International Journal of Engineering Research and Modern Education (IJERME)
Impact Factor: 7.018, ISSN (Online): 2455 - 4200
(www.rdmodernresearch.com) Volume 5, Issue 1, 2020

FIREWALL OPTIMIZATION TECHNIQUE USING SMART CONTEXT SENSITIVE METHOD

J. Britto Dennis*, U. Chindhya Baby & S. Sashikumar*****
Assistant Professor, Department of Computer Science and Engineering, Dhanalakshmi Srinivasan Institute of Technology, Samayapuram, Trichy, Tamilnadu

Cite This Article: J. Britto Dennis, U. Chindhya Baby & S. Sashikumar, "Firewall Optimization Technique Using Smart Context Sensitive Method", International Journal of Engineering Research and Modern Education, Volume 5, Issue 1, Page Number 6-11, 2020.

Abstract:

Firewall is a system that secures a network, shielding it from access by unauthorized users. A firewall is designed using firewall policy; it dictates how the firewall should handle applications traffic such as web, email or telnet, it also explains how the firewall is to be managed and updated. Certain error in firewall policy either creates security holes or blocks legitimate traffic which in turn could lead to irreparable, therefore to design firewall policy is an important issue. A firewall policy is designed by using three phase: a design phase, a comparison phase, and a resolution phase. The technical challenge in the method is to discover the functional discrepancies between given firewall policy. Firewall has become an obligatory part of every organization for the prevention of attacker from the internet. Due to the invention of new protocols, services and threats on the internet, it is essential to update firewall polices frequently to block unneeded services and allow needed services. The new rule added to the existing policy has to be placed in the correct order, should avoid dependency problem and rule ambiguity. This increases the complexity of firewall, reduces the throughput of the network which results in poor performance. Rule based firewall optimization techniques produces optimal reordering rule set which is semantically equivalent to the original rule set and maintain the performance and the throughput of firewall. For enhancing the rule based optimization, intelligent feedback mechanism is used so that rules are optimized periodically to achieve efficient performance.

Index Terms: Log Analyzer, Network Traffic, Firewall Policy, Legitimate Traffic & Intelligent Feedback Mechanism

1. Introduction:

With the large number of firewall solutions available today, firewall selection and implementation can be a time-consuming and overwhelming process. The appealing manner, in which "firewall" solutions are marketed, along with claims of easy installation and management, can lead organizations to make the decision to implement a firewall solution without taking time to thoroughly examine the need for one. By making hasty decisions, organizations can overlook the impact a firewall solution can have on their existing network and users.

Figure 1: Basic firewall

Organizations with a connection to the Internet or to any other untrusted network may need to implement a firewall solution. However, they should consider the impact a firewall will have on all network services, resources and users, and how a firewall will fit in with their particular business needs and network infrastructure [6].

A firewall is an integrated collection of security measures designed to prevent unauthorized electronic access to a networked computer system. It is also a device or set of devices configured to permit, deny, encrypt, decrypt, or proxy all computer traffic between different security domains based upon a set of rules and other criteria [1]. A system designed to prevent unauthorized access to or from a private network. Firewalls can be implemented in both hardware and software, or a combination of both. Firewall is frequently used to prevent unauthorized Internet users from accessing private networks connected to the Internet, especially intranets. All messages entering or leaving the intranet passes through the firewall, which examines each message and blocks those that does not meet the specified security criteria [7].

In this paper, we develop a general framework for rule based firewall optimization. Our framework precisely captures the semantics of an ACL in terms of whether each packet is accepted or rejected. To achieve this, it divides the packet space into independent *partitions* to correctly consider the changed set of packets matched by rules as the packets are processed within an ACL. In addition, compared to existing approaches, we require only that the action taken for each packet remains the same, rather than the rules themselves. With such a

precise model, our framework is able to find the optimal rule reordering [12]. Thus, it can also be used to compare and evaluate other optimization approaches and understand their practical benefits or limitations. This paper chooses to focus more on the optimality of rule orders generated by the optimization because its direct impact on firewall performance.

The objective of this paper is to develop software that performs packet filtering based on access policies and packet capturing to maintain log information and to monitor the network traffic .Firewall policy is of two types: High-level policy and Low-level policy, High-level policy is a service access policy, it's a part of network security policy. It defines TCP/IP protocols, services that are allowed or denied, service usage, exception handling and its goal is to keep the outsiders out. And Low-level policy is a firewall design policy; it's the refinement of service access policy for specific firewall configuration. The two approaches are open system which permits any service unless explicitly denied and closed system which denies any service unless explicitly permitted. Diverse Firewall Design explains about designing firewall where the policy is given to different teams later the functional discrepancy is found and new rule is being generated. Rule reordering technique is defined as the one which checks whether the rule matches the packet and checking whether the rules intersect and have different actions. Optimal rule reordering technique is formed which equivalent to original rule set which maintain the performance and throughput of firewall. The main objective of this paper is to generate a refined rule using diverse firewall design technique and rule reordering technique. The rest of the paper is structured as follows: section II presents details about the related works. Section III gives an overview of our proposed optimization technique. Finally Section IV concludes the paper.

2. Related Work:

Diverse Firewall Design: A firewall policy may consist of large number of rules, Ordering the rules correctly in a firewall is critical yet difficult. The implication of any rule in a firewall cannot be understood correctly without examining all rules listed above the rule. Correctness of a firewall policy is the focus of this paper. The method of diverse firewall design is used, which consists of three phases: a design phase, a comparison phase, and a resolution phase. In the design phase, the same requirement specification of a firewall policy is given to multiple teams who proceed independently to design different versions of the firewall policy. In the comparison phase, the resulting multiple versions are compared with each other to detect all functional discrepancies between them. In the resolution phase, all discrepancies are resolved, and a firewall that is agreed upon by all teams is generated. The data structure used in this paper for comparing multiple firewalls is Firewall Decision Diagrams. In this paper the method of Diverse Firewall Design is proposed which consists of three phases: design phase, comparison phase, and resolution phase. But there is no guarantee that there will be increase in the performance.

Discovery of Policy Anomalies in Distributed Firewalls: The core element in network security is firewall managing firewall rules, particularly multi-firewall enterprise networks, have become a complex and error-prone task. Firewall filtering rules have to be written, ordered and distributed carefully in order to avoid firewall policy anomalies that might cause network vulnerability. All anomalies that could exist in single and multi environment is identified. A set of techniques and algorithms to automatically discover policy anomalies in centralized and distributed legacy firewall is presented**.** These techniques are implemented in a software tool called the "Firewall Policy Advisor" that simplifies the management of filtering rules and maintains the security of next-generation firewalls[5] Therefore, inserting or modifying filtering rules in any firewall requires thorough intra- and inter-firewall analysis to determine the proper rule placement and ordering in the firewalls. But low-level filtering rules to perform a complete anomaly analysis and guided editing of centralized and distributed firewall policies are not applicable.

Rule Based Optimization: The complexity of modern firewall policies has raised the computational requirements for firewall implementation, potentially limiting the throughput of networks. Administrators currently rely on ad hoc solutions to firewall optimization. Firewall has become obligatory part of every organization for the prevention of attacker from the internet. Due to the invention of new protocols, services and threats on the internet, it is essential to update firewall polices frequently to block unneeded services and allow needed services [12]. The new rule added to the existing policy has to be placed in the correct order, should avoid dependency problem and rule ambiguity. This increases the complexity of firewall, reduces the throughput of the network which results in poor performance. Hence in this project we proposed a general framework for rule-based firewall optimization. In this we give a precise formulation of firewall optimization as an integer programming problem and show that our framework produces optimal reordered rule sets that are semantically equivalent to the original rule set. Rule based firewall optimization techniques produces optimal reordering rule set which is semantically equivalent to the original rule set and maintain the performance and the throughput of firewall. The limitation of this paper is that during optimization some packets take more time for re-ordering. Due to this we cannot assure that whether the firewall performance is improved.

Traffic Aware Firewall Optimization: A traffic aware optimization framework is developed to improve the operational cost of firewall [13]. Based on this framework, set of tools was designed to inspect and analyze both multidimensional firewall rules and traffic logs and construct the optimal equivalent firewall rules based on the

observed traffic characteristics. A novel adaptation mechanism is developed which dynamically detects anomalous traffic behavior and adaptively alters the firewall rules to avoid serious performance degradation due to the traffic anomaly. It provides full flexibility to reorder the rules based on traffic characteristics and also reduces cost. The limitation of this paper is that there is increase in process overhead and space complexity is high. This framework will only work for List based firewall.

Dynamic Rule Reordering Optimization: Previous techniques required impractically high space complexity, which undermines the performance gain offered by these techniques. Also, these techniques offer upper bounds for the worst case search times; average case scenarios are not necessarily optimized. The types of packet filtering fields used in most of these techniques are limited to IP header fields and cannot be generalized to cover transport and application layer filtering. So only we go for dynamic rule ordering optimization. This Technique presents utilizes internet traffic characteristics to optimize the traffic characteristic to optimize the firewall filtering policies. The technique adapts to the traffic conditions using actively calculated statistics to dynamically optimize the ordering of packet filtering rules [14]. The limitation of this paper is that rule overlapping is performed and it is difficult to create a new rule. It is also difficult to reorder the rule.

3. Our Contribution:

The implementation plan includes a description of all the activities that must occur to implement the new system and to put it into operation. Many people say that a firewall is only as good as its implementation. In other words, if a firewall is not implemented correctly, it may be ineffective. In complex network environments it can be easy to make mistakes during the implementation process.

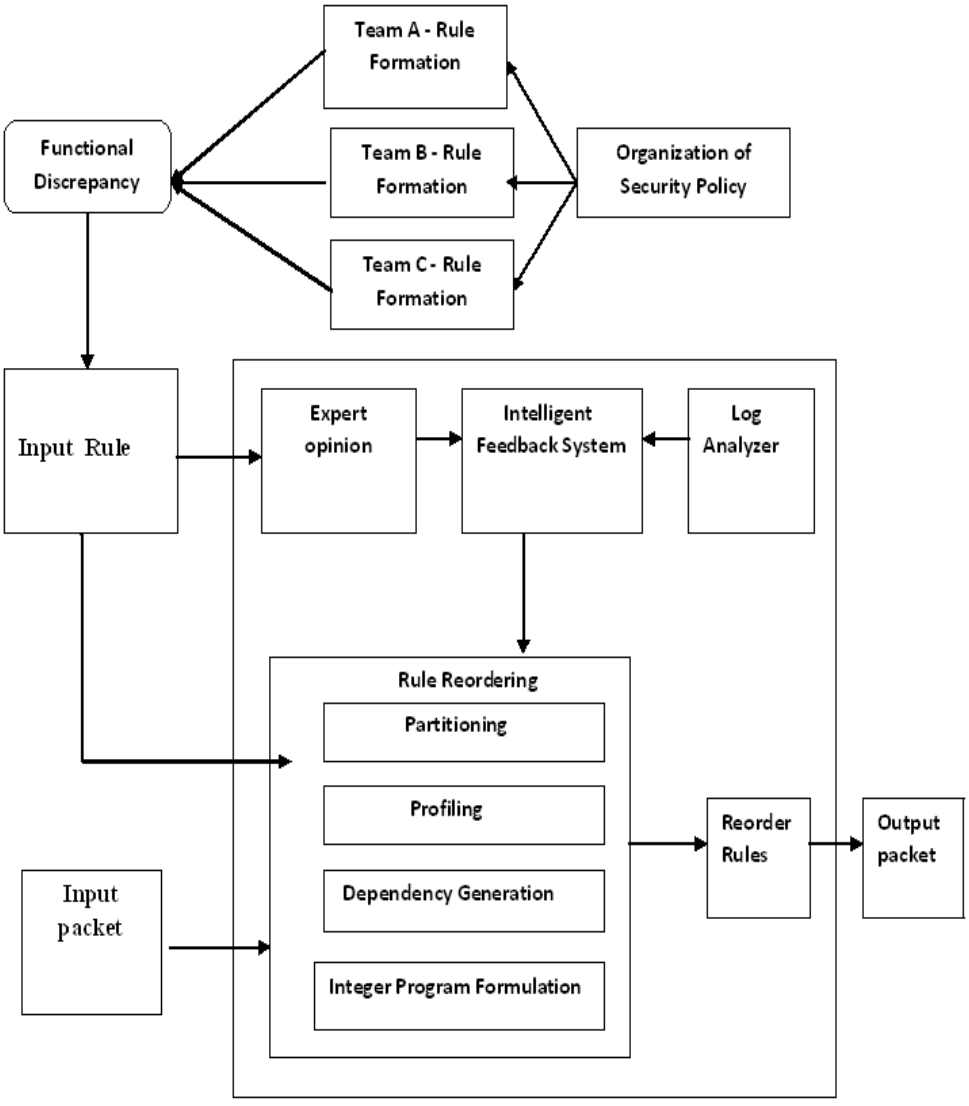

Figure 2: Architecture of Proposed System

Firewalls allow network administrators to offer access to specific types of Internet services to selected users. This selectivity is an essential part of any information management program, and involves not only protecting private information assets, but also knowing who has access to what. This can be done if the rule that is designed should be correct as it restrict access to certain users. The problem is given to different teams such as Team A, Team B and Team C. It undergoes certain phase such as design phase, comparison phase, and resolution phase. The major technical challenge in the method is to discover all functional discrepancies between two given firewall policies. Three algorithms are used for solving this problem: a construction algorithm, a shaping algorithm, and a comparison algorithm. These algorithms can be used directly to compute the impact of firewall policy changes by computing the functional discrepancies before and after the policy changes as firewall policy often needs to be changed, as network evolves, and new threats emerge.

Firewall has become obligatory part of every organization for the prevention of attacker from the internet. Due to the invention of new protocols, services and threats on the internet, it is essential to update firewall polices frequently to block unneeded services and allow needed services. The new rule added to the existing policy has to be placed in the correct order, should avoid dependency problem and rule ambiguity. This increases the complexity of firewall, reduces the throughput of the network which results in poor performance. Rule based firewall optimization techniques produces optimal reordering rule set which is semantically equivalent to the original rule set and maintain the performance and the throughput of firewall [12]

It contain two module -Firewall Optimization and feedback operation At first input of the packet will be received by reordering module along with the input rule which involving four methods of reordering namely partitioning, profiling, dependency generation and Integer program formulation. Intelligent feedback system collects the dynamic opinions regarding the reordering rules from the expert and it will modify the rule based reordering .The output of the process will be reordered rules. The overall architecture is given below

Diverse Firewall Design: In the design phase, the same requirement specification of a firewall is given to multiple teams who proceed independently to design different versions of the firewall. In the comparison phase, the resulting multiple versions are compared with each other to find out all the discrepancies between them. In the resolution phase, all discrepancies are resolved, and a firewall that is agreed upon by all team is generated.

Design Phase: In design phase, multiple firewalls are being designed .For example if the requirement specification for this firewall is given as follows:

The mail server with IP address 192.168.0.0/16 can receive e-mail packets. The packet from an outside malicious domain 152.163.80.11 should be blocked. Other packets should be accepted and allowed to proceed. Let the specification is given to two teams, TEAM A & TEAM B

Rule	Interface	Source IP	Destination IP	Destination Port	Protocol	Decision
r1	0	*	192.168.0.1	25	TCP	Accept
r2	0	152.163.80.11	*	*	*	Discard
r3	*	*	*	*	*	Accept

Figure 3: Firewall designed by Team A

Rule	Interface	Source IP	Destination IP	Destination Port	Protocol	Decision
r1	0	152.163.80.11	*	*	*	Discard
r2	0	*	192.168.0.0/16	25	TCP	Accept
r3	*	*	192.168.0.0/16	*	*	Discard
r4	*	*	*	*	*	Accept

Figure 4: Firewall Designed by Team B

Comparison Phase:

Method for computing functional discrepancy between two given firewall is done. It compares the relevant rules that are already proposed and generate a hybrid rule set.

#	Interface	Source IP	Destination IP	Destination Port	Protocol	Team A	Team B
r1	0	152.163.80.11	192.168.0.0/16	25	TCP	accept	discard
r2	0	!152.163.80.11	192.168.0.0/16	25	!TCP	accept	discard
r3	0	!152.163.80.11	192.168.0.0/16	!25	*	accept	discard

Figure 5: Functional Discrepancies between the Two Firewalls Designed by Team A and Team B

Steps to Reorder the Rules: The reordering of rules takes place according to partitioning, dependency generation and optimization of the existing rules. Rule-based Partitioning of Packet Space: Partitioning is used to divide the packet space into disjoint blocks according to the given ACL. Rule-based partitioning of the packet space is a key step in our optimization framework. The (disjoint) blocks of the partition are created such that for any two packets within a single block, the same set of rules from the ACL matches those two packets [12].

- Since all packets within a block will be matched by the same rule in any reordering of the ACL, checking for correct block action is sufficient;
- Cost assignment can be attributed to blocks rather than rules, thus making cost calculation independent of the choice of rule ordering.

Partition Profiling and Rule Cost: The profiling step then measures the weights of blocks within the partition. A good metric for ACL cost is the expected time to process a single packet. Intuitively, with a lower packet processing time, the firewall can achieve higher throughput. Traffic profiling tool to dynamically monitor the traffic and update the traffic profile as needed ProgME allows administrators to write a small definition for the partitions of interest and collects traffic statistics for each partition directly.

Dependency Generation: The dependencies between the relative positions of two rules that overlap can be determined, but with different actions. While this reasoning suffices for most dependencies, it can be too conservative in general. A dependency must not be between a rule pair. Instead, it should be between a rule *i* and a block's matching rules that have a different action from that of rule *i*. We denote these dependencies using the following format: *i depend {j, k, l}*.Such a dependency requires that rule *i* must follow the *earliest rule* of *{j, k l}*.

Optimization: Optimization step uses information from previous steps to produce an integer program whose solutions yield semantically equivalent, optimal rule reordering. Experimental results shows improved performance based on packet processing time. Normally in firewalls during rule based reordering, some packets take a long time for processing. So log analyzer is used to monitor all packet information. In Intelligent feedback system, whichever packet takes high processing time will be again reordered and performance is improved.

Intelligent Feedback System: In the previous optimization techniques there is no guarantee that performance has improved. So we are going for intelligent feedback mechanism, where we assure that there is improvement in this regard. Using log analyzer, we can easily identify information about the whole packet.

Log Analyzer: When a visitor surfs the internet, the server registers all of that visitor's requests to the server in a special log file. The program that analyzes the content of that file is called log analyzer. Logs are emitted by network devices, operating systems, applications and all manner of intelligent or programmable device. A stream of messages in time-sequence often comprises a log. Logs are directed to files or stored on disk. Typical reasons why we perform log analysis are: Improved security, System trouble shooting and network administration [11].

Intelligent Feedback System: Some packets take lot of time for processing due to rule reorder. So we are going for feedback mechanism which will lead improvement in performance and throughput of firewall. For example, if there three similar rules, feedback mechanism will identity it and combine them to single rule. This will result in improved performance

Export Opinion: Intelligent feedback system collects the dynamic opinions regarding the reordering rules from the expert and it will modify the rule based reordering .The output of the process will be reordered rules. This will lead to improved performance in the rule based reordering of firewalls

4. Experimental Result:

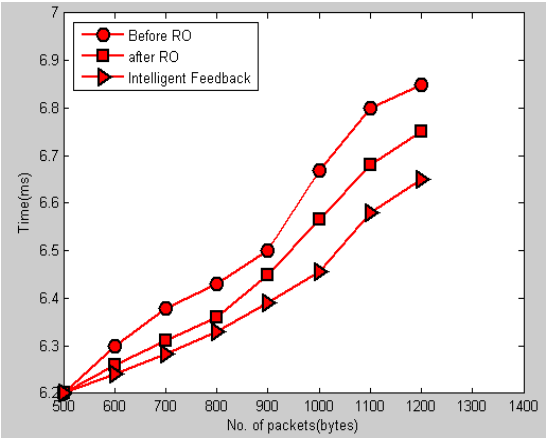

Figure 6: Performance Rate

We have implemented our framework to measure the optimality of several algorithms: (1) the optimal partition-based algorithm. (2) The heuristic ACL splitting algorithm for space. (3) rule-based algorithm. We have implemented each algorithm, using shared code for generating traffic profiles, calculating costs, and generating integer programs. We have used an industrial-strength integer program solver, CPLEX, to solve our

integer programs. For integer programs, the solver uses a branch and bounding algorithm with several heuristics to find optimal solutions. The application of Rule Reordering techniques with intelligent feedback system increases the throughput and performance of the Firewall. This intelligent feedback system is based on log analyzer and expert opinion. This log analyzer is used for find out all packet information by IPFW tool. Expert opinion just like Firewall Policy Advisor so it knows 95% accurate result and it give opinion to feedback system .We ran our experiments on a system with a Xeon 2.8Ghz processor, 2GB of RAM, and Linux kernel/windows and designing parts are done using Visual Basic 6.0.

In this figure 6 the upper curve shows the performance of the firewall before the reordering rule was applied. The next curve indicates that the performance is good after reordering rule was applied because the time decreases. After applying the Intelligent Feedback system the overall performance increases and the transmission of packets was also efficient as indicated by the last one. For enhancing the rule based optimization, intelligent feedback mechanism is used so that rules are optimized periodically to achieve efficient performance..

5. Conclusion:

Diverse Firewall Design technique is proposed by comparing two given firewall which compute the functional discrepancy, this result in hybrid rule set. We have presented a general framework for evaluating optimization techniques for rule-based firewalls. Here the input packets are reordered by rule reordering techniques. Rule reordering is enhanced by intelligent feedback system. Intelligent feedback system support periodically updating the rules, then improve the efficiency of firewall in host. This will lead to improved performance in the rule based reordering of firewalls. Thus the intelligent rule reordering will maintain the performance and throughput of firewall.

6. References:

1. Ghassan Misherghi, Lihua Yuan, Zhendong Su, Chen-Nee Chuah, and Hao Chen (2008). “A General Framework for Benchmarking Firewall Optimization Techniques,” IEEE Transactions on network and service management, vol. 5, no. 4, December 2008.
2. Richard A. Deal (2003), Firewalls-The Ultimate Reference, Dream Tech Press.
3. Keith E. Stassberg, Richard J.Gondek, Garry Rollie, Firewalls-The Complete Reference.
4. Charlie Kaufman, Radia Perlman, Mike Speciner, Network Security, Prentice Hall of India.
5. Bob Hughes and Mike Cotterell, Software Project Management, Third Edition, Tata McGraw Hill.
6. [online] http://en.wikipedia.org/wiki/Firewall_(networking)
7. [online] http://www.vicomsoft.com/knowledge/reference/firewalls1.html
8. [online] http://www.pc-help.org/www.nwinternet.com/pchelp/security/firewalls.htm
9. [online] http://www.roseindia.net/java
10. [online] http://www.java2s.com/code/java
11. [online]http://netresearch.ics.uci.edu/kfujii/jpcap/doc/javadoc/
12. [online] http://en.wikipedia.org/wiki/Log_analysis
13. S. Acharya, J.Wang, Z. Ge, T. F. Zane, and A. Greenberg,(2006), “Traffic-aware firewall optimization strategies,” in Proc. International Conference on Communications
14. H. Hamed and E. Al-Shaer,(2006) “Dynamic rule-ordering optimization for high-speed firewall filtering,” in Proc. ACM Symposium on Information, Computer and Communications Security, pp. 332–342.
15. P. Gupta and N. McKeown,(2001) “Algorithms for packet classification,” IEEE Network, Mar. 2001..
16. E. Cohen and C. Lund,(2005) “Packet classification in large ISPs: design and evaluation of Decision tree classifiers,” in Proc. ACM SIGMETRICS, pp. 73–84.
17. F.Baboescu and G. Varghese,(2005) “Scalable packet classification,”IEEE/ACM Trans. Networking, vol. 13, no. 1, pp. 2–14.
18. E. Al-Shaer and H. Hamed,(2004) “Discovery of Policy Anomalies in Distributed Firewalls,” Proc. IEEE INFOCOM ’04, pp. 2605-2616, Mar. 2004.
19. M.G. Gouda and A.X. Liu,(2004) “Firewall Design: Consistency, Completeness and Compactness,” Proc. 24th IEEE Int’l Conf. Distributed Computing Systems (ICDCS ’04), pp. 320-327, Mar. 2004.
20. H. Hamed, E. Al-Shaer, and W. Marrero, (2005) “Modeling and Verification of IPsec and VPN Security Policies,” Proc. 13th IEEE Int’l Conf. Network Protocols (ICNP ’05), pp. 259-278,Nov. 2005.
21. M.G. Gouda and A.X. Liu, (2007) “Structured Firewall Design,” Computer Networks J., vol. 51, no. 4, pp. 1106-1120, Mar. 2007.
22. S. Kamara, S. Fahmy, E. Schultz, F. Kerschbaum, and M. Frantzen, (2003) “Analysis of Vulnerabilities in Internet Firewalls,” Computers and Security, vol. 22, no. 3, pp. 214-232.
23. P. Gupta and N. McKeown, (2001) “Algorithms for Packet Classification,” IEEE Network, vol. 15, no. 2, pp. 24-32.
24. H. Hamed, E. Al-Shaer, and W. Marrero, (2005) “Modeling and Verification of IPsec and VPN Security Policies,” Proc. 13th IEEE Int’l Conf. Network Protocols (ICNP ’05), pp. 259-278,Nov. 2005.