Research Journal of Applied Sciences, Engineering and Technology 5(7): 2439-2444, 2013
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: July 26, 2012 Accepted: September 12, 2012 Published: March 11, 2013

P2PRPIPS: A P2P and Reverse Proxy Based Web Intrusion Protection System

Qian He, Yong Wang, Yao Linlin and Qin Kuangyu
Key Laboratory of Cognitive Radio and Information Processing, Ministry of Education
(Guilin University of Electronic Technology), Guilin, 541004, China

Abstract: In order to protect web sites with various program languages and high throughput efficiently, a web Intrusion Protection System (IPS) based on P2P and reverse proxy architecture was designed and implemented. The P2P based web intrusion protection system has multi web firewall nodes and nodes with same program cooperate with each other under P2P architecture. Some nodes work as net flow allocator and some work as detector and they can convert to each other according to the requirements dynamically. The WAF program has the characteristics of session keeping and load balancing and it can detect messages by using expert library and many plug-in components. The technology of reverse proxy is used for response the web request. Experiments show that the system can effectively prevent attacks form application layer. It is proved more efficient and stable than single node.

Keywords: Peer-to-peer, reverse proxy, web intrusion protection system

INTRODUCTION

Internet web sites are used widely in today's society and almost all government departments, companies, colleges and research institutes have established their own websites. However, some malicious intrusions intruders easily exploit these vulnerabilities and flaws to attack the website, thereby undermining the normal operation of websites. The latest Top 10 list of security risks of web issued by OWASP (2010) (Open Web Application Security items Mesh) shows that web attack means like Injection, XSS (Cross Site Scripting), CRSF (Cross-site Request Forgery) has become the main threats to security of web server. China is also confronted with serious network security threats. The white paper: China Network Status which released in June 2010 shows that more than 100 million computer IP addresses are controlled by the hackers and up to more than 42000 websites are tempered in 2009 (News Office of State, 2010). Various cases show that many attacks have turned to the application layer, the traditional Firewall and web pages tamper-resistant device can not completely prevent such attacks. We need to install the Web Application Firewall (WAF) to protect the Web application. As a professional Web security tool, WAF is based on the bi-directional decoding and analysis of the HTTP/HTTPS traffic. It can not only deal with various types of security threats in HTTP/HTTPS application, such as Injection, XSS, CSRF, Cookie tampering, application-layer DDoS, but also help to solves the security issues like web tampering, pages hung horse, sensitive information leakage and other issues effectively, thus sufficiently guarantees the high availability and reliability of Web apply and accomplish the tasks beyond the capability of traditional IDS. Existing WAF products are inadequate mainly in the following aspects (Vigna *et al*., 2003; Liu *et al*., 2004).

Firstly, they are not flexible and efficient enough. Routing and core embedded deployment mode have a strong dependence on the network environmental. It is not convenient for rapid deployment. Both core embedded and transparent bridge modes are using single node deployment. When the concurrent flow is larger, it demands higher server performance, especially for the core embedded mode and it has a greater impact on service resources.

Secondly, its stability and function are not strong. A common feature of the existing structure is a single-section deployment. The entire web network will be limited to the collapse and lack of effective fault-tolerant mechanisms if a node fails. Though the existing products are designed in connection with the specific network or web server, its requirements for the cross-platform deployment remote protection and various host protection are relatively weak. So the applicability is still in need of improvement.

According to the Application Firewall evaluation standard released by WebAppSec (2006), the WAF mainly uses routing, core embedded, transparent bridge and reverse proxy architecture model, among which the routing mode sent only the HTTP/HTTPS flow to the detection system by configuring the routing policy, with poor real-time defense efficiency. The core embedded mode is embedded into web server through the ISAPI or Apache Module technology. Different platforms require different techniques. And the occupation of service

Corresponding Author: Qian He, Key Laboratory of Cognitive Radio and Information Processing, Ministry of Education (Guilin University of Electronic Technology), Guilin, 541004, China

resources has a certain impact on web server performance. Transparent cable mode refers to inserting system between two running devices. It has no influence on the flow. The system can block and filter the attacks from web application layer while let the other normal flow through, but it does not support remote agents or multi-domain detection (Vigna *et al*., 2003). The reverse proxy mode cascades WAF outside of the web server and sets the reverse proxy server IP to match with the web domain IP, so that it can hide the web server and has the advantage of deploying flexibly, expandability and supporting for remote monitoring.

Since single WAF can't satisfy the requirement of protecting many web sites with high throughput, a distribute architecture having multiple nodes is needed. Peer-to-Peer (P2P) technology develops very much recently and it is widely used in content sharing, instant messaging and so on. P2P has been seen as one of the important scientific and technological future of the impact of Internet. In the P2P system, each peer is both client and server and they can share and exchange resources directly (Androutsellis-Theotokis and Spinellis 2004; Liu *et al*., 2008). Therefore, P2P is a very good architecture to organize many WAF nodes.

In this study, we design a P2P and Reverse Proxy based web Intrusion Protection System (P2PRPIPS) that resists illegal website tampering and attacking for various program languages. P2PRPIPS is composed of multi web firewall nodes which have the same WAF program. Some nodes work as net flow allocator and some work as detector and they can convert to each other according to the requirements dynamically. All the nodes cooperate with each other under P2P architecture, each nodes complements part of the protection tasks. The WAF program has the characteristics of session keeping and load balancing and it can detect messages by using expert library and many plug-in components. The technology of reverse proxy is used for response the web request, therefore it can protect web programs written with various languages, such as ASP, JSP, PHP and so on.

OVERVIEW OF P2PRPIPS

P2P architecture: The web intrusion protection system: P2PRPIPS is composed by multi web firewall nodes and they are organized in the P2P protocol. In the P2P architecture, all the nodes have the same WAF program, but they can work as flow allocator or detector dynamically. From the function view of P2PRPIPS, the network architecture of is shown in Fig. 1.

The whole P2PRPIPS works as a reverse proxy server that receives a common user's request and then forwards it to the web server as well as receives web server response and forward it to the common user. At the same time it detects the request and response packets to prevent web attacks from the application layer.

The Net flow allocator and detector nodes are all work WAF program which can relay and detect the web request and response. The net flow allocator receives the request of ordinary users, then transmits the requests to the back of the detection server according to the characteristics of the message and load balancing algorithm and give feedback to the requesting user with final web response packet back on condition of maintaining the session. There are multiple flow detectors and they receive request packets sent by the load balancing server and the response packets by the web server. To detect the exchange packets using the expert database system which is based on regular expressions and special plug-ins and intercept the offensive message as well as record the attacks. The Monitoring center is used to monitor allocator server and detector servers into line monitoring, receive the status reports sent by each node and the attacking logs by the detectors, configure each node dynamically, transfer the node role and add or delete nodes.

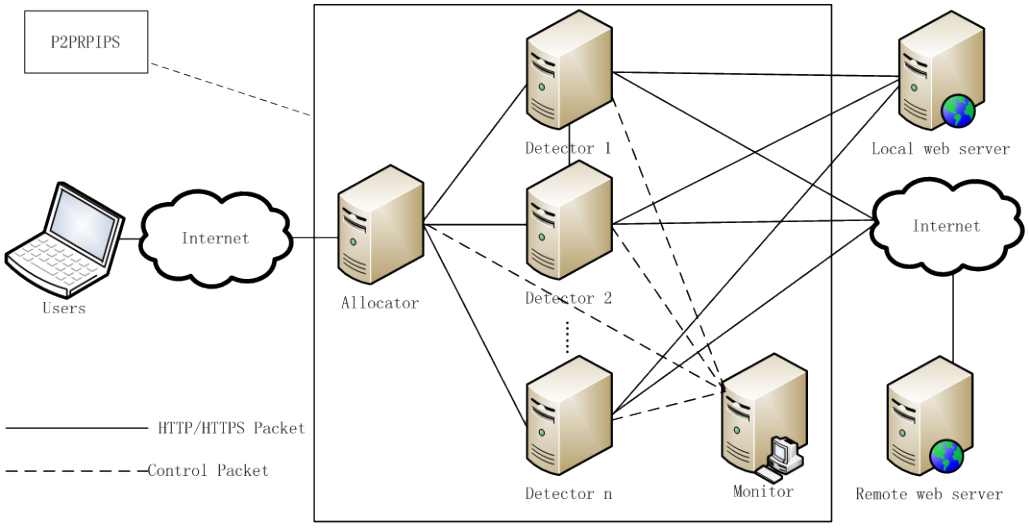


Fig. 1: The network architecture of P2PRPIPS

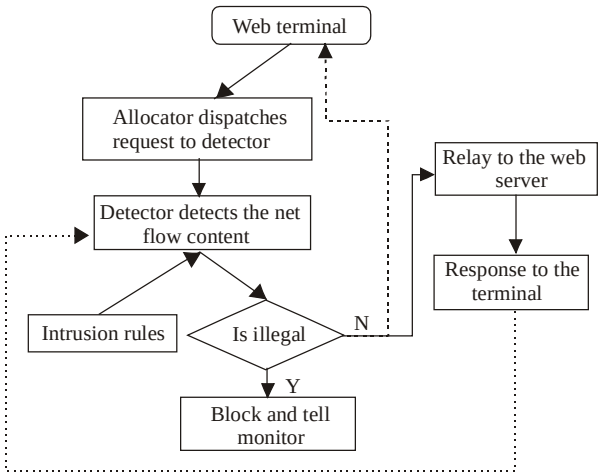


Fig. 2: The main workflow of P2PRPIPS

Operating mechanism: The main network model is the distributed architecture. The Net flow allocator receives the request and makes a balanced forwarding. Multiple secondary nodes (net flow detector) provide testing services. According to the configuration of its parameters, the detect node can support local, remote and multi-web servers testing. Each node adopts peer-to-peer model. Primary and secondary nodes use the same Code and either plays the role of load balancing server, or play the role of the server detection, depending on the configuration parameters. Besides, each node can substitute for one another according to actual needs, especially when the primary node breaks down, the secondary node takes over its work quickly.

When the node starts, it firstly reads its own configuration parameters, the main parameters of the configuration profiles named “self Responsibility conf” which is like as follows:

```
is Detector = false | true // Allocator or detector
balancing Algorithm = simple Hash
main Host = 202.193.74.199 // Detector node IP
main Host Port = 8881 // Detector node port
```

If it is allocator node, record the IP and port numbers of the auxiliary nodes in the multiAgent.cof file, otherwise, record web server IP, port and domain name information in the multi Host. conf. After the system starts up, the main process is shown in Fig. 2. In Fig. 2, the web request is tagged by real line and the response, by broken line. Web requests are dispatched by allocator to different detector nodes for analyzed in depth. All the request and response flow are filtered by the detectors and illegal flow is blocked.

NODE COMMUNICATION AND MANAGEMENT

Since there are many nodes in the system, how to corporately work together is very important. The node management including node joining, exiting and failure should be considered at first. In order to make full use of the WAF capacity, an efficient load balance allocator algorithm is needed.

Node management: There are three possible situations: node to join in, node exit and node failure in the node management and they are processed separately as follows:

- **Node to join in:** When adding nodes, firstly update the auxiliary node list of the master node, thereby adjusting the value of N of load balancing algorithm; Configuring the communication parameters of the joining node to ensure the smooth of communication. This action may cause Fn’s transformation, resulting in the session turbulence.
- **Node exit:** When the node exits, firstly update the auxiliary node list of the master node, disconnect exit node connection, this operation can lead to the interruption of the session.
- **Node failure:** When the auxiliary node fails, report to the monitoring center, implement the node exit operation. When the allocator node fails, a detector node is choose by the monitoring center to modify the parameters, transform the role. At the same time, other detector nodes update the new parameters to communicate with the new allocator node. Because of the dynamical switch between allocator and detector nodes, thus ensure that the system has strong stability and fault tolerance.

Node Manager's main problem is that whatever the circumstances, there will always impact on the user's session, at present, a simple load-balancing has been achieved, the algorithm is still necessary for further improvement. In addition, the Node Management should operate in the low-peak hours.

Load-balancing algorithm: Studies have shown that the XOR, bit shift principle based hash algorithm has good uniform property and it can meet the measurement requirement of the high-speed network traffic (Guang *et al*., 2005). Therefore, we select the string hash Code algorithm brought with java language to deal with the features string S1. In order to ensure balanced implementation of the tasks for each auxiliary node, but also to ensure the session consistency properties of each user's web connection, we extract HOST and AGENT domain from the user request packets, combined with the user IP make up the user's session features string S1, S1 = HOST + AGENT + IP. S1 is as the input of the hash function for choosing detector evenly. In order to make the hash result mapped to the various detectors, we mark the number of auxiliary contacts for N and Fn is conducting modulo N operation to the absolute value of hash result. The workflow of hash based load-balancing allocating algorithm is shown in Fig. 3.

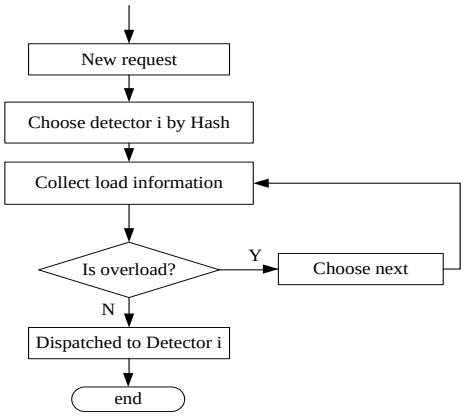


Fig. 3: The work flow of hash based load-balancing allocating algorithm

Node communication class design: Derived from various possible situations and functions of allocator and detectors, the inter-node communication related class diagram is shown in Fig. 4. Thread Pool class and Worker Thread class provide a thread pool and the worker thread, using the factory pattern to achieve. Message Listener class is used to listen for control packets request. Message Listener Observer class is used to response the changing state of Message Listener class. Message Listener Session class is used to create separate thread and deal with the request of packet, Simple Logger class adopts command mode and the observer model to realize recording log. Message Sender class is used to realize the independent control of packet.

DESIGN ABOUT WAF NODE

Reverse proxy: The reverse proxy mode cascades the system outside of the Web server and the IP of proxy server is set to match the IP and web domain name, in this way the web server can be hidden. The whole P2PRPIPS are worked as a web reverse proxy which accepts the requests of web terminals and responses them. In P2PRPIPS signature-based intrusion detection of web based attacks has been performed both at the network level. The reverse proxy opens up port 80 and 443 to receive the client HTTP and HTTPS, as shown in Fig. 5. A single thread will be build form the thread pool for each message from the client. And messages which detect to be safe will be forwarded to the web server by the messages forwarding module.

The allocator receives the Http request instead of the backward web server, while the web server is invisible from external, which allows only the detector nodes to access. The reverse proxy mode supports the remote deployment with no restrictions of geographical. Like an ordinary network router, it receives client requests and remote forward to the Web server.

In the actual network environment, the server's domain name matches the IP address of the reverse proxy server. For example, as shown in Fig. 3, we should set the reverse proxy's domain name is www.cc.com and the client wants to visit web server of 192.168.1.3 should access by the domain of www.cc.com. Through the reverse proxy, messages will be checked. And then HTTP messages are got and forwarded to the web server behind the proxy.

Intrusion detection engine: The basic functional requirements of the Web application firewall is the capability to receive the client request and server response, parse the request and response packets, decrypt the Https packets and choose whether forward or deny based on test results. The intrusion detection engine consists 7 sub modules including Packet monitoring, message management, IP analysis, Packet parsing, Coding parsing, Attack detection and Packet forwarding and the attack detection is the key sub module.

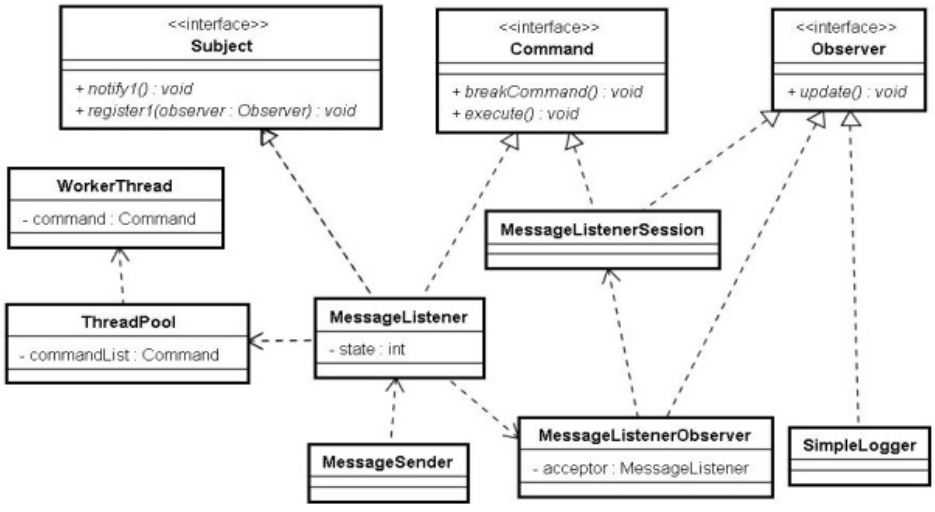


Fig. 4: The inter-node communication related class diagram

Table 1: Experimental server configuration

Name	Configuration
Web server	CPU Xeon E5503; memory: 4G; hard disk: 500; G win2003 server + IIS6/tomcat 6.0
Allocator	CPU Xeon E5503; memory: 4G; hard disk: 500; G win2003 server
Detector	CPU: dual E6700 3.2 G; memory: 2G; hard disk: 250G; win2003 server + JDK1.6
Test client	CPU: T2250; memory: 2G; windows XP
Swich	Huawei S3026

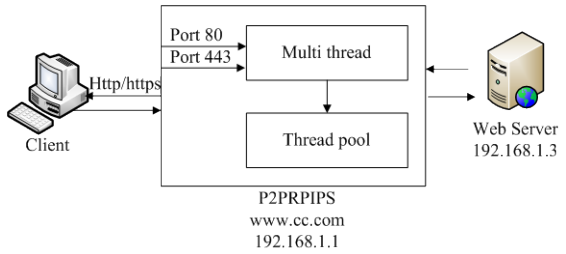


Fig. 5: Example of sending and receiving messages

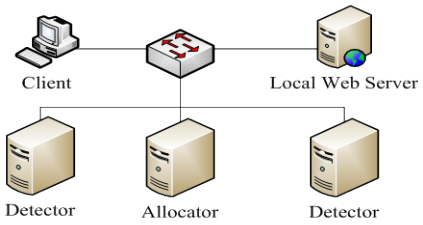


Fig. 6: Network topology of experiment

According to the attack rule library predefined by the system, match the Http message field set with the attack rule library. If attack actions appear, users are denied access. The attack detection module also detects the message server response, prevents the response from being sent to the client if the privacy leakage, data theft or other acts happen.

In the attack module, the message detection process includes black list and white list, abnormal request detection and anomaly response detection. No system is perfect and the only thing we need to do is ensure that the system is as perfect as possible. The reverse proxy detects the server response message, on which the sensitive keywords are detected and filtered. If the hacker have hacked the web server successfully and a response message contains a large number of customer information, bank account and other important information is found, The reverse proxy will prevent it immediately to ensure that the database information will not be stolen.

EXPERIMENT ANALYSIS

The experiment is conducted under single-node and multi-node deployment of web application firewall relatively, Topology shown as Fig. 6.

Since the allocator and detector have the same program indeed and they can work as WAF, only one allocator is used in the experiment of single WAF. All the clients and servers are connected in a 1000MLAN

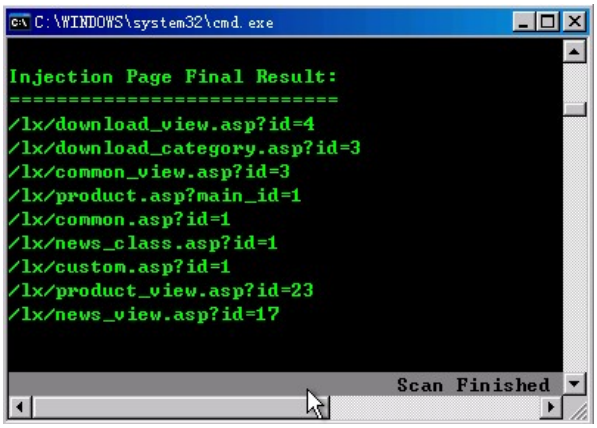


Fig. 7: The original scanning result

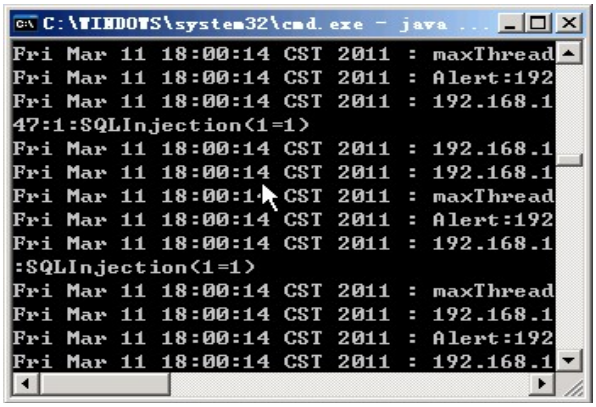


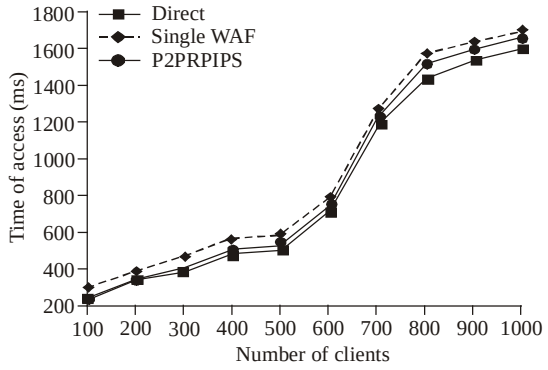
Fig. 8: P2PRPIPS logging the attack

and each server parameter and configuration is shown in Table 1.

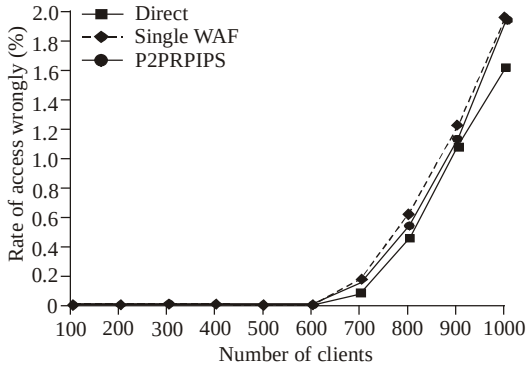
Function experiment: A simulating website is established using ASP programming and a SQL vulnerability detection tool: Wis.exe (Xiao, 2011) is used to scan that website. After scanning, there are nine possible SQL injection vulnerabilities. The scanning result is shown in Fig. 7.

Next, the simulating website is deployed on the P2PRPIPS and it is scanned by Wis.exe again. In this chance, all the SQL injection vulnerabilities are not found and the web proxy subsystem warns that there are attacks from test client which runs Wis.exe. The warning information is shown in Fig. 8.

Performance experiment: The Web server Stress Toos 7.2 is used to test the performance of P2PRPIPS. Two indexes, time of access and rate of access wrongly, are



(a) Rate of access wrongly



(b) Time of access

Fig. 9: Performance comparisons about P2PRPIPS

compared in three scenes: access directly, access under single WAF and P2PRPIPS. The experiments result is shown in Fig. 9.

By testing for three times with testing users varying from 100 to 1000 each time, due to the limit of hardware, it can simulate 1000 people on-line at most at the same time in the real environment test. The result value is the mean in three times. From Fig. 9a, we can know that the response time delay of either single-node deployment or multi-node deployment is less than 100 ms and P2PRPIPS is very close to accessing directly. Figure 9b shows that the error rate of P2PRPIPS is low. If the number of clients is lower than 600, there are no errors. In particular, P2PRPIPS can work better than single WAF. After deploying distributed multi-node, the performance improves 50 ms more than the single WAF.

CONCLUSION

In this study a web intrusion protection system based on P2P and reverse proxy architecture is provided, which works at the application layer and can deploy cross-platform, as well as supports the functions of remote proxy and multi-domain agency. Compared to other architectures, it is more simple and flexible. Meanwhile, P2PRPIPS uses distributed systems architecture, so that it solves the bottleneck problem of the inefficiently of single-node detection and supports high-speed detection of web application. The experiments show that the delay to the web services by the system is less than 40 ms and can effectively prevent malicious attacks to make sure the security of the protected web server. Because the processing time of P2PRPIPS is also bigger than accessing directly a litter, we will improve the allocator using caching technology to speed up accessing web servers in the future.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (No. 61163058, 61201250) and Guangxi Natural Science Foundation of China (2012GXNSFBA053174).

REFERENCES

Androutsellis-Theotokis, S. and D. Spinellis, 2004. A survey of peer-to-peer content distribution technologies. ACM Comp. Surveys, 36(4): 335-371.

Guang, C.H., G. Jian, D. Wei and X. Jia-Ling, 2005. A hash algorithm for IP flow measurement. J. Software, 16(05): 652-658.

Liu, R., N. Huang and C. Kao, 2004. A fast pattern-match engine fornetwork processor-based network intrusion detection system. Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC), 1: 97- 101.

Liu, Y., Y. Guo and C. Liang, 2008. A survey on peer-to-peer video streaming systems. Peer-to-Peer Networking Appl., 1(1): 18-28.

News Office of State, 2010. The White Paper of Status of Chinese Internet. Retrieved from: HTTP: // www. gov. cn/ zwgk/ 2010-06/08/ content_ 1622866.htm.

OWASP, 2010. OWASP Top 10 for 2010. Retrieved from: http://www.owasp.org/index.php/Category: OWASP_Top_Ten_Project, 2010.

Vigna, G., W. Robertson and V. Kher, 2003. A stateful intrusion detection system for world-wide web servers. 1th Annual Computer Security Applications Conference Proceedings.

WebAppSec, 2006. Web Application Firewall Evaluation Criteria. Retrieved from: Http://www. webappsec.org/projects/wafec.

Xiao, R., 2011. “WIS”. Retrieved from: HTTP: // www. netxeyes.com/ main.html.