration together with the knowledge defined in the KC. On the other hand, Reference can be thought as an unconditional Acquire relation between KCs.

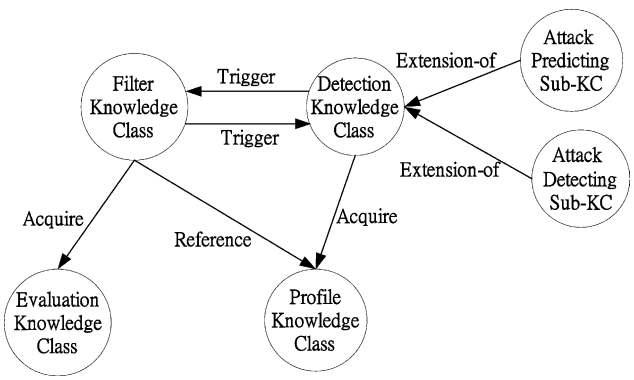

Fig. 3. Relationships between of knowledge classes.

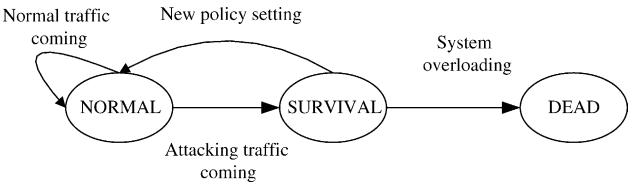

Fig. 4. System state diagram.

Unlike the Reference relation, the Extension-of relation makes a new KC to include all the knowledge contents of an existent KC. The activities of Extension-of relation include extension and modification. Therefore, it must support the overriding mechanism, including the overriding of facts and rules.

As shown in Fig. 3, the Filter KC referencing the Profile KC can be treated as a filter to filter out the malicious traffic and can be triggered by the outside traffic events. Also, it can set the new filter policy and acquire the Evaluation KC to evaluate the system performance. The Detection KC triggered by the Filter KC could be treated as a detector to detect the occurrence of DDoS attacks and could trigger the Filter KC according to the specific detection events for dynamically setting the new filtering policy to filter the malicious packets. Also, the Detection KC can acquire the Profile KC to set the suitable attack detecting or attack predicting sub-KCs, which are included by the Extension-of relation.

3.3. Profile model

According to the expert's experiences of defending DDoS attacks, the Profile model including system state and role state diagrams shown in Figs. 4 and 5, respectively, are proposed to represent the behaviors of system and users through interviewing with domain experts.

3.3.1. System state diagram

In the traditional systems, NORMAL and DEAD states are usually enough to describe the system status, because the time is too short to respond the DDoS attacks for the administrator before the victim system moving to DEAD state. Therefore, the behaviors of the most traditional information systems will be NORMAL-DEAD pattern when they are attacked. By applying the various filtering polices for DDoS attacks according to the users' behaviors, the survival time of the system can then be extended and the SURVIVAL state is further added to represent such situation. With the heuristic of setting more restricted filtering policies, the system will tend to move SURVIVAL

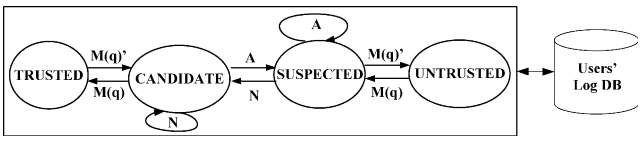

Fig. 5. Role state diagram.

state quickly backed to NORMAL state. On the other hand, the behaviors of the DDoS intrusion tolerance system will be expected to be NORMAL-SURVIVAL-NORMAL pattern during DDoS attacks.

To describe the relationships between states of the system conceptually, the system state diagram including NORMAL, SURVIVAL, and DEAD states are proposed as shown in Fig. 4. A system capability and the bandwidth utilization could be computed by the CPU load, the memory usage and the behaviors of ACL to determine the transition of the system state diagram according to the new policy setting event, the attacking traffic coming event, and the system overloading event.

3.3.2. Role state diagram

We assume the behaviors of normal users may not be changed dramatically in a short period of time. To represent the behaviors of each user, a role state diagram based upon an efficient ACL including white list and black list policy is proposed. We assume the DDoS attackers may frequently request service during the abnormal network status instead of normal ones. To monitor users' behaviors, the historical behaviors of the users based upon the current network status in a short time slice are proposed to distinguish the abnormal users from normal users efficiently, where each historical behavior is represented as a sequence of current network status.

According to the white list and black list shown in Fig. 5, all users could be categorized into TRUSTED, UNTRUSTED and CANDIDATE states. The N and A indicate that the current network status is normal and abnormal, respectively, and the user in CANDIDATE or SUSPECTED state is moved to CANDIDATE state when current network status is N; otherwise, he/she is moved to SUSPECTED state. Then, the $M(q)$ and $M(q)'$ indicate that the historical behavior of the user q, which is acquired from Users' Log DB, is normal and abnormal, respectively. The user in CANDIDATE state will be moved into TRUSTED state if he/she has accumulated sufficient normal behaviors; it means that the historical behavior of the user q should be normal, $M(q)$. On the other hand, the user in SUSPECTED state will be switched to UNTRUSTED state if he/she has accumulated sufficient abnormal behaviors, $M(q)'$. The users in TRUSTED and UNTRUSTED states will be moved to CANDIDATE and SUSPECTED states, respectively, if their historical behaviors match the corresponding constraints. With the role state diagram, the filtering policies could be adaptively generated based upon dynamically changing network environment.

3.4. Evaluation strategy

As mentioned above, there are two different types of DDoS attacking technique: bandwidth consumption and resource consumption. Since the only way we can stop a DDoS attack once it starts is to identify the addresses of all agents (zombies) sending DDoS packets and to shut off traffic from them, the behaviors of black list and white list are considered to monitor the potential latency of the network and the CPU usage and memory usage are used to monitor the degrading rate of system performance when suffering the DDoS attack and the *Tolerance* of the initial ACL is assumed to be maximal in this paper.

During resource consumption DDoS attacks, the system capacity of the victim is always decreasing (i.e. the system resource usage ρ is increasing), causing *Tolerance* low. On the other hand, the bandwidth consumption DDoS attacks may increase the members of the black list and decrease the members of the white list due to filtering the malicious traffic; hence the *Tolerance* will become small if the filter policy could not be performed well. Thus, the following formula combining network tolerance ($Tolerance_{network}$) and system tolerance ($Tolerance_{system}$) is given to represent the degree of DDoS intrusion tolerance, where α and β are used to indicate the weights of the tolerance. If we focus on protecting network performance, α is set to be larger than β. Otherwise, a large β is recommended to protect the system performance.

$$\text{Tolerance} = \alpha \times \text{Tolerance}_{\text{network}} + \beta \times \text{Tolerance}_{\text{system}}$$

Since the members in the white list (WL) may represent they could be served and the members in the black list (BL) may represent they could be blocked by the victim, the ratio of the WL and the BL is used to evaluate the network bandwidth utilization (Bw). A large BL implies the filtering policies are set to be more restricted and the tolerance may become small; otherwise, a large WL may represent the system is with more tolerance since more users can access the services of the victim. The number of users who have been moved from white list to black list ($W2B$) is further regarded as a penalty weight for network tolerance. The formula of the $Tolerance_{system}$ and the $Tolerance_{network}$ are shown as follows:

$$\text{Tolerance}_{\text{system}} = 1 - \rho$$

$$\text{Tolerance}_{\text{network}} = \frac{\text{WL}}{\text{BL} + \text{W2B}}(1 - \text{Bw})$$

The system state will be set as NORMAL when the *Tolerance* is larger than the predefined thresholds. Otherwise, the system state will be in SURVIVAL state and the new filter policy based upon Evaluation knowledge will be generated to move the state to NORMAL.

As mentioned above, each KC may include several sub-KCs due to the hierarchy of the knowledge in DDoS intrusion tolerance. In order to obtain the knowledge of each KC, an integrated KA framework including interviewing with domain experts, training the predicting and detecting features, and learning the filter policies for adaptively filtering malicious traffic is proposed. All knowledge of DDoS intrusion tolerance can be represented as a natural rule format, IF *Conditions* Then *Conclusions*. The bodies of

Conditions are the facts generated from network traffic flow, detecting results, and filtering policy. The *Conclusions* may include the alarming events, system state changing events, and user state changing events. Furthermore, constructing the knowledge base can facilitate the maintenance of the knowledge for defending DDoS and can help the administrators manage their networks.

4. Knowledge base construction

Due to the difficulty of acquiring and collecting the various DDoS characteristics from domain experts, an integrated KA framework including three related KA methods is used for reducing the effort of accumulating the expertise and speeding up the knowledge collection of DDoS characteristics.

4.1. The integrated knowledge acquisition (KA) framework

The problems that we are faced with during the KA process are usually very hard. In general, KA involves: (1) elicitation (gathering) of data from the expert, (2) interpretation of the data to infer the underlying knowledge or reasoning procedure and (3) guided by this interpretation, creation of a model of the expert's domain knowledge and performance. Although quite many different kinds of KA approaches have been proposed in many research studies (Hirasawa & Chu, 2003; Nikiforou & Fink, 2002; Rafea & Hassen, 2003; Tsai & Liu, 1999; Webb & Wells, 1999; Yan & Jiang, 2002) including interviewing with experts, Repertory Grids, and machine learning, few studies have focused on integrating various kinds KA approaches for an application due to the lack of a common vocabulary. Based upon the DDoS ontology we mentioned above, an integrated KA framework is proposed in this paper. Through the KA framework shown in Fig. 6, four kinds of KCs including Profile KC, Evaluation KC, Detection KC, and Filter KC could be easily obtained by various KA processes. Other new discovered or defined KCs can also be easily added or modified in our knowledge base due to the nature of KBS.

As shown in Fig. 6, all of these KCs of expertise can be obtained in the integrated KA framework, which includes modeling the DDoS environment through interviewing with domain experts, selecting useful features by analyzing the attacking tools in the Characteristic Trainer, and adaptively learning filter policies in the User's Behavior Learner. The behaviors of users, communication signatures, and other useful features are the characteristics of the Detection KC, so called detecting rules, which can be obtained by the Characteristic Trainer. On the other hand, the User's Behavior Learner is responsible for generating the various filtering policies in Filter KC. All other KCs including Profile KC and Evaluation KC can be directly obtained by interviewing with experts.

4.2. The knowledge obtained by interviewing with domain experts

As interviewing is one of the traditional approaches to acquire the expertise from domain experts by knowledge engineers, many approaches have been proposed to acquire expertise from experts through interviewing. As mentioned above, the Profile KC and the Evaluation KC could be modeled by domain experts using interviewing approach. Besides, the default knowledge including default communication ports and the white list and black list policy in the Detection KC and the Filter KC could be also acquired by interviewing with domain experts. Because the characteristics of DDoS attacking tools and the filtering knowledge are dramatically increasing, the Characteristic Trainer and the User's Behavior Learner are proposed to obtain the useful knowledge for DDoS intrusion tolerance.

4.3. Training the detection KC by Characteristic Trainer

To obtain the previously undiscovered DDoS characteristics/behaviors, a training process, namely Characteristic

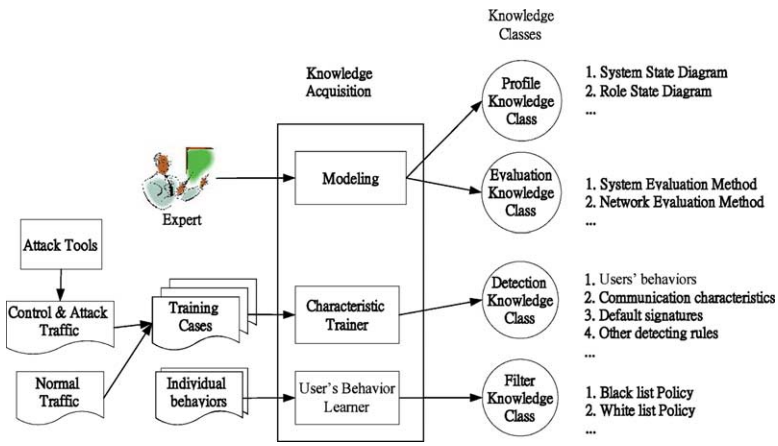

Fig. 6. The framework of KA process.

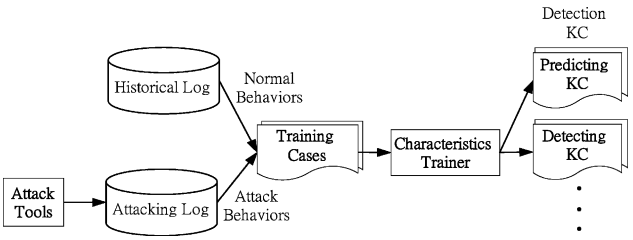

Fig. 7. The training process of Characteristic Trainer.

Trainer, is proposed to learn the useful, new features and store them into the Detection KC for predicting and detecting DDoS attacks.

4.3.1. Training process of Characteristic Trainer

Fig. 7 shows the new features of DDoS attacks can be selected by comparing the normal behaviors during the NORMAL system state with the attacking behaviors launched by DDoS attacking tools in the systematic training process of Characteristic Trainer. Thus, to distinguish attacking behaviors from normal behaviors, each kind of characteristics represented from the DDoS behaviors could be easily identified using a Repertory Grids approach, which is a table with four attributes including the feature, the feature threshold δ, the feature operation θ, and the corresponding actions. The δ is a parameter adaptively determined in a short period by different features. For example, if one feature value is larger than δ, it needs to be considered as abnormal behaviors. Therefore, the small δ will increase more false alarms. On the contrary, larger δ will treat more attacking behaviors as normal. After a DDoS attacking feature is detected, the corresponding action, e.g. trigger the Filter KC, alarm the attacking traffic coming, or specify the attacking type, must be taken.

In addition to the above previously known features, new characteristics/features may be observed by using the Repertory Grids approach after analyzing fingerprints of DDoS attacks such as spoofing, flooding-based and communication techniques and more attacks could then be detected and predicted.

The Characteristic Trainer algorithm

Input: Training cases TC, Actions A, and feature set F with n Features
Output: The Detection KC
Step 1: Select a skeleton feature set $F_k = \{f_1, f_2, \ldots, f_k\} \subseteq F$ by interviewing with domain experts.
Step 2: Set δ_i, for each feature f_i in F in each TC.
Step 3: Choose the proper actions A_i for the feature f_i selected in Step 2.
Step 4: Generate the detection rule as '**IF** f_i θ_I δ_i **THEN** A_i'.
Step 5: Repeat Steps 2–4 until all detection rules have been generated.

4.3.2. Examples of detailed rules obtained by the Characteristic Trainer

According to the Characteristic Trainer algorithm, the following six examples show the obtained partial characteristics/features and the corresponding detection rules

- *Example 1: the ratios of untrusted IP addresses and ports*. Since the attacking traffic of DDoS is launched from the compromised hosts which disperse on Internet, most of the attacking traffic comes from the untrusted IP addresses. Also, some DDoS attacking tools may generate the destination port of attack packets randomly. Therefore, we select the ratios of traffic from untrusted IP addresses and ports as two important characteristics in Step 1 to detect the occurrence of DDoS attacking traffic, set dangerous ratio $\delta = 50$ by comparing the list of ports and ACL defined by ISPs in Step 2, and the action 'Trigger Filter KC' for these two detecting features is chosen in Step 3. Finally, the rule 'IF (the ratio of untrusted IP addresses > 50) OR (the ratio to untrusted ports > 50) Then (Spoofing DDoS attacking) AND (Trigger Filter KC)' is generated to discover DDoS attacks.
- *Example 2: number of flows from the same source increase rapidly*. Generally speaking, the number of flows from one source normally does not increase dramatically if the attacks without spoofing did not occur. Hence, we select the number of flows from one specific source as a characteristic of suspected user to detect the DDoS attacks in Step 1. Next, the rapidly increasing rate $\delta = 50$ in Step 2. In Step 4, the rule 'IF (Number of flows from the same source > 50) Then (Same source DDoS attacking) AND (Trigger Filter KC)' is generated to discover DDoS attacks.
- *Example 3: number of flows in the state of 'SYN_RECEIVED'*. Resource consumption attacks often use the SYN flood technique (Criscuolo, 2000), such as TFN, TFN2K, stacheldracht and so on, to overwhelm the victim, where client sends a fake packet to Server, and Server accepts this SYN packet, responds the SYN/ACK packet to the fake address, and waits for the response of SYN/ACK packet. But the response will never arrive due to the faked source address. Therefore, the number of flows in the state of SYN_RECEIVED is first selected to detect the DDoS attacks in Step 1 and the δ is set to 1000 in Step 2 in this example. Finally, the rule such as 'IF (#SYN_RECIEIVED > 1000) Then (SYN-flooding DDoS attacking) AND (Trigger Filter KC)' is generated to discover DDoS attacks.
- *Example 4: percentages of UDP and ICMP packets*. The percentages of UDP and ICMP packets usually representing the error control messages during communications are always small in normal traffic and they will become large if the bandwidth consumption DDoS attack, the UDP or ICMP flood, could be launched to overwhelm victims. Since the percentages of UDP and ICMP packets are various for different autonomous

systems, the environment-dependent characteristics on monitoring the huge traffic of UDP and ICMP packets are required and selected to detect the DDoS attacks in Step 1. The δ is then set to be 30 in Step 2. Finally, the corresponding detection rules such as 'IF (P(UDP) $>$ 30) OR (P(ICMP) $>$ 30) Then (Flooding-based DDoS attacking) AND (Trigger Filter KC)' are generated to discover DDoS attacks.

- *Example 5: oversize of UDP packets and ICMP packets.* As mentioned above, the UDP and ICMP packets are usually the error control messages during communication, the encryption of payload in packet for DDoS tools is usually used for communication between attackers, handlers, and agents in controlling traffic, since the controlling traffic stores the source addresses of attackers, handlers and victims and attacks do not want to be revealed. Besides, the accounts, the passwords, and the commands to control handlers and agents from attackers are also necessarily encrypted. Since encrypting the information may enlarge the size of UDP and ICMP packets, the ratio of oversize UDP and ICMP packets might be selected as good features in Step 1 and set threshold value = 100. Finally, the corresponding prediction rules such as 'IF (#Oversize $>$ 100) Then (Controlling traffic attacking) AND (Trigger DDoS Prediction KC)' are generated to predict the DDoS attacks.
- *Example 6: percentage of BASE64 encoding packets.* Once the communication packets are encrypted, the resulting special characters should be encoded to avoid the occurrence of errors in communication during DDoS attacks. BASE64 encoding is very popular in solving this problem, but the payload of packet contains only alphanumeric character and 0×41('A') always appears at the end of the BASE64 encoding packet in the TFN2K attack. Both may be selected as useful features to predict the occurrence of DDoS attacks in Step 1. Finally, the predicting rule 'IF (AlphaBeta = Yes) AND (Tailing = 'A') Then (TFK2K Communication traffic attacking) AND (Trigger Filter KC)' is generated.

The following table shows the summarization of Examples 1–6

4.4. Learning the filter knowledge class by User's Behavior Learner

In traditional network management system such as firewall, intrusion detection system, network management system, etc. ACL is widely used not only to filter suspected connection from untrusted sources but also to admit the access from trusted sources black list and white list strategies used in ACL are included in the Filter KC. The former is used to interdict the access right, but the latter is used to permit the access right. Moreover, the various filtering policies can be set according to the configurations of current system and network environments.

In order to dynamically construct suitable ACL to mitigate the damage of DDoS attack, a learning process is also proposed to generate appropriate filtering policies for various network environments. The black list is used to drop the malicious attacking traffic and the white list is used to allow the trusted IP to access the critical service of the victim server. The principles of the User's Behavior Learner for ACL are twofold in this paper. One is to keep the legal users, whose behaviors are determined as normal, in the ACL. The other is to remove the possible suspected users or the users, who do not request critical service for a long time, from the ACL.

Since the normal user may not change his/her behavior rapidly, a user behavior scoring function is designed to calculate the score of each user by his/her own behavior to determine his/her status of historical behavior. The basic idea of the scoring function is to incrementally adjust the weight. If a current network status is normal, the score becomes larger; otherwise, it becomes smaller. The initial score value of each user is given 0 and the behavior of user changes from A to N or from N to A, the score is no change. To evaluate the historical network status of the user q, Γ is defined by expertise to determine the user's historical behavior network status. The historical behavior of the user ranging from $[MAX, MIN]$ is determined as normal ($M(q)$) if the score is larger than Γ. Otherwise, it is judged as abnormal ($M(q)'$) when the value is smaller than $-\Gamma$.

Feature name (f)	Operator (θ)	Threshold (δ)	Actions (A)
Ratio of untrusted IP	$>$	50	(Spoofing DdoS attacking) AND (Trigger Filter KC)
Ratio of untrusted ports	$>$	50	(Spoofing DDoS attacking) AND (Trigger Filter KC)
Ratio of same IP	$>$	50	(Same IP DDoS attacking) AND (Trigger Filter KC)
#SYN_RECIEIVED	$>$	1000	(SYN flooding attacking) AND (Trigger Filter KC)
UDP percentage	$>$	30	(Flooding-based attacking) AND (Trigger Filter KC)
#Oversize packets	$>$	*100*	(Controlling traffic) AND (Trigger Prediction KC)
AlphaBeta	=	*Yes*	(Communication traffic) AND (Trigger Filter KC)
Packet tailing	=	'A'	(Communication traffic) AND (Trigger Filter KC)

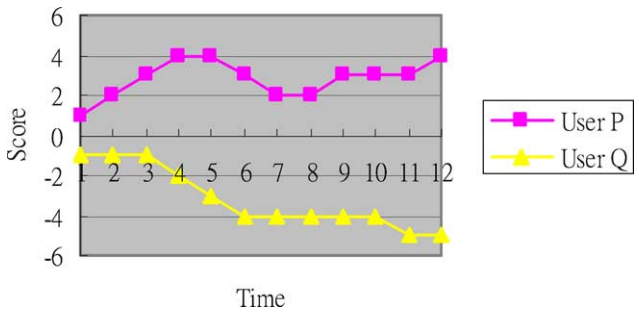

Fig. 8. An example of users' behaviors.

In Fig. 8, an example of $P = \langle N, N, N, N, A, A, A, N, N, A, N, N\rangle$ and $Q = \langle A, N, A, A, A, A, N, A, N, A, A, N\rangle$ is given to explain our proposed scoring function. The final score of P and Q are 4 and -5, respectively. If $\Gamma = 3$, the behavior of P is M and the behavior of Q is M' in this example.

To more accurately categorize the user behavior, a penalty weight (PW) could be attached when the characteristic of DDoS attack is discovered from individual user. The range of PW could be also defined according to the degree of dangerous behavior.

5. The verification of selected features/characteristics of DDoS

To evaluate the selected features/characteristics of DDoS attacks, the selected characteristics of DDoS which are similar to the normal behaviors will be eliminated due to the reduction of the false detection rate in the DDoS intrusion tolerance system. And then the Drama-based DDoS intrusion tolerance system will be implemented to evaluate the performance of detection power by the selected characteristics.

5.1. NORM-based DDoS intrusion tolerance system using KCs

In order to evaluate the efficiency of the KCs, the DDoS intrusion tolerance system using KCs is implemented by an inference engine Drama for detecting and predicting the occurrence of DDoS attacks. As shown in Fig. 9, the four KCs could be easily used to infer the other system components through the inference engine, and each component can be easily replaced according to different configurations of the network environment. The network traffic can be characterized as feature vectors (Lin et al., 2002) by Feature Vector Constructor. The Facts Collection is used to store all facts including the feature vectors, detecting results, system status, user states, and system evaluation results.

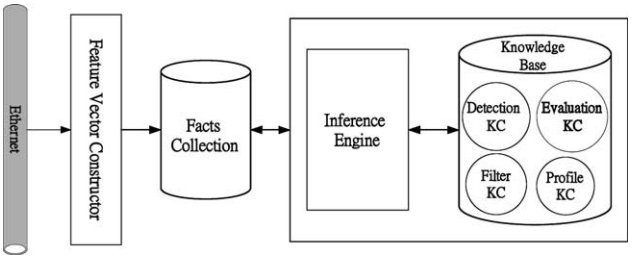

Fig. 9. The NORM-based DDoS intrusion tolerance system.

When the anomaly network traffic is detected in the Detection KC, the event of attacking traffic coming is triggered and the Profile KC is acquired to change the state of system. And the alarm event is thus triggered to set the suitable ACL in Filter KC for dropping the huge traffic from the attackers and allow the legitimate traffic from trusted users. Since then, the event of updating filter rules would be triggered to generate a new filter policy for dropping the malicious traffic. It implies the Evaluation KC will be acquired and used to compute the tolerance of the system for updating the filtering policy of the Filter KC. Finally, the Profile KC would be triggered to indicate the proper system state. However, the complex attacking behaviors sometimes make the filtering policy fail. When it is failing, the event of generating new filter policy would be triggered again until system state in Profile KC is stable. Otherwise, experts are asked to solve the problem of DDoS attacks.

5.2. An example of knowledge classes

The following shows a simple example of KC to defend the TFN2K attack, where the Profile KC includes system state transition rules and user state transition rules, the Detection KC includes attack detecting and attack predicting sub-KCs, the Detection KC includes rules focusing on detecting TFN2K attack, the Filter KC lists the heuristics of policy learning proposed in this paper, and the Evaluation KC illustrates the evaluating rules including system performance evaluating rules and the network performance evaluating rules.

A partial Profile KC

System state transition rules

IF Ss = NORMAL AND Traffic = Normal Then Ss = NORMAL //Ss is system state

IF Ss = NORMAL AND Traffic = Attack Then Ss = SURVIVAL

IF Ss = SURVIVAL AND Policy_Set = Enable Then Ss = NORMAL

IF Ss = SURVIVAL AND System = Overload Then Ss = DEAD

User state transition rules

IF Us = TRUSTED AND Uhb = M' Then Us = CANDIDATE //Us is user state

IF Us = CANDIDATE AND Uhb = M Then Us = TRUSTED //Uhb is historical user behavior

IF Us = CANDIDATE AND Ncb = N Then Us = CANDIDATE //Ncb is current user behavior

IF Us = CANDIDATE AND Ncb = *A* Then Us = SUSPECTED
IF Us = SUSPECTED AND Ncb = *N* Then Us = CANDIDATE
IF Us = SUSPECTED AND Ncb = *A* Then Us = SUSPECTED
IF Us = SUSPECTED AND Uhb = M' Then Us = UNTRUSTED
IF Us = UNTRUSTED AND Uhb = *M* Then Us = SUSPECTED

A partial Detection KC

Attack detecting sub-KC
IF P(Protocol) $> \delta_{\text{Protocol}}$ Then Flooding-based DDoS attacking AND Trigger Filter KC //P(Protocol) is the percentage of protocol
IF IR(U) > 50% Then Same source DDoS attacking AND Trigger Filter KC //IR(U) is the increase rate of the user
IF (Untrusted > 50%) AND (#Request > 200) Then Spoofing DDoS attacking AND Trigger Filter KC //UNTRUSTED is the rate of users in Untrusted
IF #SYN_RECEIVED $> \delta_{\text{SYN}}$ Then SYN-flooding DDoS attacking AND Trigger Filter KC
Attack predicting sub-KC
IF AlphaBeta = Yes AND Tailing = 'A' Then TFK2K Communication traffic attacking AND Trigger Filter KC //BASE64 encoding
IF #OverSize > 10 Then Controlling traffic attacking AND Trigger DDoS Prediction KC //Oversize packet

A partial Filter KC

Acquire Evaluation KC
IF $N_{cb} = N$ AND Us = CANDIDATE Then S(U) = S(U) + 1 //S(U) is the historical behavior score of user
IF $N_{cb} = A$ AND Us = SUSPECTED Then S(U) = S(U) − 1
IF $N_{cb} = N$ AND $N_{cl}(U) = N$ AND Us = TRUSTED Then S(U) = S(U) + 1 //N_{cl} is the last network state
IF $N_{cb} = A$ AND $N_{cl}(U) = A$ AND Us = TRUSTED Then S(U) = S(U) − 1
IF $N_{cb} = N$ AND $N_{cl}(U) = N$ AND Us = UNTRUSTED Then S(U) = S(U) + 1
IF $N_{cb} = A$ AND $N_{cl}(U) = A$ AND Us = UNTRUSTED Then S(U) = S(U) − 1
IF S(U) $\geq \Gamma$ Then $U_{hb} = M$
IF Us = TRUSTED Then Set U in *WL*
IF S(U) $\leq -\Gamma$ Then $U_{hb} = M'$
IF Us = UNTRUSTED Then Put U in *BL*
IF U in *BL* Then (Block U) AND (Policy_Set = Enable)
IF Policy_Set = Enable Then Acquire Evaluation KC
IF U in *WL* Then Trigger Detection KC

A partial Evaluation KC

IF CPU = X AND MEM = Y Then ρ = AVG(X, Y)
IF $\rho \geq 90$ Then System = Overload
IF $\rho \leq 45$ Then Traffic = Normal AND Policy_Set = Disable
IF ($\rho > 45$) AND ($\rho < 90$) Then Traffic = Attack
IF $Bw \geq 50$ Then $N_{cb} = A$
IF $Bw < 50$ Then $N_{cb} = N$

When the TFN2K traffic is coming, the Filter KC will be firstly triggered to filter the users in black list. Next, it will acquire the Evaluation KC to obtain the current network situation and then report abnormal ($N_{cb} = A$). In the meanwhile, the Filter KC will also trigger the Detection KC for detecting the users who pass the Filter KC. If the attacker lunched 'TCP-SYN flood' attack, then the rule '(IF #SYN_RECEIVED $> \delta_{\text{SYN}}$ Then SYN-flooding DDoS attacking AND Trigger Filter KC)' will be matched; hence the Filter KC is triggered to adapt the ACL. If one user's behavior is abnormal comparing to his/her historical behavior, he/she is gradually moved to black list and the Policy_Set is enabled. After new policy is set, the Filter KC will trigger Evaluation KC to determine the performance of the policy. If the system capacity become high, then system state will be transformed into NORMAL; otherwise, the policy will be reset if necessary.

6. Concluding remarks

In this paper, the ontology of DDoS is firstly given. Based upon the ontology, the models (including Profile model and Defense model), four KCs, and the relationships between these KCs are then proposed accordingly, where each KC may contain a set of sub-KCs or knowledge. To evaluate the degree of the *Tolerance* during DDoS attacks, two criteria including system capacity and network bandwidth utilization are proposed. For capturing the knowledge in the various KCs, a KA framework integrating interviewing, the Characteristics Trainer, and the User's Behavior Learner has also been proposed. Now, we are implementing a NORM-based DDoS intrusion tolerance system for DDoS attacks to evaluate the proposed models. In the future, we are going to develop an XML-based language for modeling the complicated environment of DDoS intrusion tolerance.

Acknowledgements

This work was partially supported by MOE Program for Promoting Academic Excellence of Universities under grant number 89-E-FA04-1-4, High Confidence Information Systems, and National Science Council of the Republic of China under grant No. NSC93-2752-E-009-006-PAE.

References

Barlow, J., & Thrower, W. (2001). *TFN2K—An analysis by Jason Barlow and Woody Thrower*, http://www2.axent.com/swat/swat.htm.

Bridis, T (2002). Powerful attack cripples majority of key Internet computers. *Yahoo! News*, Oct. 22, 2002.

Cabreraa, J. B. D., et al. (2001). *Proactive detection of distributed denial of service attacks using MIB traffic variables—A feasibility study*. *Proceedings of integrated network management, 2001 IEEE/IFIP international symposium*.

CERT Coordination Center, (2003). *DDoS attacks*, http://www.cert.org, 2002.

Chang, K. C. (2002). Defending against flooding-based distributed denial of service attacks: A tutorial. *IEEE Communications Magazine*, *40*(10).

Criscuolo, P. J (2000). Distributed denial of service-Trin00, Tribe Flood Network, Tribe Flood Network 2000, and Stacheldraht. *Technical Report CIAC-2319, Department of Energy—Computer incident advisory capability*, February 2000.

Dittrich, D. (2000). *DDoS: Is there really a threat? Proceedings of USENIX Security Symposium, August 16*.

Garber, L. (2000). Denial-of-service attacks rip the Internet. *IEEE Computer*, *33*(4), 12–17.

Hirasawa, S., & Chu, W. W. (2003). *Knowledge acquisition from documents with both fixed and free formats* (*Vol. 5*). *Proceedings of IEEE international conference on systems, man and cybernetics 2003*, (pp. 4694–4699).

Lin, S. C., et al. (2002). *A new mechanism of mining network behavior*. *Proceedings of PAKDD'2002, Taipei, Taiwan, May 6–8*, and Lecture Notes in Artificial Intelligence, 2336.

Lin, Y. T., et al. (2003). Design and implement of new object-oriented rule base management system. *Expert Systems with Applications*, *25*, 369–385.

Nikiforou, S., & Fink, E. (2002) (*Vol. 1*). *Proceedings of IEEE international conference on systems, man and cybernetics, 6–9 Oct*, (pp. 66–71).

Park, K., & Lee, H. (2001). On the effectiveness of route-based packet filtering for distributed DOS attack prevention in power-law Internets. *Proceedings of ACM Sigcomm, August*.

Rafea, A., & Hassen, H. (2003). Automatic knowledge acquisition tool for irrigation and fertilization expert systems. *Expert Systems with Applications*, *24*(1), 49–57.

Stavridou, V., et al. (2001). *Intrusion tolerant software architectures* (*Vol. 2*). *Proceedings of DARPA information survivability conference and exposition (DISCEX II'01), Anaheim, California, USA, June 12–14*.

Tsai, J. J. P., & Liu, A. (1999). Knowledge-based software architectures: Acquisition, specification, and verification. *IEEE Transactions On Knowledge and Data Engineering*, *11*(1), 187–201.

Webb, G., & Wells, J. (1999). An experimental evaluation of integrating machine learning with knowledge acquisition. *Machine Learning*, *35*(1), 5–23.

Xu, J., & Lee, W. (2003). Sustaining availability of web services under distributed denial of service attacks. *IEEE Transactions on Computers*, *52*(2), 2003.

Yan, H. M., & Jiang, Y. T. (2002). *Internet-based knowledge acquisition and management method to build large-scale medical expert systems* (*Vol. 3*). *Proceedings of the second joint conference on EMBS/BMES*, (pp. 1885–1886).