A Web-based Firewall Simulator Tool for Information Security Education

Zouheir Trabelsi and Umniya Mustafa
College of Information Technology
UAE University
PO Box 17551, Al Ain, UAE
{trabelsi, umniya.M}@uaeu.ac.ae

Abstract
Teaching practical information security requires the use of techniques, security and network devices and software, simulator tools, testbed networks, and hands-on lab exercises to support the educational process. This paper presents an educational web-based firewall simulator tool to help students learn the intricacies of writing firewall filtering rules to filter and inspect network traffic. The design principle of the simulator tool is to be easy to use while teaching the students the details of writing basic and advanced filtering rules. The simulator tool offers a set of educational functions that are not commonly available in professional firewalls. The tool can be used in any network and security course by instructors and students in the classroom. The impact of offering the simulator tool on the students’ performance in terms of achieving the course outcomes is also discussed.

Keywords: Information security education, Firewall, Packet filtering, Filtering rules, Simulator tools.

1 Introduction
Security has become a major concern throughout the world. Increasing information security education and teaching knowledge of security techniques to students who major in computer science and information technology, and satisfying new expectations for information technology professionals has become a very urgent need (Conklin 2006 and Al-Shaer 2009).

Moreover, the need to use a practice and application oriented approach in information security education is paramount. A security education curriculum that does not give the students the opportunity to experiment in practice with security techniques cannot prepare them to be able to protect efficiently the confidentiality, integrity, and availability of computer systems and assets. In addition, teaching practical information security requires the use of techniques, security and network devices and software, simulator tools, testbed networks and hands-on lab exercises to support the educational process.

On the other hand, since firewalls are an import security topic, information security programs should include courses that cover firewall concepts and technologies. According to students’ feedback (Williams 2011), after traditional lectures on firewall concepts some of them have difficulty fully understanding the use and configuration of a firewall. For many students an interactive education tool can help them to understand the functions of firewalls by getting hands-on and configuring a firewall step by step.

Therefore, to enhance the student’s practical skills on firewalls, schools offering information security programs need to acquire professional firewalls in order to offer to the students hands-on lab exercises on firewalls. However, professional firewalls are designed mainly for professional uses, and are not adequate for the academia environment. That is, most of the offered functions in professional firewalls are not needed for information security education and do not allow students to better anatomize the firewall concepts and technologies taught in the lecture. For example, most professional firewalls do not allow the user to directly manipulate and freely set the values of the fields in a filtering rule. In addition, firewalls are usually expensive hardware devices; however academic institutions have limited budgets to setup security lab facilities.

In order to enhance information security education, firewall simulator tools can be a very effective solution, and can help students better understand the functions of a firewall and filtering rule implementation. Also, firewall simulator tools let students get hands-on experience and are ways of learning the practical aspects of firewalls. In addition, compared to professional firewalls, firewall simulator tools are an affordable solution for schools offering information security programs with limited budget for setting up security lab facilities.

This paper discusses the design and implementation of a web-based firewall simulator tool, called Edu-Firewall, that is dedicated for educational purpose. Edu-Firewall tool offers basic firewall functions as well as advanced educational security functions that are not available in current professional firewalls. Moreover, compared to professional firewalls, Edu-Firewall tool allows students to better anatomize the advanced concepts of writing effective and efficient firewall filtering rules for standard and nonstandard services. In contrast to many professional firewalls, the tool offers a graphical user-friendly interface that can be used to create advanced and low level filtering rules, and to freely set the values of the fields of filtering rules.

The paper is organized as follows: Sections 2 and 3 discuss related simulator tools, and limitations of professional firewalls, respectively. Section 4 discusses the design considerations of the proposed firewall simulator tool. Sections 5 and 6 describe the educational...
functions and the implementation of the tool, respectively. Section 7 discusses the effect of offering the tool on the student performance, and the evaluation of the learning outcomes. Section 8 discusses the students’ satisfaction. Finally, Section 9 concludes the paper.

2 Related Simulator Tools

Firewalls are security devices or software tools used to apply an organization’s security policy by implementing filtering rules to filter the incoming and outgoing network traffic. A number of educational firewall simulation systems have been previously developed. The goal of some systems (Garrido 2009 and Ye 2006) is to provide the student with a statistically supported understanding of a firewall’s effectiveness. The configuration of the firewall in these systems is fixed by the simulation and not defined by the student. These systems give the student an understanding of network and firewall load, but provide very little training on the configuration of a firewall. Several systems (Wang 2010, Hu 2004, and Stewart 2009) make use of virtualization for security simulations. Virtualization is effective for providing each student virtual resources, including firewalls, that the student can deploy to secure a network.

CyberCIEGE (Irvine 2005) is an interactive game designed to teach computer and network security concepts. It provides the student with a wide range of security threats and possible solutions. Firewall configuration within CyberCIEGE is done at a high level without defining the details of what specific packets are to be filtered. Students are required to configure a simulated firewall using Cisco-like commands to prevent other students from attacking them.

The objective of the interactive firewall simulator presented in (Williams 2011) is to provide students with an interactive competitive system to help them better understand the concepts of firewall configuration and operation. However, this tool provides limited number of educational functions to optimize the filtering rule set, such as the verification of the consistency and efficiency of filtering rules. Also, the tool does not allow students to implement low level filtering rules, and better anatomize common advanced firewall concepts, such as stateful packet inspection, nonstandard service filtering, and packet content inspection (known also as Deep Packet Inspection (DPI)).

All the above related simulator tools have a common limitation regarding the covered firewall concepts. That is, they offer students basic firewall functions for firewall configuration and simple filtering rules implementation. However, students enrolled in an information security program require to anatomize basic as well as advanced firewall concepts and writing low level filtering rules.

3 Limitation of Professional Firewalls

This section describes a comparison study that we conducted to evaluate the available educational functions offered by four common professional firewalls (Cisco ASA 5520, Juniper Networks, Preventia and Jetico Personal Firewall). The main objectives of this study are to identify the ability of the firewall interface to allow its users to:

1. Implement low level and advanced filtering rules for packet filtering and content inspection.
2. Visualize the status of the network stateful sessions.
3. Identify and correct inconsistent and inefficient filtering rules.
4. Assign a wide range of value types to the filtering rule’s fields.

Table 1 shows the results of the study and the limitations of the evaluated professional firewalls. In fact, when they are used by students for hands-on lab exercises, the evaluated firewalls do not allow students to (1) freely set the values of the fields in the filtering rules, (2) write low level and advanced filtering rules and (3) verify the consistency and efficiency of filtering rules. Consequently, the evaluated firewalls do not allow students to better anatomize the advanced firewall concepts. This study confirms the claim that professional firewalls are designed mainly for professional uses, and are moderately adequate for the academia environment.

In addition, small multi-port wireless routers are for general purpose use and are not designed for educational purpose. They have the same educational limitations as in the case of professional firewalls.

On the other hand, compared to traditional firewall approaches such as IPtables and IPfw, the simulator tool approach is more adequate for developing educational lab exercises and allows providing the students with more advanced educational firewall functions, such as writing low level filtering rule, packet content inspection, and consistency and efficiency verification of filtering rules. In addition, the simulator tool approach provides more friendly interaction between educators and students.

4 Design Considerations

Firewalls control the access into and from networks based on a set of filtering rules which reflect and enforce the organization’s security policy. Within a network, the firewall is typically the first filtering device that encounters packets that attempt to enter an organization’s network from the outside, and is typically the last device to see exiting packets. It is the firewall’s job to make filtering decision on every packet that crosses it: either to let it pass, or to drop it.

There are a number of basic topics about network packet filtering and inspection that should be taught when offering a security course on firewalls:

- Basic packet filtering
- ICMP traffic filtering
- Common standard services (HTTP, FTP, SMTP, POP3, Telnet, and DNS) filtering
- Nonstandard services filtering
- Packet content inspection
- Stateless and stateful firewalls
- Consistency and efficiency verification of filtering rules
- Filtering rules order management

When designing and implementing an educational firewall simulator tool, the above basic topics should be taken into consideration. The simulator tool is expected to
offer advanced functions that are not usually available in common professional firewalls. In fact, the simulator tool should provide its users with the adequate functions that allow implementing basic and advanced packet filtering rules for both standard and nonstandard network traffic and services, using preferably graphical user-friendly interfaces. In addition, the simulator tool’s users should be able to verify the consistency and efficiency of their defined filtering rules as well as viewing the contents of the stateful tables of the established network sessions. Moreover, the simulator tool should allow its users to easily modify the order of the filtering rules to adequately reflect the security policy under consideration.

Other topics, such as application gateway firewalls (Proxy), Virtual Private Networks (VPN), firewall secure network architectures, are also usually covered by a security course on firewalls. However, the current version of the firewall simulator tool does not cover them. A future version of the tool can include more educational functions to cover these topics.

Based on the above design considerations, the educational objectives of Edu-Firewall tool are outlined as follows:

- Provide an educational solution that helps students to improve their hands-on security skills on firewalls.
- Offer students the adequate means to deal with advanced firewalls security features.
- Provide students with a user-friendly simulated environment for firewall configuration, and for writing low level packet filtering rules.
- Provide educators with means to evaluate students’ hands-on security skills on firewalls.

5 Educational Functions of Edu-Firewall Tool

Edu-Firewall tool can be used by security educators as well as by students to write basic and advanced filtering rules for given security policies, and to optimize the filtering rule set. Edu-Firewall tool allows students to select security policies from the available list which has been created by the educators, and then write the corresponding filtering rules. On the other hand, Edu-Firewall tool allows the educators to evaluate the students’ work. Depending on the complexity of the selected security policy, the corresponding filtering rules can be simple or complex to implement. Edu-Firewall tool uses web interfaces that include a set of functions to allow the students implementing both simple and advanced filtering rules. Edu-Firewall tool’s educational functions cover the main topics taught in a security course on firewall concepts. The offered educational functions of Edu-Firewall tool allow performing mainly the following tasks:

- Define advanced TCP filtering rules by freely setting all TCP header fields, including the TCP flags.
- Define advanced ICMP filtering rules by freely setting all ICMP header fields.
- Define advanced UDP filtering rules by freely setting all UDP header fields.
- View the contents of the TCP, ICMP, and UDP stateful sessions table.
- Verify the consistency and efficiency of the filtering rules.
- Optimize the order of the filtering rules.
- View the logs of filtered packets.

The following subsections describe briefly the main Edu-Firewall tool’s educational functions.

5.1 Basic Packet Filtering

Basic firewall packet filtering is the selective passing or blocking of packets as they pass through a network interface. The most often used criteria that packet filtering uses when inspecting packets are source and destination IP addresses, source and destination TCP/UDP ports, TCP flag bits, type and code fields in an ICMP header, and the protocol field of the Layer 4 header. For example, to filter out all Ping traffic coming to a network, the firewall simulator tool should allow implementing filtering rules that block all incoming ICMP echo request packets (Type=8 and Code=0). The following filtering rule reflects the above security policy: (Direction = Incoming, Source IP = Any, Destination IP = Any, Protocol = ICMP, Type = 8, Code = 0, Action = Deny).

5.2 Packet Content Inspection

Packet content inspection is a form of network packet filtering that examines the data part (Payload data), and possibly also the headers of a packet as it passes a firewall. Usually, packet content inspection searches for protocol non-compliance, viruses, spam, intrusions or predefined criteria to decide if the packet can pass or if it needs to be routed to a different destination. Usually, packet content inspection process uses a set of signatures, commonly created by the firewall administrator. For example, you might want to prevent your network’s users from receiving emails from a specific email address.

5.3 Nonstandard services filtering

Standard services run usually on standard ports. For example, the standard ports for HTTP and FTP services are 80 and 21, respectively. Firewalls include usually predefined rules to filter standard services, and are unable to filter nonstandard services unless the user provides the firewall with the TCP or UDP ports of the nonstandard services. In practice, this is achieved by creating a new service profile for the nonstandard service and specifying its corresponding TCP or UDP port number.

5.4 Consistency and Efficiency Verification of Firewall Filtering Rules

The consistency and efficiency of a firewall strongly depend on the ability of the administrator to develop well defined and coherent filtering rule set, and to be able to continuously clean and verify the correctness of the rules. It is important to mention that in cases where there are dozens or hundreds of filtering rules, inconsistent and inefficient filtering rules with anomalies might not be easy to spot. Hence, checking a large set of filtering rules for inconsistencies and inefficiencies is difficult and prone to errors when it is done manually and in an ad-hoc manner. Thus, automated tools are required to assume such a task. However, to our knowledge, currently there is no professional firewall that integrates strong filtering rule consistency and efficiency verification capabilities.
For example, Juniper firewalls (Table 1) provide a very limited online command to identify inconsistent and/or inefficient filtering rules.

Edu-Firewall tool allows verifying the following three types of inconsistency and inefficiency within the filtering rules.

**Shadowing:** A rule is shadowed by a preceding rule if it is a subset of the preceding rule; and the two rules define different actions. That is, an upper rule shadows a lower rule, when the upper rule matches all the packets that are also matched by the lower rule, such that the lower rule will never be reached by the firewall. To correct this situation, simply remove one of the filtering rules, or reverse the order of the two filtering rules, putting the more specific one (shadowed rule) first. The following two rules are example of shadowing rules:

**Rule #1:**
- **Direction** = Incoming, **Source IP** = Any, **Destination IP** = Any, **Protocol** = TCP, **Source Port** = Any, **Destination Port** = 80, **Action** = Allow.

**Rule #2:**
- **Direction** = Incoming, **Source IP** = 192.168.3.30, **Destination IP** = Any, **Protocol** = TCP, **Source Port** = Any, **Destination Port** = 80, **Action** = Deny.

**Redundancy:** A redundant rule performs the same action on the same packets as another rule such that the removal of it would not affect the operation of the firewall.

Contradictory: two rules have the same parameters except the action is opposite.

## Edu-Firewall Tool’s Implementation

Edu-Firewall tool is a web application, and has been implemented using Java language as a programming language; specifically, JSP (JavaServer Pages) using NetBeans IDE software as a development environment. In addition, mySQL has been used to create the Edu-Firewall tool’s database of security policies and filtering rules.

### 6.1 Edu-Firewall Tool’s Web Interfaces

Edu-Firewall tool offers two main web interfaces. The first web interface is dedicated for the instructors to create and update security policies (Figure 1).

The second web interface is for students to use the educational firewall functions offered by Edu-Firewall tool. That is, after logging to the tool, the student will be redirected to a web page from which he/she can select a security policy scenario. Then, the student is asked to enter the adequate filtering rules for the selected security policy. This requires creating, editing and ordering the filtering rules. The student will also be able to verify the consistency and efficiency of the entered filtering rules. All the added filtering rules will be saved in the database and then will be evaluated by the instructors. Before defining the filtering rules, the student should configure the network interfaces through the Interface configuration web page.

**Network interface configuration:**

The Interface configuration web page is designed to set the Ethernet network interfaces. Students can configure up to two network interfaces. This page allows the student to specify the network IP address, the interface IP address, the network mask address, and the Dynamic Host Configuration Protocol (DHCP) IP address range (Figure 2).

### Filtering rule definition:

This web page allows students to write basic and advanced filtering rules for standard and nonstandard services, and deep packet content inspection. Depending on the selected security policy, the student is requested to enter the appropriate filtering rules by specifying the values of the packet’s fields that should be inspected by the firewall.

For each filtering rule, the simulator tool shows also the direction of the network traffic (incoming or outgoing) that will be filtered by the firewall. Figure 3 shows the list of the TCP packet’s fields that can be set when writing a filtering rule.

**Packet deep inspection:**

Edu-Firewall tool’s Deep Inspection web interface is designed to provide the application layer inspection for the most prevalent Internet-facing protocols, such as HTTP, FTP, SMTP and POP3. In this web interface, the student will specify the attack name, attack context and attack pattern for packet deep inspection (Figure 4).

### Inconsistency and inefficiency verification:

Edu-Firewall tool’s Rule verification web page allows students to verify the consistency and efficiency of the created filtering rules. Practically, this function allows detecting shadow, contradictory and redundancy filtering rules (Figure 5).

### 7 Evaluation of Learning Outcomes

Several hands-on lab exercises on firewall have been offered in our Network Border Control course (SECB358) during the last three academic years. SECB358 course teaches students mainly firewall concepts and technologies, and packet filtering and inspection techniques. The exercises require the use of Edu-Firewall tool by the students.

This section discusses the effect of introducing Edu-Firewall tool during the hands-on lab exercises on the achievement of the SECB358 course’s outcomes (COs). SECB358 course has five COs as shown in Table 2. Since SECB358 course is an advanced course in information security, the outcomes have been selected carefully to reflect the top three levels in the bloom’s taxonomy of cognitive domain (analysis, synthesis, and evaluation). After creating the course outcomes, 10 course topics were identified and mapped to the course outcomes. Four assessment tools are also selected to assess the achievements of COs including quizzes, exams (midterm and final), lab reports, and term project.
Table 2: Mapping the course outcomes to Bloom’s Taxonomy

<table>
<thead>
<tr>
<th>Outcome</th>
<th>Level of Bloom’s Taxonomy</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1: Describe TCP/IP protocols and network services.</td>
<td>Analysis (4)</td>
</tr>
<tr>
<td>CO2: Identify common security threats.</td>
<td>Analysis (4)</td>
</tr>
<tr>
<td>CO3: Configure personal firewalls, network firewalls, and VPNs.</td>
<td>Synthesis (5)</td>
</tr>
<tr>
<td>CO 4: Implement firewall filtering rules for different network architectures and services.</td>
<td>Synthesis (5)</td>
</tr>
<tr>
<td>CO 5: Evaluate different types of network architectures.</td>
<td>Evaluation (6)</td>
</tr>
</tbody>
</table>

To assess the course outcomes, we follow the course assessment process adopted by our institution. A nominated course coordinator assembles a course committee that includes all the lecture and lab instructors teaching the course in a given semester. During the first week of the semester, the course committee meets to decide on the assessment tools that will be used to assess the COs. They also decide on the corrective actions that will be applied to address the recommendations from the previous assessment cycle. Throughout the semester, the course committee applies the assessment tools to collect assessment data. By the end of the semester, the collected assessment data are mapped to the COs. The achievement level of each CO is then calculated in terms of mean and standard deviation using (1) and (2).

\[
\mu (CO_i) = \frac{\sum t \mu_i \times n_i}{\sum n_i} \quad (1)
\]

\[
\sigma (CO_i) = \sqrt{\frac{\sum \sigma_i^2 \times n_i}{\sum n_i}} \quad (2)
\]

where \( \mu_i \) and \( \sigma_i \) denote respectively the normalized mean, and standard deviation of the students’ marks when assessment tool \( t \) is used, and \( n_i \) denotes the number of students. For example, if three quizzes and two final exam questions are used to assess COs, the normalized mean and standard deviation of the students’ marks are calculated separately for each tool, then (1) and (2) are used to calculate the achievement level for CO. After calculating the achievement level for each CO, the course committee meets again to discuss the assessment results and decide on the needed recommendations to address any discovered shortcoming. To close the assessment cycle, the course committee also discusses the effectiveness of the corrective actions applied during the semesters on the new assessment results.

During the 2008/2009 and 2009/2010 academic years, students enrolled in SECB358 course were not offered Edu-Firewall tool during the hands-on lab exercises. Only the firewall theoretical concepts were described during the lecture time. However, starting from fall 2010 the course committee decided to offer Edu-Firewall tool during the hands-on lab exercises as a corrective action to improve the COs achievement levels and improve the students’ hands-on skills on firewalls. Three quizzes are used to compare the achievement of the COs before and after the introduction of Edu-Firewall tool. These quizzes are directly mapped to CO3 and CO4 outcomes. The grades of the students in the three quizzes are measured, normalized, and then aggregated using (1) and (2) to calculate the achievement level of the two outcomes.

7.1 Assessment Results

Figure 6 shows the students’ average grades for the three quizzes used to evaluate the students’ comprehension of firewall configuration and filtering rule implementation. It shows clearly that starting from 10/11 academic year, the total average grade has started improving. This is mainly due to the fact that the offered Edu-Firewall tool allowed students to better analyze and assimilate the firewall concepts learned from the lectures. The students have learned better with Edu-Firewall tool which had a positive effect on their performance. For example, in case of Quiz 2, introducing Edu-Firewall tool improved the average student grade by 9% from 0.68 to 0.74 and maintained the improvement for the following two academic years.

Figure 7 illustrates the achievement of the two course outcomes (CO3 and CO4) for five consecutive academic years from 08/09 to 12/13. It shows an improvement in the two outcomes achievements level after introducing Edu-Firewall tool. For example, the introduction of Edu-Firewall tool in 10/11 academic year improved the achievements levels by 3% and 9% for CO3 and CO4 respectively compared to the achievement levels in the year before. It is important to indicate that the introduction of Edu-Firewall tool improved slightly the achievement level of the outcome CO3 since the VPN concept is not covered by the tool. In contrast, the achievement level of the outcome CO4 has been improved well since the tool offers a set of educational functions covering all the topics related to this outcome, specifically implementing basic and advanced firewall filtering rules.

![Figure 6: Average student performance in three quizzes before and after introducing Edu-Firewall tool](image-url)
8 Student’s Satisfaction
An anonymous questionnaire was administered to 120 students who used Edu-Firewall tool, to measure their satisfaction level and collect their feedback regarding the tool. The results of the questionnaire showed that about 93% of the students who answered the questionnaire believed Edu-Firewall tool to be useful and helped them better understand the underlying theoretical concepts associated with firewalls and packet filtering (Table 3). The questionnaire also revealed that about 80% of the students agreed that the tool helped them to develop further their hands-on skills, and about 91% of the students would strongly see similar simulator tools offered for other security topics.

<table>
<thead>
<tr>
<th>Question</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do you think it is easy to create and manage filtering rules using Edu-Firewall tool?</td>
<td>5% Disagree, 10% Neutral, 23% Agree, 62% Strongly agree</td>
</tr>
<tr>
<td>Do you think Edu-Firewall tool offers to you the expected basic firewall features?</td>
<td>3% Disagree, 8% Neutral, 10% Agree, 79% Strongly agree</td>
</tr>
<tr>
<td>Do you feel that Edu-Firewall tool helped you to develop further your hands-on skills?</td>
<td>7% Disagree, 13% Neutral, 13% Agree, 67% Strongly agree</td>
</tr>
<tr>
<td>Do you feel you understand the firewall theoretical concepts better after using Edu-Firewall tool?</td>
<td>2% Disagree, 5% Neutral, 8% Agree, 85% Strongly agree</td>
</tr>
<tr>
<td>Do you think you will recommend Edu-Firewall tool to others?</td>
<td>5% Disagree, 5% Neutral, 3% Agree, 87% Strongly agree</td>
</tr>
<tr>
<td>Would you like to see similar simulator tools offered for other security topics, such as intrusion detection?</td>
<td>5% Disagree, 4% Neutral, 18% Agree, 73% Strongly agree</td>
</tr>
</tbody>
</table>

Table 3: Student satisfaction questionnaire results

9 Conclusion
It is necessary that students enrolled in information security programs acquire solid hands-on skills on firewalls configuration and filtering rules. This paper described an educational web-based firewall simulator tool to help students learn the intricacies of writing firewall filtering rules to filter and inspect network traffic. The simulator tool offers a set of educational functions that are not commonly available in professional firewalls. The tool is designed to be used as a part of an undergraduate or graduate level course on firewalls.

As a case study, the impact of using the simulator tool on the students’ performance in terms of achieving the course outcomes is also discussed. The assessment results showed a significant improvement in the achievement level of related course outcomes. Overall, the students have learned better with the tool which had a positive effect on their performance.

Future versions of the tool will include several uncovered educational firewall functions, such as VPN network configuration. In addition, to effectively evaluate the students’ defined filtering rules, the network traffic that will be filtered and inspected by the simulator tool will be captured directly from the Internet or from LAN networks.

10 Acknowledgement
The authors acknowledge the contribution of the following students in the implementation of the web interfaces of Edu-Firewall tool: Alia Sultan AlNuaimi, Khawla Salem Al-Mazrooe, Latifa Rashed Al-Ketbi, and Shamsa Rashed Al-Neyadi.

11 References
<table>
<thead>
<tr>
<th>Firewall</th>
<th>Juniper Networks Version: 6.2.0r5.0 (firewall + VPN)</th>
<th>Jetico Personal Firewall v.2.1.0.10</th>
<th>Cisco ASA 5520 Version: 7.0 (7)</th>
<th>Proventia Version: 1.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create new services</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Rules parameters</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>IP address</strong></td>
<td>Multiple Yes</td>
<td>Multiple Yes</td>
<td>Multiple Yes</td>
<td>Multiple No</td>
</tr>
<tr>
<td></td>
<td>Single Yes</td>
<td>Single Yes</td>
<td>Single Yes</td>
<td>Single Yes</td>
</tr>
<tr>
<td></td>
<td>Range Yes</td>
<td>Range Yes</td>
<td>Range Yes</td>
<td>Range Yes</td>
</tr>
<tr>
<td></td>
<td>Discrete No</td>
<td>Discrete No</td>
<td>Discrete No</td>
<td>Discrete No</td>
</tr>
<tr>
<td></td>
<td>Network Yes</td>
<td>Network Yes</td>
<td>Network Yes</td>
<td>Network Yes</td>
</tr>
<tr>
<td><strong>Port number</strong></td>
<td>Standard/Default</td>
<td>Single port Port range</td>
<td>Standard/Default</td>
<td>Single port Port range</td>
</tr>
<tr>
<td><strong>Actions</strong></td>
<td>Allow/Deny/Reject</td>
<td>Accept/Reject/ Continue</td>
<td>Permit/Deny</td>
<td>Drop/Protect/Ignore/Monitor</td>
</tr>
<tr>
<td><strong>Protocols</strong></td>
<td>Wide range of supported protocols</td>
<td>Wide range of supported protocols</td>
<td>TCP/UDP/ICMP/IP</td>
<td>Wide range of supported protocols</td>
</tr>
<tr>
<td><strong>Deep inspection</strong></td>
<td>Supported</td>
<td>Supported</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td><strong>Writing low level rules</strong></td>
<td>Not supported</td>
<td>Partially supported</td>
<td>Not supported</td>
<td>Partially supported</td>
</tr>
<tr>
<td><strong>Default rules</strong></td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td><strong>Consistency and efficiency verification</strong></td>
<td>Partially supported</td>
<td>Not supported</td>
<td>Not supported</td>
<td>Not supported</td>
</tr>
<tr>
<td><strong>Setting TCP flags</strong></td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td><strong>Stateful/Stateless</strong></td>
<td>Stateless</td>
<td>Stateful</td>
<td>Stateful</td>
<td>Stateful</td>
</tr>
</tbody>
</table>

Table 1: Limitation of professional firewalls

Figure 1: Web interface for creating and updating security policies

Figure 2: Web interface for network interface configuration
Figure 3: Web interface for filtering rule definition

Figure 4: Web interface for packet deep inspection

Figure 5: Web interface for filtering rule consistency and efficiency verification