Repowering an Open Source Firewall Based on a Quantitative Evaluation

Walter Fuertes , Patricio Zambrano, Marco Sánchez , Mónica Santillán, César Villacís, Theofilos Toulkeridis, Edgar Torres

Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador

Summary
The increased cyber-attacks in recent years have included violation of firewalls. Based on these facts, our study and main objective is to present the formation of a platform for open source firewall, which induces a highly efficient method to strengthen detection, control and also mitigation of such assaults. In order to fulfill these aims, we designed and implemented an experimental LAN / WAN network environment separated by a firewall device configured in three different software implementations: PfSense, Zentyal and ClearOS. Subsequently, in order to assess quantitatively the performance and efficiency of such systems against cyber-attacks, a firewall was repowered integrating and consolidating an inspection, unifying threat management solutions. The outcome of this research demonstrates clearly and objectively the efficiency and importance of our study.

Key words:
Cyber-attacks, quantitative evaluation, Firewall open source, repowering solutions.

1. Introduction

Information systems of a variety of institutions permanently suffer cyber-attacks, despite of multiple efforts to implement and acquire detection and control mechanisms as firewalls. Such assaults have increased in recent years, having as main targets organizations, institutions, corporations and government entities worldwide [1]. Based on these above-described circumstances, programmers and researchers are doing great efforts to detect early and mitigate efficiently such modern day digital attacks.

In relation to several existing detection mechanisms the technically more robust mechanism is called firewall. This device appears to be the main defense of border security for networks against attacks and unauthorized traffic. However, the effectiveness of these mechanism is dependent on policy management techniques that network administrators are able to use, in order to analyze, debug and verify the correctness of the available filter rules [2].

Currently, due to the evolution and sophistication of the abovementioned attacks, industry has designed numerous open source software tools [3] by incorporating multiple security levels of protection [4]. Industry has also developed several techniques to improve detection and control attacks that have upon firewalls [5][6][7][8], and it has even strengthened firewall decisions by the use of machine learning techniques such as fingerprints, to infer the solution [9]. In this context, the scientific community has performed comparative studies about firewalls effectiveness [10][11][12][13]. Nonetheless, despite of these efforts, firewall vulnerabilities have increased as demonstrated by National Vulnerability Database statistics. The data indicate, as one of many examples of such nature, that in the third quarter of 2014, the number of reported vulnerabilities has been approximately 66.611, being much higher compared to previous years where the numbers did not reach 50,000 (2011) [14]. In addition, the number of new reported security vulnerabilities in 2013 (4794) continued to increase when compared to 2012 (4347), and to 2011 (3532). “On average, 13 new vulnerabilities per day were reported in 2013, for a total of 4,794 security vulnerabilities: the highest number in the last five years” [14].

Taking in consideration all these facts and associated problems, this research study proposes to improve the evaluation and repowering of an open source firewall platform. This evaluation has been based on the distinctive characteristics of a private system and additionally verified their behavior in terms of performance and effectiveness against cyber-attacks, in order to strengthen the detection, control and mitigate these assaults at companies. To fulfill this aim, we designed and implemented an experimental LAN / WAN network environment separated by a firewall device, which has been configured using three different software implementations such as Pfsense [15], Zentyal [16], and ClearOS [17], alternating in three different distributions such as Linux Redhat, Debian and FreeBSD, respectively. To prove and evidence non-biased results with same applied conditions we performed all implementations in identical real environment that allowed a variety of tests to the selected firewalls.

There are three main innovative contributions of our study, such as: a) a quantitative evaluation of three open source security firewalls; b) creation of real testing environment on a single machine, by means of simulating the application for a hundred Web pages, solving the limitation of the number of required physical machines, and, finally c) repowering of an open source firewall by

Manuscript received November 5, 2014
Manuscript revised November 20, 2014

changing its GUI, reprogramming the source code, and adding libraries to provide a comprehensive solution and consolidated various security services. Additionally, we described in Section 2 the theoretical framework and in Section 3 the design, implementation and execution of the experimental topology, realized tests; developed algorithms and explained how firewall repowering is done. Lastly, the experimental results are provided in Section 4, while in Section 5 we discuss some related work, while in Section 6 we present our conclusions and also some recommendations for future studies.

2. Open Source vs Private Firewalls

According to the RFC 2979 recommendation by the IETF in [18], the performance characteristics and interoperability requirements are defined for Internet security firewalls. These requirements are proposed for consistent behavior across firewalls-platforms. This recommendation is conceptualized to a "firewall as an agent that filters network traffic somehow blocking it considers inappropriate, dangerous, or both" [18].

Firewalls are hardware devices or software systems that control the traffic-flow between two or more networks using certain security policies [19]. The functionality of firewalls is limited to compliance with a set of rules, in order to protect the internal network from any kind of threats. This technology has evolved so it is able to be integrated with other systems generating new concepts such as packet inspection (Deep Packet Inspection, DPI), which incorporates Intrusion Detection Systems (IDS) and Intrusion Protection Systems (IPS) being capable to analyze associated traffic. The Next Generation Firewalls (NGFW), are those where policies can be defined based on combining applications and Network Address Translation (NAT), IPS and Virtual Private Network (VPN). Furthermore, Unified Threat Management (UTM) consolidates multiple security services on a single machine, such as VPN, IDS, IPS, Antivirus, Anti-Spam, Web Proxy, NAT, as well as content filtering [20].

There are different ways to protect a network using security firewalls: i) Access Control Lists (ACLs), which is the simplest but most insecure form of topology, where the router performs routing and packet filtering; ii) Simple Demilitarized Zone, which is defined as a public area, in which both public servers and the internal network are protected by a firewall; iii) dual Demilitarized Zone, which adds a firewall to control and protect the internal network gateway, being the most fundamental reason to protect public servers against attacks from the internal network; iv) Depth on defense, which is a concept that is implemented using security layers to maximize protection. This is demonstrated, on the edge of the perimeter firewall gateway that performs packet filtering and protect the DMZ and LAN of traffic from the Internet; v) Firewall appliance-based hardware, which is designed and optimized for exclusive firewall works, and hence especially packet filtering machines; vi) software based firewalls, being a less economic alternative compared to hardware-based firewall, but present greater challenges in their configuration and deployment, such as exposed with Linux IPtables.

The criteria for choosing a firewall, according to [20][21][22] include throughput, budget, amount of networks to protect, level of depth filter and capacity scalability: (i) Throughput expected, it is a necessary measure in which one may lose in performance but gains in security; (ii) Budget, as there are various free and commercial alternatives, within one must evaluate advantages and disadvantages in terms of quality, support, benefits and costs; (iii) Number of networks to protect, as the size of the network defines the capabilities of the hardware. Also, it should be considered the estimated traffic that will handle the firewall; (iv) Level of depth filter, which depends on the resources to be protected you where one needs to decide which layer requests to check the packages provided; (v) Capacity scalability of the functions of firewall, which is the possibility of growth.

The above listed arguments have been implemented by the industry in various commercial hardware products (Cisco, Checkpoint, Fortinent, Watchward, Nokia, Juniper etc.) and open source software or proprietary (Zentyal, ClearOS, Untangle, Pfsense, Smooth wall) that perform firewall functions. A brief description of the mechanisms discussed in this research is described below such as:

Watchward 550-E, is a comprehensive proprietary solution that includes inside its hardware, the implementation of multiple services and subscriptions for the perimeter network. Its functionality includes support for VPN, SSL, IPSEC, content filtering, anti-virus, anti-spam, IPS, NAT, Networking and Web capabilities management;

PfSense, is an open source network firewall and free distribution, FreeBSD customizable, having a Web interface to be configured. According to its official website [15], it has been successfully installed in different environments;

Zentyal, is an open source software that is able to act managing the network infrastructure such as Internet gateway, solving security threats, using a Web interface [16];

ClearOS, is a Linux distribution based on CentOS and Red Hat. This product is focused on its use as gateway (Gateway router), proxy server, firewall and DNS [17].

3. Experimental Setup

3.1 Design and Implementation of Test Topology

Fig. 1 illustrates the design of test topology. One is able to observe, that we designed and implemented a LAN and WAN network scenario separated by a firewall, which was configured in a real environment and validated in a virtual network environment, through three software systems: Pfsense, Zentyal and ClearOS. This laboratory allowed performing different assessments of the selected firewalls in a real environment using an identical system of tests to their different distributions.

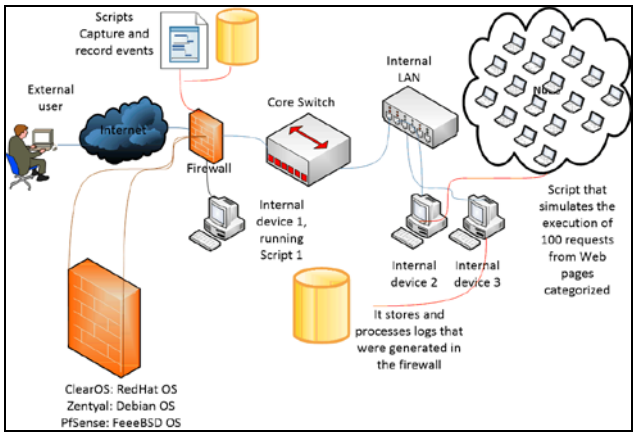

Fig 1. Design and implementation of the experimental topology. See text for explanation

As part of the experiment, for the analysis of the different systems for firewalling, we installed each of one in an external storage unit (hard disk). The purpose has been to offer equality to different scenarios of planned tests, which provisioned portability, independence, ease of assembly and disassembly, flexibility as well as stability evaluation. Through this performance, we were able to achieve lower loss boot system configuration of the local operating system and a complete backup of the hard disk in order to avoid potential reinstallation.

During this exploration we had a firewall black box (firebox-core) known as UTM WATCHGUARD 550E, whose annual license provides access control to the Web, IPS/IDS, Quality of Service (QoS), Antispam/Malware, viruses, worms, Trojans, phishing, spyware, and further more malware border level.

Within the network infrastructure LAN, a dynamic IP addressing DHCP scheme was organized, which has been assigned by the firewall in a network segment in the range 192.168.0.10 to 192.168.50.20 (for internal network equipment).

Tests

Once the experimental environment has been designed, the procedures and types of attacks for the execution of the tests remained established. The tests were addressed to determine the performance, CPU and memory consumption of firewall security system. The testing scheme to calculate the resource consumption of the CPU and memory firewall has been established based on some metrics such as number of requested Web pages, duration time of the test and finally the period of Web application pages.

The tests were realized from a computer on the local network, which has been based on the creation of a batch processing script (algorithm), generating the request for multiple connections to Web pages. The scenarios are listed in Table 1, while the conducted series of the several attacks are listed in Table 2.

Table 1: Scenarios evaluated, configured in firewalls.

Evaluated Scenarios
NAT + Content filtering
NAT + Content filtering + antivirus
NAT + Content filtering + antivirus + IPS

Table 2: Type of attacks and tools used

Attack	Tool
By scanning	ZenMap, nmap
Sniffing	The dude
Denial of services	Loic

3.2 Algorithm that simulates the request of various Web pages

One of the challenges in modern technology with limited availability of computers is the ability to perform multiple requests for web sites without a subsequent breakdown. This kind of environment in a usual production network is considered for the experiment we are proposing and presenting being the most common case when we are testing potential network attacks. Based on this situation, our experiment should be able to receive the request of a certain high amount of Web pages, for which it requires to possess a variety of computers capable to perform several requests as well as some additional desired requests above the total amount of such.

We may consider ten different computers perform simultaneously ten different queries. The outcomes would be relatively expensive because users should be able to achieve requests simultaneously and in an exact sequence, leading to an inefficient practice due to the lack of synchronization between each other. Therefore, an alternative solution has been implemented with a batch-processing algorithm, allowing to perform simultaneously multiple requests for Web pages from a single computer at different periods. During the time period when this algorithm has been is executed, the contents commands are carried out in groups sequentially, allowing automatically

various tasks such as reading the text file in which are downloadable, registered different addresses (URLs) of Web pages. Later, 15 seconds after the start of the execution, the default browser either gets initialized or it closes. Hence, it turns to run the requests of the pages encountered in the file, accomplishing a cyclic loop to n numbers of repetitions. The number of pages and the time period are both configurable. Fig. 2 indicates the proposed flow diagram and the corresponding algorithm code.

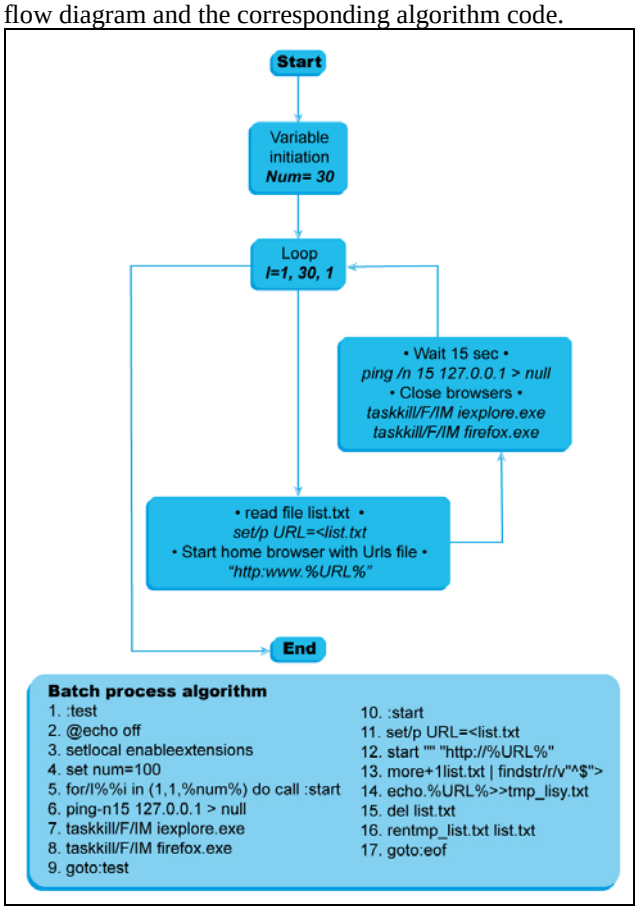

Fig 2. Algorithm which simulates the request of a hundred of web pages

3.3. Repowering of Firewalls

One of the main aims of our research study claimed to reinforce one of the open source tools that were evaluated at baseline, named CLEAROS. We have handled the concept of integrating and consolidating an inspection solution and unified threat management in the following two basic steps: Firstly, joining and repowering PSAD (Port scan attack Detector) [25] as a tool to detect attacks to port scanning. To reach this goal we have improved the source code, giving a resolution as a programming platform to produce optimal rules reordered in the engine [26]. Secondly, we included and reconfigured an IDS/IPS system, through Fw-snort and Snort filtering tools. Additionally, malicious network traffic were filtered, configured dynamically and cooperatively [27][28], with the potential of ClearOS, PSAD, Snort and Fw-snort tools. Fig. 3 illustrates the elements, which repowered the ClearOS firewall:

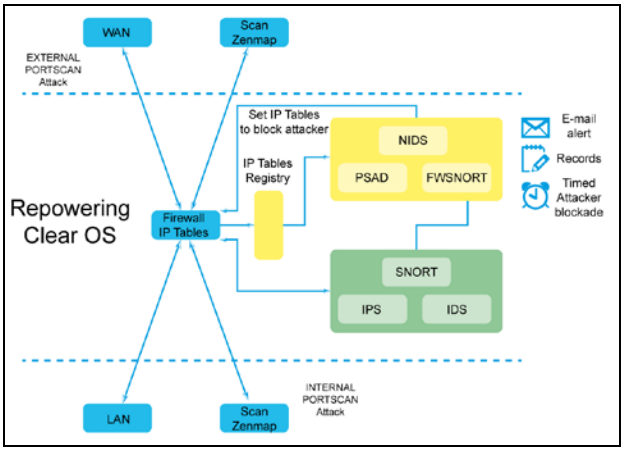

Fig 3. Diagram indicating the basic tools used in Repowering of ClearOS

This diagram consists of the following parameters to be functional: When including PSAD inside of ClearOS, it Iptable was reprogrammed to register any activity that it converges. In this manner, PSAD detects and blocks port scan attacks or other suspicious traffic. In addition to its activity PSAD interoperates with Snort to generate alerts to the rules to be detected through the use of a set of dynamic constraints generated by Fw-snort. The Fw-snort analyzes the rules included in the Snort and builds a set of rules equivalent to IPtables for any number of rules as possible. This allows PSAD to submit alerts to the application layer attacks.

When repowering ClearOS with PSAD, we obtain a powerful tool, which enables the use of several TCP, UDP and ICMP signatures included in Snort to detect suspicious traffic as backdoors, Distributed Denial of Service (DDoS) tools, and Operating Systems (OS) identification, among others. Thus, PSAD analyzes the TCP flags to determine its type (syn, order, xmas, etc.) for TCP scanning, which simultaneously solves the options corresponding command line that allows to be used in order to generate the port scanning with Nmap. This particularity has been explained in detail in a previous study [26].

ClearOS repowered saves the messages in the kernel information, as well as later all messages from that pipe (flow) in the syslog. In this manner, it is supplied to PSAD a flow of pure data that contains only packets that ClearOS and Snort are able to decide the relevance of entering or denying the entrance to the protected network.

Clear OS repowered with PSAD, increases the reliability of the security system for their continuous communication with a network manager, due to an automatic alert via e-mail with malicious activity reported by the tool. This tool includes the source IP address of port scan attack, the

number of packets sent to each port, any TCP, UDP or ICMP signature that have been identified (e.g. "NMAP XMAS scan"), the ports range, the danger level (1 to 5) reverse DNS information, as well as other necessary information.

4. Evaluation results

4.1 Firewalls evaluation, baseline

As previously explained (see Section 1, page 2), this research focuses on the evaluation of three systems of open source firewall and in the firewall repowering, based on the distinctive characteristics of a proprietary system. Therefore, to compare their presentation in terms of performance and effectiveness, we most likely select one tool of these, which basically gathers the best features and subsequently repower them.

The procedure started to run scripts inside the firewall by 60 seconds from device 1 of the test topology. At the same time, the device 2 executes the script of 100 http connections. After 60 seconds the device 3 is backing and processes the logs generated in the firewall. The procedure aimed to measure the consumption of CPU, memory, and performance in the three firewall systems in the scenarios described in Table 1 and also during the attacks given in Table 2. The results are presented in Figures 4, 5 and 6.

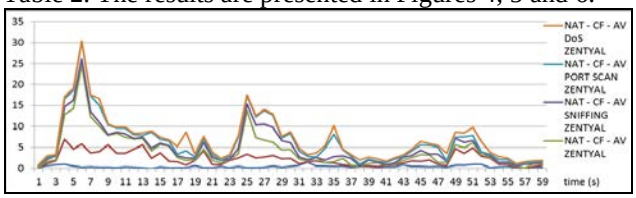

Fig 4. Number of records per unit time, using Zentyal

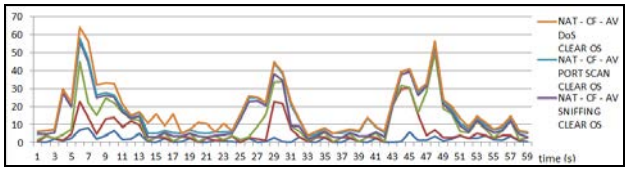

Fig 5. Number of records per unit time, using ClearOs

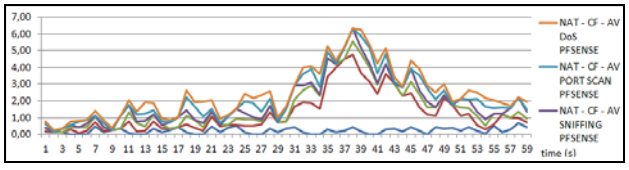

Fig 6. Number of records per unit time, using PFsense

In the Figures 4, 5 and 6 a clear compatibility is illustrated in the performance of each of the analyzed firewalls tested being directly proportional to the desired applicability. A preliminary assessment of the results in the physical environment, allows us to indicate that the firewall, which has a lower performance is the PFSense, followed by the Zentyal. Therefore, we conclude that ClearOS based on its performance is the best tool so far for firewall protection.

The evaluation of each tool in function of the consumption of CPU based on different scenarios (Table 1; 2) and is illustrated in the figures 7, 8 and 9. Obviously, PFsense consumes a greater percentage of the CPU, followed by ClearOS, while the percentage of CPU usage of Zentyal is much lower as it does not use its IPS.

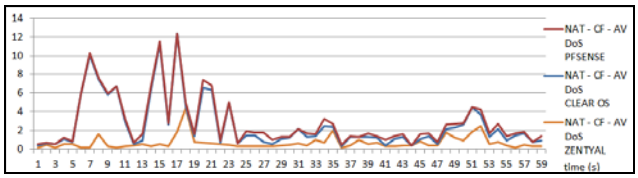

Fig 7. Evaluation of the firewall CPU consumption using scenario 1

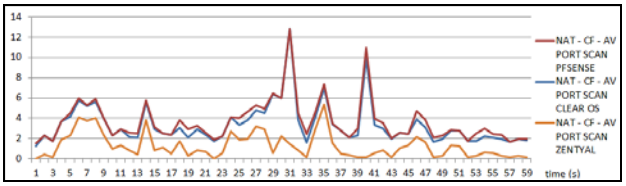

Fig 8. Evaluation of the firewall CPU consumption, using scenario 2

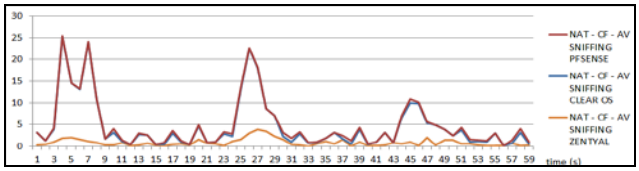

Fig 9. Evaluation of the firewall CPU consumption, using scenario 3

Finally, with respect to memory consumption, using the scenario 3 of Table 1, we are able to establish that PFsense has a higher memory consumption (35%), followed by ClearOS (26%), whereas Zentyal has the lowest consumption. This observation is compelling since during their processes they lacks of IPS, unlike ClearOS and PFsense respectively. Fig. 10 demonstrates these mentioned results. Therefore, in this research, the tool selected to be repowered, has been ClearOS.

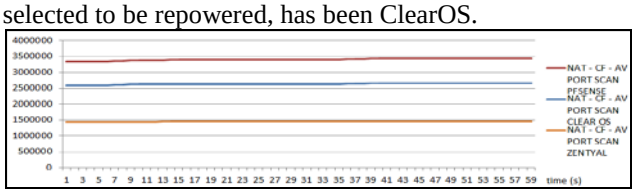

Fig 10. Evaluation of the firewall Memory consumption, using scenario 3

4.2 Evaluation of results after repowered the firewall.

ClearOS platform is a Linux-based server that is handled through a management tool in a Web environment. Most of the source code originates from Red Hat Enterprise Linux distribution. The programming is distributed in layers and levels and has a client-server architecture. Thus, the presentation layer contains all the pages that the user observes and informs the activities performed. Business layer contains programs, which are executed when the user needs a task. The data layer is located where the data resides.

ClearOS being a development tool that is used to level presentation PHP for its design in the Web environment. The Web server uses Apache, while in their business layer Java is used.

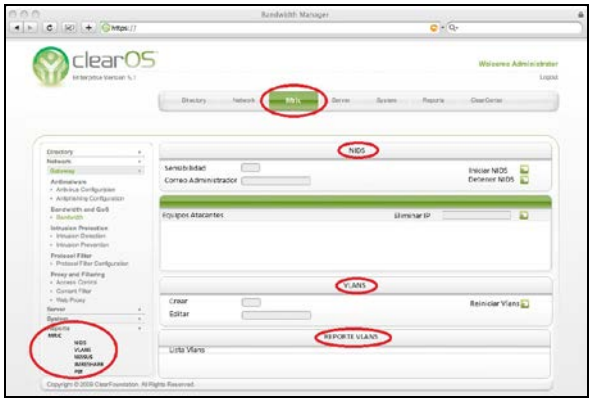

Fig 11. Graphical User Interface of repowered ClearOS

In Figure 11, our research demonstrates how the graphical user interface (front-end) ClearOS has been modified by incorporating personalized Mric option (integrated module upgrade, for its acronym in Spanish), which adds new libraries consolidating various security services in a single firewall, such as IDS / IPS, PSAD, and content filtering, repowering to ClearOS as a solution of Unified Threat Management (UTM).

Figure 12 points out the execution of the repowered ClearOS. To prove the proposed concept, the following task has been carried out. From a devise of the experimental topology, port scanning is performed with Zenmap towards the firewall ClearOS, which enabled services Antivirus, Content Filtering and Snort (IDS/IPS). Concurrently, a connectivity test was conducted in the firewall during 120 seconds.

Figure 13 demonstrates how permanent link continuity is maintained during a firewall test interval. It is also illustrated how a test interval is unable to detect or block the attacking device. In the same graphic, we are able to observe how the OS firewall Clear repowered with PSAD, whose code was improved, detects as well as blocks the attacking machine in a time period of 14 second.

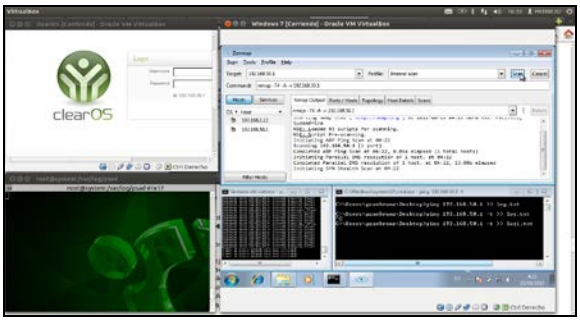

Fig 12. Evaluation of the repowered ClearOS

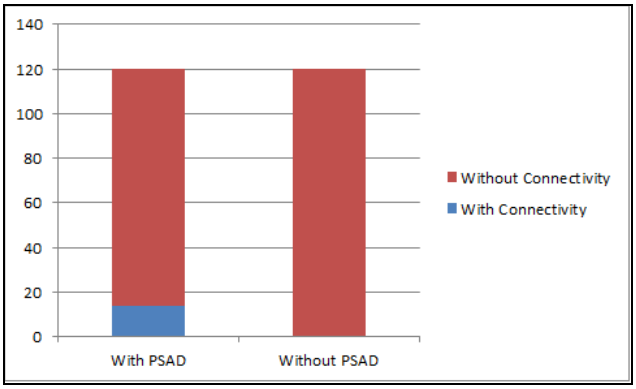

Fig 13. Analysis of continuity with repowered Clear OS

5. Related Work

There has been a significant amount of research regarding firewalls repowering since the 2000s. As an underpinning and a guide of our research has been used the method proposed by [3], where technical details are discussed to reprogrammed in the firewalls the inspection capabilities, detection, and control of cyber-attacks. Similarly, it has been based on the model proposed by Lyu and Lau in [4], in which the security is defined in seven different levels allowing quantifying the impact on performance correlation with the increment levels of security.

In other studies similar to our research, Misherghi et al., in [5], presents a general framework for optimization based on programs to produce optimal sets of firewall policy rules. In the model proposed by [6], a firewall that works with the integration of open source packages was assembled, recompiling the Linux kernel to ensure interoperability of these packages. In this mentioned paper the integration of open commercial products establishing ways of improving code are compared. In the model proposed by [7], different types of firewalls analyzing the causes and effects of performance deficiencies of existing firewalls both open source as well as commercial. The detailed analysis and comparison has been made in terms of cost, safety and ease of operation. The results indicated

that Cisco ASA outperforms its competitors in most performance criteria. In the model proposed by [8] and [17] study and evaluate both two specific firewall solutions that incorporate VPN, a firewall open source available for the Linux operating system, and a commercial firewall solution from Cisco. Compared to our study, none of them has repowered open source solution, modifying the source code of its components.

Regarding the techniques used by other researchers in evaluating the performance of firewalls, in [10] and [11], an analytical model based on Markov Chains is presented for analyzing the performance of firewalls based rules when they are exposed to normal traffic flows, as well as against Denial of service (DoS) attacks. In this article, equations of performance metrics including performance, packet loss, delay, and CPU utilization firewall are derived. They validated their model by simulation and actual experimental measurements. In [12], an innovative framework for management policies are represented by adopting a segmentation technique based on rules for identifying anomalies; in particular, rendering based technique used Grid. Compared with our work, we have efficiently quantitatively evaluated the performance of some systems open source firewall, comparing the features of a commercial solution and have repowered the chosen tool, giving the functionality of UTM solution.

6. Conclusions and Future Work

Present study focused on how to repower ClearOS firewall, in order to improve the detection, control and mitigation of cyber-attacks. It has been designed and implemented an experimental network scenario separated by a firewall, to evaluate Pfsense, Zentyal and ClearOS. It has been followed and performed by a quantitative evaluation of the performance and effectiveness of these systems. Using information obtained in the study, we were able to reinforce ClearOS by integrating and consolidating inspection solution and unifying threat management. The interpreted and quantified results demonstrate the effectiveness of our research.

Based on these results, our next goal will be to focus on how to repower ClearOS using machine-learning techniques.

References

[1] ABC Hoy Tecnología: “Las entidades públicas sufren unos 5.400 ataques al año”, disponible en: http://www.hoytecnologia.com/noticias/entidades-publicas-sufren-unos/173132. Última actualización: 01.11. de 2014.

[2] Al-Shaer, E., Hamed, H., Boutaba, R., Hasan, M.: “Conflict classification and analysis of distributed firewall policies”. IEEE J. Sel. Areas Communication, Oct 23, pp: 2069–2084 (2005).

[3] Michael Rash, Linux Firewalls: Attack Detection and Response with IPtables, psad, and fwsnort, ISBN: 10.1-59327-141-7. 2007.

[4] M. Lyu and L. Lau: “Firewall security: policies, testing and performance evaluation” in Proceding of 2000 IEEE International Computer Software and Applications Conference, pp. 116–121.

[5] G. Misherghi, L. Yuan, Z. Su, C. Chuah, and H. Chen, “A General Framework for Benchmarking Firewall Optimization Techniques”, IEEE Trans. Network and Service Management, vol. 5, no. 4, pp. 227-238, Dec. 2008.

[6] Ying-Dar Lin, Shao-Tang Yu, Huan-Yun Wei: “Integrating and Benchmarking Security Gateway with Open Source Firewall, VPN, and IDS”.

[7] Sheth, C. and Thakker, R., “Performance Evaluation and Comparative Analysis of Network Firewalls”, In Proceedings of International Conference on Devices and Communications (ICDeCom), 2011, IEEE, 2011.

[8] S. Patton, D. Doss, and W. Yurcik: “Open Source versus Commercial Firewalls: Functional Comparison”, Proceedings of the Conference on Local Computer Networks, pp. 223-224, November 2000.

[9] Hulst, J.W., Zihui Ge, Liu, A.X., Dan Pei, Jia Wang , “Firewall fingerprinting”, In Proceedings of IEEE INFOCOM, ,March 2012, Page(s): 1728 – 1736.

[10] Khaled Salah, Khalid Elbadawi, and Raouf Boutaba: “Performance Modeling and Analysis of Network Firewalls”, IEEE Transactions on Industrial Electronics, Vol. 59, No. 1, January 2012.

[11] JeeHyun Hwang, Tao Xie, Fei Chen, and Alex X. Liu: “Performance Modeling and Analysis of Network Firewalls”, IEEE Transactions on Network and Service Management, Vol. 9, No. 1, March 2012.

[12] H. Hu, G.J. Ahn, and K. Kulkarni: “Detecting and resolving firewall policy anomalies”, IEEE Transactions on Dependable and Secure Computing, Vol. 9, No. 3, pp. 318–331, June 2012.

[13] Hayajneh, Thaier, et al. Performance and Information Security Evaluation with Firewalls. International Journal of Security & Its Applications, 2013, vol. 7, no 6.

[14] National Vulnerability Database, automating vulnerability, security measurement, and compliance checking. Available at: http://web.nvd.nist.gov/view/vuln/statistics. Last updated, November 1, 2014.

[15] Pfsense, Home page: http://www.pfsense.org/, Last updated, November 1, 2014.

[16] Zentyal, Home page: http://www.zentyal.com/ last updated, November 1, 2014.

[17] ClearOS, Home page: http://www.clearfoundation.com/, Last updated, November 1, 2014.

[18] N. Freed, “Behavior of and Requirements for Internet Firewalls”, IETF RFC 2979, October 2000.

[19] SCARFONE, Karen; HOFFMAN, Paul. “Guidelines on firewalls and firewall policy”. NIST Special Publication, 2009, vol. 800, p. 41.

[20] DragonJAR, “Firewalls - Diseño y Panorámica Actual”, White Paper, Last update, November 15, 2014.

[21] D. Newman, “Benchmarking Terminology for Firewall Performance”, IETF RFC 2647, August 1999

[22] B. Hickman, D. Newman, S. Tadjudin, and T. Martin: "Benchmarking methodology for firewall performance" RFC 3511, Apr. 2003.
[23] Yan, W.; Zhang, Z.; Ansari, N. & Micro, T., "Revealing packed malware", IEEE Security & Privacy, 2008, 6, 65-69.
[24] Yongxin, Y., "The comparative study on network firewalls performance", In proceeding of IEEE 3rd International Communication Software and Networks (ICCSN), pp: 427-430, 2011.
[25] Psad, Home page: http://www.cipherdyne.org/psad/ Last update: 08/30/2012.
[26] Walter Fuertes, Patricio Zambrano, Marco Sánchez, and Pablo Gamboa, "Alternative Engine to Detect and Block Port Scan Attacks using Virtual Network Environments", in IJCSNS-International Journal of Computer Science and Network Security", Special Issues: Communication Network & Security. Vol. 11, No. 11, pp. 14-23. ISSN: 1738-7906. Nov. 30, 2011.
[27] J. Zhai and Y. Xie, "Researh on Network Intrusion Prevention System Based on Snort", In Proceedings of IEEE Strategic Technology (IFOST), 2011 6th International Forum on, vol. 2, pages:1133-1136, 2011
[28] Beale, J. and Baker, A.R. and Caswell, B. and Poor, M., "Snort 2.1 Intrusion Detection", Syngress Media Inc., 2004.

Walter Marcelo Fuertes Díaz (wmfuertes@espe.edu.ec) is a professor-researcher in the Department of Computer Science of the Universidad de las Fuerzas Armadas ESPE. He received his MSc degree in Computer Networking, from the Escuela Politécnica Nacional in Quito-Ecuador, in 1999, and the Ph.D. (honors) degree in Computer Science and Telecommunications engineering from Universidad Autónoma de Madrid (UAM), Madrid, Spain, in 2010.

Patricio Zambrano Rodriguez (patricio.zambrano@tce.gob.ec) currently works as Network Administrator at the Tribunal Contencioso Electoral in the technology's area at Quito-Ecuador. He received a master in science degree in Information Networks and Connectivity at the "Escuela Politécnica del Ejército" of Sangolquí-Ecuador. His research line is computer networks security.

Marco Polo Sanchez Aguayo (mpsanchez@tvcable.com.ec) currently works as Software Designer at the "Grupo TvCable" in the technology's area at Quito-Ecuador. He received a master in science degree in at the "Escuela Politécnica del Ejército" of Sangolquí-Ecuador in 2012. His research line is computer networks security.

Mónica Lucía Santillán Trujillo mlsantillan@espe.edu.ec
Currently is a Professor-Researcher at the Universidad de las Fuerzas Armadas ESPE of Ecuador. She is Ph.D. in Philological Sciences; their research area concerns Semiotics and Culture. Is part of two research groups in the "contents" area: SMARTCOM –Digital TV and RACKLY Distributed systems, Security, and contents.

César Javier Villacís Silva (cjvillacis@espe.edu.ec) currently works as a professor-researcher in the Computer Science Department of the Universidad de las Fuerzas Armadas ESPE from Sangolquí, Ecuador. He received his MSc in Computer Technology Applied to Education, from the Universidad Tecnológica Israel, in Quito-Ecuador, in 2014.

Theofilos Toulkeridis (ttoulkeridis@espe.edu.ec) received his B.Sc., M.Sc. and Ph.D. degrees in 1988, 1992 and 1996 respectively, in the areas of geology / paleontology and natural sciences (geology and isotope geochemistry) from the Johannes Gutenberg University in collaboration with the Geochemistry Division of the Max Planck Institute in Mainz, Germany and the Center of Surface Geochemistry in Strasbourg France. He became full professor in the University Of San Francisco De Quito (1999) and later in the Universidad de las Fuerzas Armadas ESPE (2011).

Edgar Porfirio Torres Proaño (eptorre@espe.edu.ec) currently is a faculty member in the Computer Science Department (DECC) at the Universidad de las Fuerzas Armadas ESPE, Sangolquí Ecuador. He is a Professor Researcher mainly in the areas of Distributed Systems and Digital Image Processing. He graduated and received the Master of Science (MSc) degree, in Electrical Engineering and Computer Science (1980), and completed All Formal Academic Coursework required for the PhD degree Program (1987), at Ohio University, Athens Ohio, United States Of America.