*I. J. Computer Network and Information Security*, 2018, 2, 53-64
Published Online February 2018 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2018.02.07

# A Multi-agent System-based Method of Detecting DDoS Attacks

**Xin ZHANG[1], Ying ZHANG[2,*], Raees ALTAF[1], Xin FENG[1]**
[1]School of Computer Science and Technology Changchun University of Science and Technology
[2]School of Control and Computer Engineering North China Electric Power University
E-mail: yingzhang@ncepu.edu.cn

Received: 13 December 2017; Accepted: 07 January 2018; Published: 08 February 2018

***Abstract*—**Distributed denial of service attacks are the acts aiming at the exhaustion of the limited service resources within a target host and leading to the rejection of the valid user service request. During a DDoS attack, the target host is attacked by multiple, coordinated attack programs, often with disastrous results. Therefore, the effective detection, identification, treatment, and prevention of DDoS attacks are of great significance. Based on the research of DDoS attack principles, features and methods, combined with the possible scenarios of DDoS attacks, a Multi-Agent System-based DDoS attack detection method is proposed in this paper to implement DDoS attack detection for high-load communication scenarios. In this paper, we take the multi-layer communication protocols into consideration to carry out categorizing and analyzing DDoS attacks. Especially given the high-load communication scenarios, we make an effort to exploring a possible DDoS attack detection method with employing a target-driven multi-agent modeling methodology to detect DDoS attacks relying on considering the inherent characteristics of DDoS attacks. According to the experiments verification, the proposed DDoS attack detection method plays a better detection performance and is less relevant with the data unit granularity. Meanwhile, the method can effectively detect the target attacks after the sample training. The detection scheme based on the agent technology can reasonably perform the pre-set behaviors and with good scalability to meet the follow-further requirements of designing and implementing the prototype software.

***Index Terms*—**Bayesian classifier; DDoS attack detec tion; Agent technology.

## I. Introduction

Security problems have a severe negative impact on the Internet and its relevant development and application, such as data tampering, information theft, and invasion of service sources, etc. The consequences would not only cause the service failures but also lead to some unimaginable incidents. DDoS attacks are made to run out of the limited service resources within a target service host, thereby preventing them from responding to network attacks that are legitimate and user-requested. DDoS attacks are designed to run out of the limited service resources on a target service host, thereby preventing them from responding to the valid user requests. From a practical point of view, the service resource exhausted by DDoS attacks could be time or space related. For example, a considerable number of password operations brought by a DDoS attack onto a target host that is in responsible for offering SSL service could not only consume up all the service resources but also occupy the whole service time. Then the valid users are then prevented from accessing the services within the host. A lot of spurious service requests can also be sent to the service host to take up the connection buffer space. When DDoS attacks are arriving, the target host is attacked by multiple coordinated attacker programs. So many forged packets are flooded from a large number of proxy hosts to the target service host, thereby limiting the planned communication between the client and the target service host. In the meantime, with hiding or tampering the source network addresses using technologically hiding the attacker identity, the source of the attack could hardly be traced, which could bring in the more severe consequences. Especially for those hosts in the critical information infrastructure that host key network services, the damage done by DDoS attacks is even greater. The most apparent difference between DDoS attacks and other cyber attacks is a large number of requests for data arriving in the short term. Therefore, the feasibility of DDoS attack detection scheme is to distinguish the network so-called "good" and "bad" connections, packets and sessions. Thus, it is of great importance to detect, identify, handle and prevent the potential hazards of network security effectively. This paper is devoted to the research of distributed attack denial of service (referred to as DDoS), which mainly focuses on the study of DDoS attack detection method and paves the way for the research on preventing DDoS attacks.

## II. Related Work

Overall, DDoS attacks can be executed at the network layer and the application layer of the network protocol, and their specific forms and corresponding detection methods are different according to the essential characteristics of each layer. Especially under the

 

condition that the network has rapidly grown and related technologies have made significant progress, DDoS attack detection technologies continue to face new challenges. First, the correct rate of detections needs to be improved [1]. Some DDoS attack strategies will dodge their attack traffic into non-malicious and sudden legitimate user traffic, which will result in the obstacles of attack recognition and put higher demands on the sensitivity of detection methods. Second, in practice, DDoS attacks do not always maintain static and fixed attack characteristics and methods. Instead, DDoS attacks change and adjust to the detection and defense measures taken by the target service host end, presenting dynamic and repetitive behaviors [2]. Third, the detection of DDoS attacks occupies and consumes resources of the target service host, and its operation process also requires a corresponding time cost. Therefore, it is necessary to synthesize the performance, complexity and time during constructing the DDoS attack detection method and seek a compromise fitting the needs of specific issues [3].

DDoS attack detection is one of the main research issues in the area of network security. At present, there are many academic and commercial organizations devoted to DDoS attack detection. Among the detection algorithms, Lemon J et al. proposed the SYN Cache detection strategy, which allocates a part of the resources on the target service host with a buffer-verification mechanism dedicated to temporarily storing the SYN request. They also suggested employing the SYN Cookies algorithm in cooperation with the above-mentioned strategy to carry out the detection onto the SYN flooding attacks [4]. According to the above detection strategy and algorithm, the buffer-verification mechanism delegates the target service host and performs establishing and verifying the connection. Only after the connection passes the verification, the target service host would actually take over the interaction through the connection. Through the above scheme, the target service host can be effectively prevented from being in a semi-connected state that the attacker performs a three-way handshake in order to establish a connection, thereby ensuring the optimized use of connection resources and reducing the security risk. Peng T et al. proposed an IP history-based method to filter the IP source address to separate the IP source addresses where the legitimate users are located from the dangerous ones. Moreover, the addresses could be filtered in the time dimension by employing a sliding time window mechanism. In this way, the target service host is enabled to circumvents DDoS attacks [5]. According to the characteristic that the number of hops in the routing of the data packet is constant, Wang et al. suggested to map the IP source address to the routing hop count and established a form tool to detect whether the sample data packet has falsification and realize the identification of the illegal data packets [6].

## III. Multi-agent-based DDoS Attack Detection Method

With the development of related technologies, DDoS attack detection technology needs to take full account of the technical requirements for real-time distributed scenarios under high load conditions to present the integrated and systematic DDoS attack detection solution within the target network [7]. Accordingly, DDoS attack detection technology solutions are required to be well adapted and have relatively low system resources and occupation when deployed. Based on the above considerations, it is feasible to construct a multi-agent system based solution to accomplish DDoS attack detection [8-10].

Agent technology is a kind of intelligent computing technology which is suitable for distributed application scenarios. It adopts the unit entities, i.e., the agents, constructed with the specific data content and the behavior(s) as the basic elements to build up the multi-agent system to resolve the problem. In this system, the problem modeling can help to identify and derive multiple agent communities. The agents belonging to the same community have the same behavior and autonomy and play the corresponding roles. Along the operations among agents, information exchange and data processing are carried out among the communities and agents, and the problem is handled autonomously according to preset rules and mechanisms till it is finally solved.

An agent can be regarded as a software program that pre-defines a specific behavior mode and carries corresponding data. It can be treated as an intention entity representing the user solution to the problem in the target system.

One agent is equipped with:

(1) Independent ability to be able to perceive the scene and implement the corresponding response;
(2) Autonomous decision-making ability, according to pre-set rules dynamically and in real time to select the appropriate strategy to execute behaviors;
(3) Survival target implies that the agents are designed to the same orientation within the system and make behavioral decisions to the system requirements;
(4) Interaction method, which can maintain the real-time interaction with other communities and the same community to ensure the overall information consistency;
(5) Flexible deployment solution that enables the rapid migration under distributed conditions and keeps sustained handling towards the targeted issues.

This paper employs the distributed agent technology to build the DDoS attack detection method and technical solution. The main work includes:

(1) Agent modeling, i.e., the identification of agent types based on the proposed agent system modeling methodology, and the formulation of agent behavior and interaction mechanism based on the methodology;

 

(2) Multi-agent system deployment. On the basis of agent modeling, the collaborative configuration of agents is realized, and a DDoS attack detection multi-agent system is constructed for the problem scenarios.

### *3.1. Bayesian Classifier Construction*

The Bayesian classification algorithm is used to construct a classifier to analyze and judge data samples in network traffic. Based on this, a judgment rule is provided for the agent to detect DDoS attacks.

According to the Bayesian theorem, Bayesian classification can be described as follows:

Given two sets F and V, there are

$$F = \{f_1, f_2, ..., f_n\}$$
$$V = \{v_1, v_2, ..., v_m, ...\}$$

With the mapping rule $f = T(v)$, any element $v_i (i = 1, 2, 3, ..., m, ...)$ in V can be only mapped to the only element $f_i (f_i \in F)$, i.e., $f_i = T(v_i)$. Here, T is the classifier and F is class set. fi is one single class and vi is one target class. The target of constructing a particular classifying method is to implement the above-mentioned classifier T.

We could construct a DDoS attack behavior classification onto the corresponding attack behaviors according to their properties. Through the accumulation and quantification of DDoS attack behavioral samples, the probability distribution of the attack behavior attributes using normal distribution is presented according to the commonly used statistical distributions of discrete events.

With calculating the average and standard deviation of the attributes of attack samples contained in the target class, we can compose the estimation and classification of samples of a given attack to achieve the DDoS attack behavior identification.

In other words, we could enable the agents to judge whether there would be some unsafe behavior. Then we will develop the corresponding agents and configure the behavior rules for DDoS attacks detection.

In general, the samples used to identify valid user requests are read from files, which are in our work defined as normal.files, and the output is constructed as a two-dimensional (2D) array, expressed as nf[][]. The sample used to identify abnormal user requests (i.e., containing potential DDoS attacks) is read from a file (defined as abnormal.file), and the corresponding 2D array output is pdos[][].

According to the Bayesian classifier constructed above, the training process to identify valid user requests can be expressed as follows:

(1) Parameter initial configuration. Set the count parameter (denoted as es) of the data sample unit counter and initialize it to 0. Set the packet count parameter (denoted as pc[]) and initialize it to 0, and clear the class count (denoted as cs) to 0. Then enter the second step;
(2) Determine whether the sample data file (i.e., normal.file) of the valid user requests is completed to read. If not yet done, read one row of data in the data file and then turn to step (3), otherwise go to step (6);
(3) Determine whether the required number (denoted as N) of the target packets has been achieved. If not finished yet, then turn to step (4). Otherwise, turn to step (5).
(4) Update the value of class count, i.e., cs, based on the value of pc[];
(5) Increase the value of the packet count to determine the type of identification that the current data content belongs to. Turn to step (2) after completing.
(6) Divide the count of each group by the number of data obtained and update the original value accordingly;
(7) The end of the process.

After the above process, the conditional probability value generated by the valid user request sample in normal.file is stored in the array nf[][]. If the file is changed to abnormal.file, the two conditional probability values will jointly constitute the conditional probability of Bayesian classification.

Before proceeding with the analysis and detection process, according to the detection principle described above, we could detect whether the arrived request belongs to a DDoS attack, during which the posterior probability is calculated as follows:

$$P(mR/k) = p(k/mR) * p(mR) / p(k) \qquad (R = 0, 1.....)$$

$P(k)$ is a constant for the file, and only $P(mR/k) * p(mR)$ needs to be calculated if to calculate $P(mR/k)$, during which $p(m0)$ and $p(m1)$ is the previously mentioned $p(ts)$ and $p(fs)$. $p(k/mR)$ is the product of multiplying the conditional probabilities of the various attributes.

The detection algorithm is described as follows:

(1) Initialize parameters: tr, fl, p (fs), p (ts) are initialized to the prior probability values in the initial stage. Set ws as 0 and then turn to step (2).
(2) Determine whether file b.txt is completed to read. If not done yet, turn to step (3). Otherwise, terminate the process.
(3) Determine whether the number of the existed packages is achieved to N. If the packages are ready, turn to step (4). Otherwise, turn to step (5).
(4) According to the above equations, calculate the priori probabilities under the normal and the abnormal conditions, then turn step (5).
(5) Set the package count as 0, and clear the counter as

0. Turn to step (7).

(6) Determine whether the data has been filled enough, and increase the package count. Turn to step (2).

With the algorithm, step (1) obtains the prior probability and initializes the parameters, and steps (2), (3) and (6) form the data unit. Steps (4) and (5) calculate the posterior probability of the data unit that has been formed and then determine whether there is a DDoS attack.

### *3.2. Multi-agent System Modelling*

The above Bayesian classification algorithm provides a theoretical basis for training to identify DDoS attacks. According to the principle of DDoS attack, combining with the scene of DDoS attacks, the identification of agent system is designed.

This paper proposes a target-driven agent-based system modeling method, the process includes:

(1) Target analysis: Taking the main targets of the multi-agent system as a starting point, carry out analyzing the attack detection target needs under the DDoS attack scenes and classifying the main targets. The targets are categorized into essential targets and optional ones.

(2) Conflict detection: Based on the category, analyze the essential targets to check whether there is some conflict(s) among the target specifications and eliminate some of the targets causing the conflict(s) to ensure the further modeling effort. Meanwhile, we employ the optional targets as complement to validate the conflict resolution and help revise some of the necessary targets.

(3) Plan design: Given the target set through the conflict detection, explain each of the targets with seeking the corresponding plans as the solutions to implement the targets. During composing the plans, there are several modeling methods can be employed. For example, IDEF-0 functional modeling can help specify and refine the target specifications; the UML sequence diagrams could support to model the procedures, the corresponding subjects and the collaboration among the subjects; the UML activity diagrams could refine the process and validate the built-up procedures, and meanwhile they could help evaluate the granularity of the various process models and iteratively refine the ones of larger granularity.

(4) Actor establishment: Relying on the obtained plans, identify and specify the subjects executing the plans. We identify and categorize the partial order relation between the subject and the actions within one plan as: subject-predicate relationship, verb-object relationship, cooperative relationship, and assembly relationship. With prescribing and modeling the relationships in the plans, we are enabled to validate the plans and make converting the plans into the agents as well as their behaviors feasible.

(5) Agent grouping and behavior configuration: According to the actors and their responsibility specifications obtained during the previous actor establishment, the agent entities could be design and the behaviors would then be configured.

(6) Multi-agent system development and debug.

When modeling multi-agent systems, we first analyze the system targets and find the necessary targets and optional ones. Combined with the principle and characteristics of DDoS attacks, we firstly design the initial functional modules and then analyze the plans generated with the targets. As follows, we decompose the main functions into the ones, each of which only is of one single functionality, according to the interrelations among the modules. With the functionalities, we re-organize the functionalities according to their characteristics. The re-organization is used to design the plans. Through analyzing the plans, we investigate their details for composing the potential agent behaviors. Finally, we design and develop the multi-agent system, in which the agents are interconnected and cooperate with each other to achieve the system targets.

### *3.3. DDoS Attack Detection Methodology*

#### *3.3.1. System Target Analysis*

Through the above analysis onto the DDoS attack behavioral characteristics, we start constructing the DDoS attack detection method with investigating the overall requirements and identifying the targets. We explore the following 10 essential and 2 optional targets.

Among them, the essential targets include: to monitor the network packages, to analyze data package information, to report the DDoS attacks, to learn the samples with Bayesian classifier, to configure detection rules manually, to present the detection result, to support further development of software, to record system log, to record the DDoS attack content and information and to propose defense schemes. The two optional targets include: to possess the autonomous learning capabilities and to report DDoS attack statistics. The targets are specified in Table-1.

 

Table 1. Target Category

| Type | Target | Description | Is integrated |
|---|---|---|---|
| Essential target | monitor the network packages | Monitor network traffic, determine DDoS attacks based on network traffic changes, and provide real-time traffic to determine whether attacks are triggered. | Yes |
| | analyze data package information | Analyze the packet parameters and content, provide the visual data analysis to the administrator to determine the attack situation. | Yes |
| | prompt DDoS attacks | According to the detection result, the attack is detected and the test result is provided to the administrator for composing the corresponding strategy. | Yes |
| | learn the samples with Bayesian classifier | Based on Bayesian classifier, make the algorithm be with self-learning ability. | Yes |
| | configure detection rules manually | Allow to modify the relevant parameters to update detection strategy for the changing DDoS attack forms during the attack detection process. | Yes |
| | present the detection result | Present the DDoS attack detection results, provide network data traffic, system logs, packages and other information for analysis. | Yes |
| | support further development of software | Support API and support to develop software prototype in future. | Yes |
| | record system log | Provide system log management, support for extension applications. | Yes |
| | record the DDoS attack content and information | Record the characteristic data generated during the DDoS attack, such as network traffic information, data packets, and system logs. | Yes |
| | propose defense schemes | Provide data support for the follow-up custom defense plan. | Yes |
| Optional target | possess the autonomous learning capabilities | Form automatic learning ability to detect the changing-form DDoS attacks | Yes |
| | report DDoS attack statistics | Save DDoS attack signature data and support statistical query. | Yes |

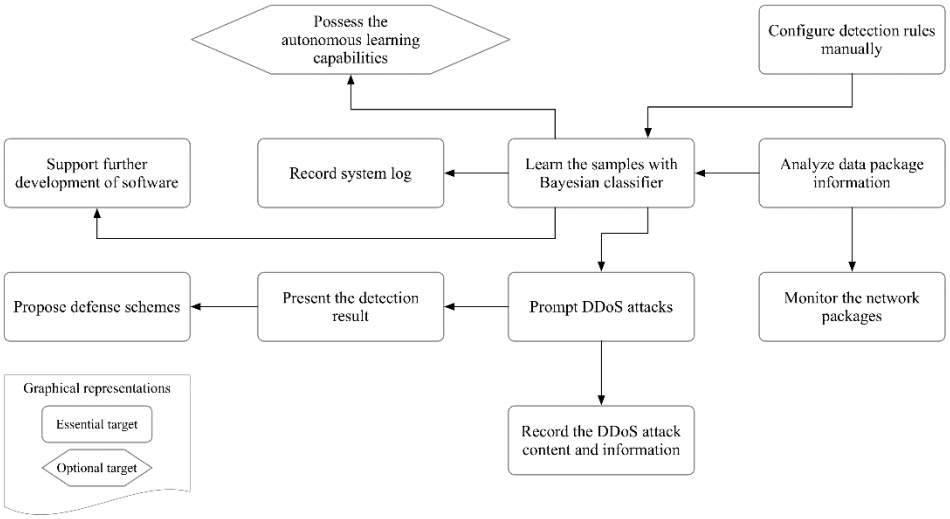


Fig.1. Relationships among Targets

The relationships among the targets are illustrated in Figure 1.

### *3.3.2. Target-oriented plan Analysis*

The initial requirements are the specifications of the overall functions of the system. Through decomposing each of the requirement modules, the particular functions can be identified. The obtained targets can not be mapped to the development requirements directly since there might exist some redundancy among the targets. Therefore, the requirements should be decomposed and refined into a set of functionalities. The similar functionalities would be combined to eliminate the redundancy.

(1) Plan specifications of monitoring network traffic

Along monitoring the network traffic, the network data traffic variation could help identify network attacks. Thus DDoS attacks are commonly detected based on the traffic variation since one of the most obvious feature of DDoS attacks is that the network traffic is increased drastically. The main attack target of DDoS is the servers or the cluster(s) within a sub-network unit. Therefore, monitoring the overall network traffic in a sub-network or a cluster does not only protect the computers within the unit area, but it could also avoid the cost of deploying the expensive monitoring devices (seen in Figure 2).

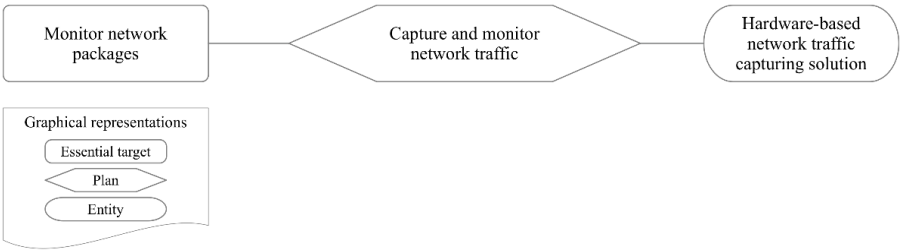


Fig.2. Plan Specifications of Monitoring Network Traffic

(2) Plan specifications of analyzing data package information

Analyzing data package information consists with three phases: catching the data packets, extracting the contents of the data packet and parsing the data packet attributes.

Catching the data packets: data packets contain the source address, port number and other information. Since most of the attack information are included in the packages, it is a proper way that to catch the data packets from the data packages. Through analyzing the caught packets, the characteristic information of DDoS attack could be identified. The catching solution could even be deployed at the source where DDoS attacks are launched.

Extracting the content of the data packet: the caught data packets are parsed according to the network protocols and then decomposed layer by layer. In this way, the IP address, port, protocol type, data packet content, etc. could be collected. With carefully and deeply analyzing all the parsed information, the characteristics of attacks could be learned.

Parsing the data packet attributes: In accordance with the requirement of detecting attacks, the relevant parameters used in the detection could be obtained through parsing the data packet attributes (seen in Figure 4).

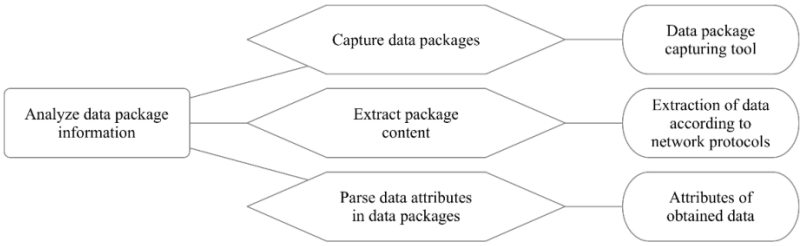


Fig.3. Plan Specification of Analyzing Data Package Information

(3) Plan specifications of prompt DDoS attacks

Prompting DDoS attacks is to report the attacks with some graphical user interface. It can be implemented in different ways including transferring the report to the network management center, pushing the report to the involved users and forwarding the attack-related parameters to some other modules.

Transferring the report to the network management center: once some attack is detected, a prompt is invoked in form of graphical user interface at the network management center which would make response.

Pushing the report to the involved users: with the message middleware deployed in the DDoS detection system, the detected attack occurrence and the relevant parameters would be pushed to the users or administrator who could prepare some defense and/or solution.

Forwarding the attack-related parameters to some other modules: because the transitivity of DDoS attacks, the message that one module is attacked by DDoS could be forwarded to other ones. The latter could prepare themselves in advance.

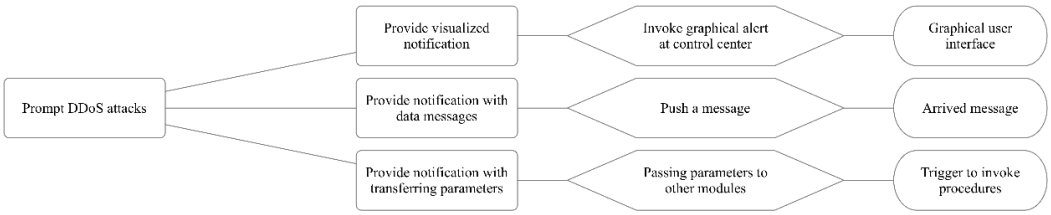


Fig.4. Plan Specification of Prompting DDoS Attacks

(4) Plan specifications of learning the samples with Bayesian classifier

Establishing the learning capability of Bayesian classifier consists with building up, training and verifying the classifier. Moreover, the detection rules allow to be adjusted to improve the detection accuracy.

Establishing the classifier is to model the conditions of detecting DDoS attacks and to identify the characteristic parameters. Training the classifier is to compute the appearance frequency of all the samples and to estimate and record the conditional probability of each categorized characteristic class. In other words, the input of the training is the characteristic attributes and the training samples while the output is the classifier. Verifying and debugging the classifier is to utilize the classifier to treat the classes and to determine whether the generated classifier could classify the samples correctly (seen in Figure 5).

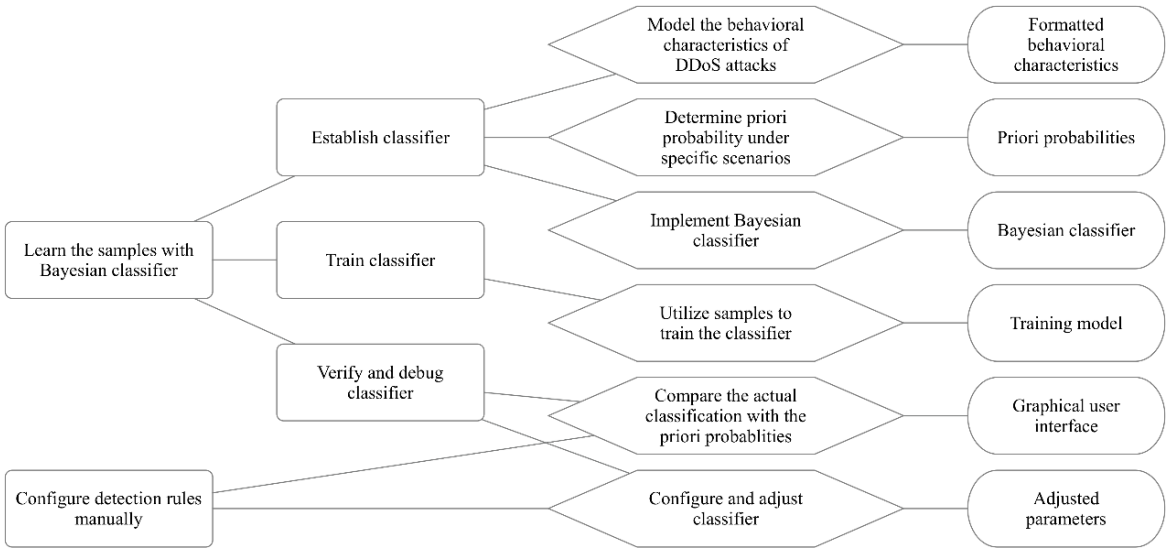


Fig.5. Plan Specifications of Learning the Samples with Bayesian Classifier

(5) Plan specifications of presenting the detection result

DDoS attacks-related information will be displayed through collecting the concerned data, such as network traffic, packet information, system logs and other information. All the information is transferred to the administrator with graphical user interface. The administrator is thus enabled to analyze the characteristic data effectively with the graphical user interface and then to make responses accordingly.

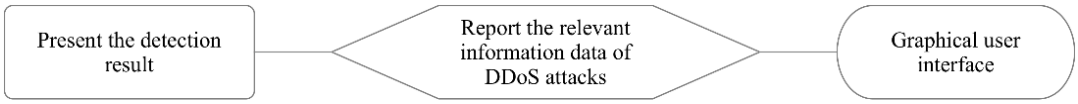


Fig.6. Plan Specifications of Presenting the Detection Result

(6) Plan specifications of recording system log

The local system log/state, the application system logs, the operating system logs are read and sent to some module to support the analysis made by the administrator (seen in Figure 7).

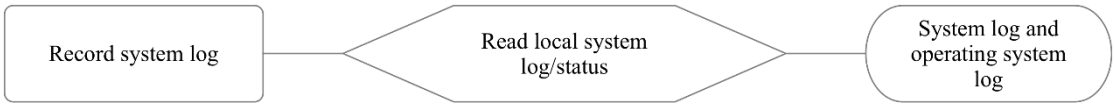


Fig.7. Plan Specifications of Recording System Log

(7) Plan specifications of recording the DDoS attack content and information

Recording attack information involves collecting the data packet information, the log information. Its main function is to collect the information of each module and store the information in the database or a file.

Obtaining the attack sample parameters is to provide support for analyzing the further attacks.

Creating a record object is to establish a record model according to the attributes, data amount and accessing method of the recorded object.

Selecting and implementing the persistence scheme is to provide different storage solutions corresponding to the different forms of data. For example, the data packets-related information could be stored in database while the system log could be stored in files (seen in Figure 8).

 

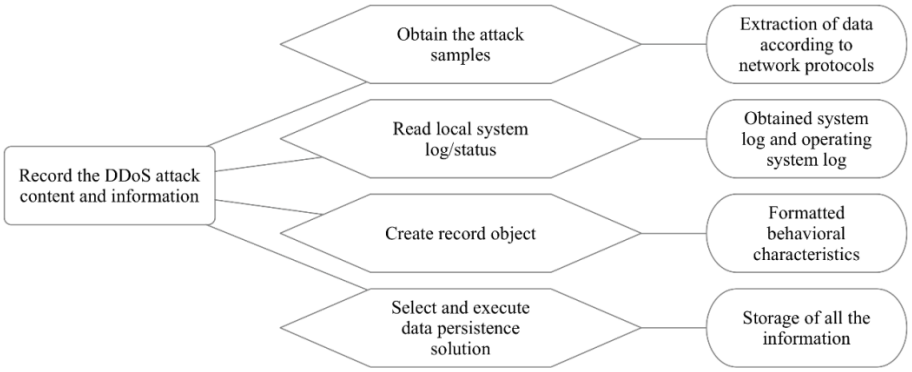


Fig.8. Plan Specifications of Recording the DDoS Attack Content and Information

(8) Plan specifications of proposing defense schemes

A defensive solution would be established through analyzing and coping with the arrived DDoS attacks, because once DDoS attacks occur, it will quickly find the next delegation which leads to the attack propagations. It is necessary to learn and categorize the classifications of the defensive strategies, attack behavioral characteristics and attack actions for constructing some proper defense schemes (seen in Figure 9).

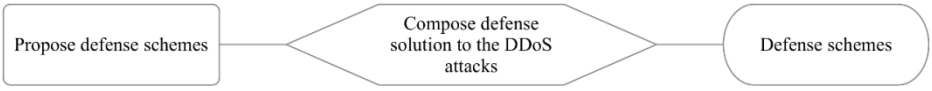


Fig.9. Plan Specifications of Proposing Defense Schemes

*3.3.3. Functional Entity Analysis*

In this section, through analyzing the above plan specifications, the relevant technical methods required to achieve the corresponding plan specifications are explored and utilized to generate the functional entities. The functional entities are identified for constructing the further agents.

(1) Capturing data packages with network monitoring tools

Command tcpdump could be used to monitor the content transferred in network.

(2) Extracting information according to the network protocol in the multiple layers

When the application layer data sent from one computer to another one through the network, the data would be coupled with a data header along each layer, i.e., the Ethernet frame encapsulation process. When the data arrives at the destination, a reverse process of the encapsulation needs to be performed, and each field in each protocol header is sent to a corresponding destination, i.e., the Ethernet frame parsing process. The concerned data related to the attack detection could be extracted through the above mentioned Ethernet frame data parsing.

(3) Parsing the corresponding attributes

With the extracted Ethernet frame data, the learned information would be re-organized according to the particular requirement.

(4) Displaying data with graphical user interface

The concerned data would be displayed with data visualization solution in the program of graphical user interface. All the data is preserved with either database or file system, which supports to report the real-time information as well as to query history.

(5) Pushing messages to administrator

With the support from some third party API, the detected attack would be reported to the administrator with pushing messages.

(6) Invoking the pre-set conditional trigger

Once the attack is detected, some conditional trigger(s) would be invoked as the response to deal with the effect brought by the attack.

(7) Formalizing the behavioral characteristic data

Given the collected attack occurrences, their identified behavioral characteristic data would be formalized through filtering the unused parts, classifying the characteristics, etc. and stored for further usage.

(8) Configuring priori probability of the behavioral characteristic data

According to the specification of Bayesian classifier, the relevant priori probability would be configured for support the classifier to learn the training.

(9) Adjusting the model parameters

Given the specific sample(s) leading to deviation, the adjustment would be made, such as increasing the number of the samples bringing in the deviation. Then the classifier would be re-trained and might be adjusted in further in aim of constructing a well-performed classifier.

 

(10) Getting the application log and the operating system log

The log produced by the applications and the operating system is of large amount and is store in the specific log files. With the graphical program, the log data could be read as required and support to analyze attacks.

(11) Storing the relevant information

The obtained data would be cleaned and stored for further usage. The data that could be formalized could be stored with database solution while the rest data would be preserved with files.

(12) Proposing defense schemes

The defense schemes would be proposed according to the occurred attacks. The schemes could be adjusted and managed with graphical user interface.

### *3.3.4. Constructing Agents*

Given the above entity analysis, the entities are grouped according to the functional similarity. In this way, some of the entities would be merged as new one. The output is listed in Table-2.

Table 2. Entity Integration

| Initial entity description | with similarity | Integrated entity description |
|---|---|---|
| Investigating network | No | Investigating network traffic |
| Capturing data packages with network monitoring tools | No | Capturing data packages |
| Extracting information according to the network protocol in the multiple layers | Yes | Parsing package information and extracting concerned parameters |
| Parsing the corresponding attributes | | |
| Displaying data with graphical user interface | Yes | Offering the API of data visualization |
| Adjusting the model parameters | | |
| Configuring priori probability of the behavioral characteristic data | | |
| Getting the application log and the operating system log | No | Parsing log files |
| Establishing the learning capability of Bayesian classifier | No | Establishing training data sample set |
| Training classifier | No | Training classifier |
| Pushing messages to administrator | Yes | Identifying/reporting attack arrival and supporting to compose further solution |
| Invoking the pre-set conditional trigger | | |
| Formalizing the behavioral characteristic data | Yes | Providing data persistence solution to treat the involved characteristic parameters |
| Classifying the relevant information | | |
| Storing the relevant information | | |
| Proposing defense schemes | No | Responding to attacks |

Through the above treatment onto the entities, a set of agents would be proposed. Referring to one agent, each of its behaviors is composed with several simple actions.

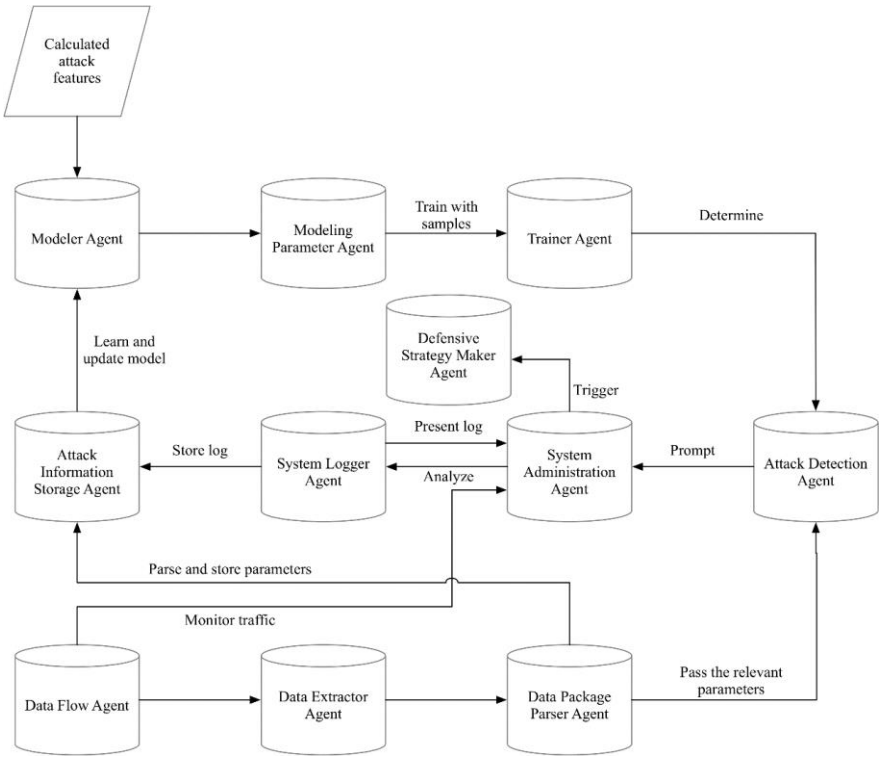


Fig.10. Communications among Agents

The identified agents cooperate with each other and compose the multi-agent system of detecting DDoS attacks. Each agent is specified as follows (seen in Figure 10).

(1) Modeler Agent: The main function is to set up DDoS attack characteristic model, combined with the built probability distribution through learning the actual attacks, according to the sequence number, time, source MAC address, source IP address, protocol type, source port number, system log and other information of DDoS attacks. All the above mentioned disparate information would be tagged and combined and used as the sample of training Bayesian classifiers. The Modeler Agent is responsible for reading the DDoS attack characteristic data, setting the priori rate conditions according to the related attributes and composing a set of classifier training samples.
(2) Modeling Parameter Agent: During the constantly detecting and learning process, the attacks would continue to produce new attacks, such as the attack frequency, the contents of the packet, the port number and the behavior characteristics. By updating the parameters in the learning process to deal with the new attacks, the detection accuracy could be improved. Along the continuous detection, the detection result will be displayed by another agent that is responsible for the visual control management. The administrator could be assisted to modify the relevant parameters of the training model according to the temporary result and train the classifier with the corrected training model to improve the accuracy of the classifier.
(3) Trainer Agent: The classifier learning process is the one of continuous training according to the training model. In the process of initialization, the classifier needs to train the detection model and the knowledge base which can detect the attacks according to the data model constructed manually by the classifier algorithm. Continuously using the attack characteristic data to train the detection model can improve the detection accuracy and the knowledge base information.
(4) Attack Detection Agent: It is the main agent to detect DDoS attacks, and it is also a product co-constructed by Modeler Agent and Trainer Agent. Meanwhile, attacks detected by Attack Detection Agent also act on modeling and training procedures, and the accuracy of detection could continue to increase during the interactions between the above mentioned agents.
(5) Data Flow Monitor Agent: It is used to monitor network traffic changes in real time, and initially determine whether there is a DDoS attack according to traffic changes. Only when it is identified that there may be an attack, the capture tool is activated to capture the packet, and the classifier is used to detect the DDoS attack. Real-time traffic monitoring can also help administrators determine whether a DDoS attack is arrived.
(6) Data Extractor Agent: Packet capture analysis is the entrance to detect attacks. Each attack detection triggers the attack detection behavior according to traffic changes. Packet capture network traffic analysis to see if there is network conflicts and network congestion. The detection result is sent to System Administration Agent, and the data is visualized and processed to monitor traffic changes in real time.
(7) Data Package Parser Agent: In combination with Data Extractor Agent, the control and management module will trigger Data Extractor Agent to capture the packet(s) when the network is abnormal, and then the packet will be analyzed by Data Package Parser Agent. The parameters will be sent to Attack Detection Agent. The latter will send the detection results to the concerned control and management center.
(8) System Administration Agent: The agent is responsible for controlling the detection system management and data visualization, and it could release operation code to the other modules and manage the cooperation among the agents.
(9) System Logger Agent: The agent is responsible for collecting system logs to identify the effect brought onto the located system and forwarding the data to System Administration Agent for supporting to make appropriate decisions on coping with attacks.
(10) Attack Information Storage Agent: The data collected by Data Package Parser Agent and System Logger Agent: are received by Attack Information Storage Agent. The latter would execute data persistence onto the received information.
(11) Defensive Strategy Maker Agent: It is responsible for selecting the pre-set strategy/action after the detected attack arrived.

Referring to design and develop the multi-agent system of detecting DDoS attacks, we suggest to adopt Java Agent Development Framework (JADE), an open source development engine of building multi-agent system initially developed by the researchers Telecom Italia Lab. Using JADE development engine, we are enabled to design and development the software prototype in Java Programming language and make full use of the basic conditions for the coexistence of multiple Java virtual machines. The JADE engine also provides internally configured communication method to the agents, which support to build up the interactions for the distributed deployment.

## IV. Result Verification

The verification work in this paper is executed with the multi-agent system developed based on the JADE engine, in which 11 types of agents are configured according to the detection methods. Specially there are 6 Attack Detection Agents deployed in the agent container. Each

 

of them coordinates with Trainer Agent to build up learning interaction with continuously receiving the concerned parameters, and it is also configured with the Bayesian classifier-based detection behaviors. In the verification experiment, the target detection data was inserted with 6 DDoS attacks samples.

A total of four simulation experiments were carried out, and data units of different sizes were adopted at each round to simulate the types of data packets generated by multiple service requests in various business scenarios. Assisted by Attack Information Storage Agent, the results of each round of was evaluated using the Confusion matrix [11] and the following results were obtained (see Table-3).

Table 3. Simulation Experiment Result

| Round | Data unit granularity | Number of detected attacks | Number of false positive | Detection rate |
|---|---|---|---|---|
| 1 | 5 | 128 | 118 | 70% |
| 2 | 15 | 16 | 4 | 70% |
| 3 | 25 | 6 | 0 | 70% |
| 4 | 40 | 6 | 0 | 70% |

Through the simulation experiments, we can see that the DDoS attack detection method based on multi-agent system proposed in this paper has better detection performance, less coupling to the granularity of the data unit, and can effectively detect the target form attacks according to the sample training.

In the process of simulation experiment, the multi-agent system can correctly implement various preset behaviors and returns the relevant operation data of the above simulation through System Logger Agent and Attack Information Storage Agent.

In addition, System Administration Agent configured in the system design provides effective system operation results and data contents through the TCP / IP protocol, which can meet the requirements of subsequent extension to realize the prototype of the software.

## V. Conclusion

In this paper, we provided a Multi-Agent System-based DDoS attack detection method based on the proposed target-driven multi-agent modeling methodology. With the modeling methodology, we are enabled to integrate the requirements of detecting DDoS attacks into the design solution to establish the corresponding agents and configure their communications. Associated by the Bayesian classifier, the particular agent is thus set with the specific behavior to analyze the samples and able to identify the potential attacks. With the suggested system trained with the samples, a better detection performance was obtained.

## Acknowledgement

This work was jointly supported by the National Natural Science Foundation of China (No.61305056), Beijing Higher Education Young Elite Teacher Project (No.YETP0702), and the Fundamental Research Funds for the Central Universities and the International Joint Research Project issued by Jilin Provincial Science and Technology Department (No.20150414055GH).

## Reference

[1] Mirkovic J, Reiher P. A taxonomy of DDoS attack and DDoS defense mechanisms[J]. Acm Sigcomm Computer Communication Review, 2004, 34(2):39-53.

[2] Zhou W, Jia W, Wen S, et al. Detection and defense of application-layer DDoS attacks in backbone web traffic[J]. Future Generation Computer Systems, 2014, 38(3):36-46.

[3] Sun Z X, Tang Y W, Zhang W, et al. A Router Anomaly Traffic Filter Algorithm Based on Character Aggregation[J]. Journal of Software, 2006, 17(17):295-304.

[4] Lemon J. Resisting SYN flood DoS attacks with a SYN cache[C] Proceedings of the BSD Conference 2002 on BSD Conference. USENIX Association, 2002:10-10.

[5] Peng T,Leckie C,Ramamohanarao K. Survey of network-based defense mechanisms countering the DoS and DDoS problems [J]. ACM Computing Surveys, 2007, 39(1):1-42.

[6] Wang H,Zhang D,Shin K. Detecting SYN flooding attacks[C]. In: Proc. of IEEE INFOCOM, IEEE Computer Society, 2002: 1530-1539.

[7] Zade M A R, Patil S H. A Survey On Various Defense Mechanisms Against Application Layer Distributed Denial Of Service Attack[J]. International Journal on Computer Science & Engineering, 2011, 3(11).

[8] Ismaila Idris, Obi Blessing Fabian, Shafi'i M. Abdulhamid, Morufu Olalere, Baba Meshach,"Distributed Denial of Service Detection using Multi Layered Feed Forward Artificial Neural Network", International Journal of Computer Network and Information Security(IJCNIS), Vol.9, No.12, pp.29-35, 2017.DOI: 10.5815/ijcnis.2017.12.04

[9] Ashish Kumar Khare, J. L. Rana, R. C. Jain,"Detection of Wormhole, Blackhole and DDOS Attack in MANET using Trust Estimation under Fuzzy Logic Methodology", International Journal of Computer Network and Information Security(IJCNIS), Vol.9, No.7, pp.29-35, 2017.DOI: 10.5815/ijcnis.2017.07.04

[10] Karanbir Singh, Kanwalvir Singh Dhindsa, Bharat Bhushan,"Distributed Defense: An Edge over Centralized Defense against DDos Attacks", International Journal of Computer Network and Information Security(IJCNIS), Vol.9, No.3, pp.36-44, 2017.DOI: 10.5815/ijcnis.2017.03.05

[11] Kai M T. Confusion Matrix[M]. Springer US, 2017.

 

## Authors' Profiles

**Xin ZHANG Ph.D.**, Lecturer at School of Computer Science and Technology in Changchun University of Science and Technology. His interests include: Mobile Internet, design engineering, software engineering and system engineering.

**Ying ZHANG Ph.D.**, Associated Professor at School of Control and Computer Engineering in North China Electric Power University. Her research interests include Artificial Intelligence, Urban computing, Next-generation Internet.

**Raees ALTAF** Master candidate at School of Computer Science and Technology in Changchun University of Science and Technology. His research interests include Wireless Sensor Network, Network Engineering.

**Xin FENG Ph.D.**, Associated Professor of School of Computer Science and Technology in Changchun University of Science and Technology. His research interests include Internet of Things technology and applications, software engineering and information system, database and data mining.

**How to cite this paper:** Xin ZHANG, Ying ZHANG, Raees ALTAF, Xin FENG,"A Multi-agent System-based Method of Detecting DDoS Attacks", International Journal of Computer Network and Information Security(IJCNIS), Vol.10, No.2, pp.53-64, 2018.DOI: 10.5815/ijcnis.2018.02.07

 