4th International Conference on Automation, Control Engineering and Computer Science (ACECS - 2017)
Proceedings of Engineering and Technology – PET
Vol.19, pp.12-16

Implementation and Performance Evaluation of Intrusion Detection and Response System

Mohammed Ghaouth BELKASMI, Mohammed SABER, Sara CHADLI, Mohamed EMHARRAF, Ilhame EL FARISSI
Laboratory SE2I, ENSAO, Mohammed First University, Oujda, Morocco
Email: (mosaber,ghaouth,chad.saraa,m.emharraf,ilhame.elfarissi)@gmail.com

***Abstract*—Modern intrusion detection systems (IDS) are deployed in high-speed networks. Thus, they must be able to process a large amount of data in real time. This raises the issue of performance and required an evaluation of these IDS.**

We present in this paper an evaluation approach, based on a series of tests. The aim is to measure the performance of the components of an IDS and their effects on the entire system, as well as to study the effect of the characteristics of the deployment environment on the operation of the IDS. So, we have implemented the IDS SNORT on machines with different technical characteristics and we have designed a network to generate a set of experiments to measure the performances obtained in the case of a deployment in high-speed networks . These experiments consist in injecting various traffic loads, characterized by different transmission times, packet numbers, packet sizes and bandwidths, and then analyzing, for each situation, the processing performed on the packets.

Our experiments have revealed the weaknesses of the IDS in a precise way. Mainly, the inability to process multiple packets and the propensity to deposit, without analysis, packets in high-speed networks with heavy traffic. Our work also determined the effect of a component on the entire system and the effect of hardware characteristics on the performance of an IDS.

***keywords* – Intrusion Detection System, SNORT, Performance Evaluation, Traffic, Packet Dropped.**

I. Introduction

For many years attacks made on networks have risen dramatically. The major reason for this is the unlimited access to and use of software by inadequately trained people. Network disruptions may be caused intentionally by several types of directed attack. These attacks are made at various layers in the TCP/IP protocol suite, including the application layer. Besides the external body, attacks can be made on the network by the internal body as well. However, an IDS is considered to be one of the best technologies to detect threats and attacks. IDSs have attracted the interest of many organisations and governments, and any Internet user can deploy them.

The evaluation of intrusion detection systems is a challenging task; it requires a thorough knowledge of techniques relating to different disciplines, especially intrusion detection, methods of attack, networks and systems, technical testing and evaluation [1], [2]. What makes the evaluation more difficult is the fact that different intrusion detection systems have different operational environments and can use a variety of techniques for producing alerts corresponding to attacks.

In practice, most of IDSs suffer from several problems, taking into consideration the large number of false positives and false negatives, and the evolution of attacks. All these problems increase the need of implementing an IDSs evaluation system. In this context, many attempts took place [3], [4], [5], [6], [7], [8], [9].

We present in this paper an evaluation approach, based on a series of tests. The aim is to measure the performance of the components of an IDS and their effects on the entire system, as well as to study the effect of the characteristics of the deployment environment on the operation of the IDS. So, we have implemented the IDS SNORT on machines with different technical characteristics and we have designed a network to generate a set of experiments to measure the performances obtained in the case of a deployment in high-speed networks . These experiments consist in injecting various traffic loads, characterized by different transmission times, packet numbers, packet sizes and bandwidths, and then analyzing, for each situation, the processing performed on the packets.

In the remaining sections of this article, we quote related works in Section 2. We discuss the proposed evaluation approach for evaluating performances of IDS in Section 3. In Section 4, we present the results and evaluation. Finally, we end up our paper with a conclusion and future works in the 5 section.

II. The proposed approach for evaluating performances IDS

A. Performance Test

To measure the performance of the IDS components. We focus on the IDS SNORT [10], [11] capability as a network intrusion detection system as we aim to see how many packets could be analysed by SNORT under varying conditions. It has been shown previously that in high speed and heavy load conditions, some packets are dropped (skipped or not processed) [8], [5]. In the proposed experiments SNORT analyser is not set up to perform actions based on user defined rules. It simply analyses and identifies the packets.

B. Test scenarios

These scenarios were designed to test the performance of SNORT on different station. IDS were subject to the same tests and under the exact same conditions. In order to get more accurate results, we consider the following scenarios :

- **Scenario 1 : High speed traffic** : we have sent the packets (1KB in size) at different transmission time

ISSN 2356-5608

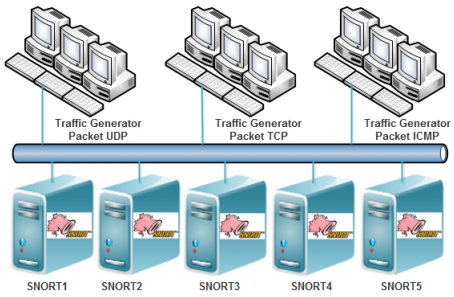

Fig. 1. Network test for scenario

frames (1, 4, 8 and 16 ms), to analyze the system response within high speed traffic.

- **Scenario 2 : Heavy traffic** : we have sent the different sets of packets having different volumes (100, 1000, 5000 and 10000), to analyze the system response within heavy traffic.
- **Scenario 3 : Large data traffic** : we have sent the different sizes (lengths) of packets (128, 256, 512 and 1024 bytes), to analyze the system response in case of large data traffic.
- **Scenario 4 : Traffic bandwidth** : we have generated traffics with different bandwidth (250 Mbps, 500 Mbps, 1.0 Gbps and 2.0 Gbps), to analyze the system response according to traffic bandwidth.

C. Test Bench

To perform those tests, we selected the SNORT version 2.9.7.2. And we have created a network including a computer connected to our platform for supervising operations and gathering results. The network is composed of 14 computers (table I), depending on our need of generating (running both open source tools and commercial tools) smaller packet size on high traffic speeds. All these computers are connected via Cisco Catalyst Series 2960-G switch using 24 ports of 1 Gigabit Ethernet desktop connectivity as shown in figure 1.

D. Packet generation

The performance of TCP, and UDP protocols was measured when running over the IPv4 header. The LAN traffic generator (WinPcap and Packets Generator tool) were used to vary the type of traffic in terms of IP header protocol (TCP, and UDP), speed, the number of packets and packet size. The traffic generator was used to send a packet of various sizes (128, 256, 512 and 1024 Bytes) that contain data and other attacks.

E. Performance Metrics

Performance metrics are used in the experiments to measure the ability of the SNORT to perform a particular task and to fit within the performance constraints. These metrics measure and evaluate the parameters that impact SNORT performance. The following aspects were measured in the experiments. The metrics are the percentages of the total packets processed by system SNORT. The specific metrics used are shown in Table II.

TABLE I
NETWORK COMPONENTS SPECIFICATIONS TEST SCENARIOS

Machine	Hardware Description	Operating System	Tools used
Generator	Dell T1650 μp i3, 4GB RAM, 1Gbps network card	Windows 7 / Debian 8	LAN Traffic Generator (TCP and UDP)
SNORT1	DELL PowerEdge R910 4 x Intel Xeon E7-4820 8 core 2Ghz, 64GB RAM, Network card Broadcom 57711(2 x 10 GbE et 2 x 1 GbE)	Debian 8	SNORT
SNORT2	Station HP Z620 Intel Xeon E5-1607 4 core, 12GB RAM, Intel(R) 82579LM Gigabit Network Connection	Windows Server 2008 R2	SNORT
SNORT3	Station HP Z620 Intel Xeon E5-1607 4 core, 12GB RAM, Intel(R) 82579LM Gigabit Network Connection	Debian 8	SNORT
SNORT4	Dell T1650 μp i3, 4GB RAM, 1Gbps network card	Debian 8	SNORT
SNORT5	HP Pro 3010 Intel Core 2 Duo E7500, 4GB RAM, 1Gbps network card	Debian 8	SNORT

TABLE II
DESCRIPTION OF PERFORMANCE METRICS

Performance metrics	Description
Packets captured (**PCA**)	The number and percentage of packets received.
Packets analysed (**PAN**)	The number and percentage of packets analysed from the total packets captured.
Packets dropped (**PDR**)	The number and percentage of the packets dropped from the total packets captured.

III. SENARIOS RESULTS AND EVALUATION

A. Scenario 1: SNORTs response to high-speed network traffic

For this scenario, we sent ≅60000 packets 1KB in size (≅40000 TCP, and ≅20000 UDP) at different transmission time frames (1ms, 4ms, 8ms, and 16ms) for the five systems (SNORT1, SNORT2, SNORT3, SNORT4 and SNORT5). Table III shows the SNORT output and results of experiments.

In this experiment we notice that SNORT analysis performance decreased as the speed of transmission was increased. We deduce that the components ability of analysis becomes weaker as we increase the transmission speed.

ISSN 2356-5608

TABLE III
SAME NUMBER OF PACKETS BUT DIFFERENT TRANSMISSION TIME FRAMES

Traffic Type		SNORT1				SNORT2				SNORT3				SNORT4				SNORT5			
		1ms	4ms	8ms	16ms	1ms	4ms	8ms	16ms	1ms	4ms	8ms	16ms	1ms	4ms	8ms	16ms	1ms	4ms	8ms	16ms
TCP	PCA	40256	40925	41013	41002	40769	40953	40831	40726	40081	41007	40913	40993	40081	41007	40913	40993	41952	42018	42006	42003
	PAN	21016	29672	36821	40981	21361	29351	36425	40685	21031	29462	36521	40981	21031	29462	36521	40981	11231	21380	31211	41902
	PDR	19240	11253	4192	21	19408	11602	4406	41	19050	11545	4392	12	19050	11545	4392	12	30721	20638	10795	101
UDP	PCA	20098	20101	20017	20006	20061	20045	20081	20092	20119	20007	20103	20017	20143	20089	20071	20069	20023	20089	20011	20039
	PAN	10528	14539	17898	19997	10439	14296	18002	20027	10486	14237	17901	19987	7951	12187	15933	19977	5391	10238	14889	19926
	PDR	9570	5562	2119	9	9622	5749	2079	65	9633	5770	2202	30	12192	7902	4138	92	14632	9851	5122	113

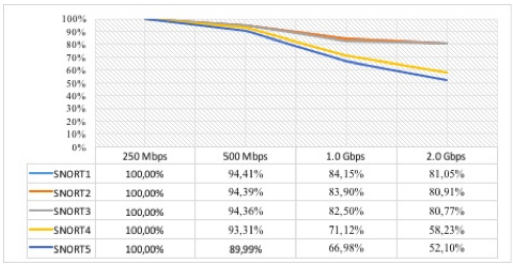

Fig. 2. Results: Packets (PCA, PAN, PDR (%)), Different bandwidth traffic

B. Scenario 2: SNORTs response to heavy-traffic networks

Here, the transmission rate of packets was kept to the same speed of 16 ms time frame, choosen in order to avoid droppin packets as shown in previous experiment. So we will obtain a fair analysis of different numbers of packets (each packet carried 1024). We sent 100, 1000, 5000, and 10000 packets sets at 16 ms time frame. Table IV show the SNORT output and results of the experiment.

This experiment shows that as the number of packets increases, more packets are dropped (Figure **??**).

C. Scenario 3: SNORTs response to large packets

For this experiment, the number of packets was kept to the same value (≅30000) and the same speed (16ms) to obtain a fair analysis of different sizes (lengths) of packets which are : 128, 256, 512 and 1024 bytes. Table V show the performance detection results.

In this experiment we realize that more packets will be dropped as packet size increases.

D. Scenario 4: SNORTs response to traffic bandwidth

Test scenarios were designed to assass the SNORT performance on different Computers. IDSs were subject to the same tests and under the same conditions. The test was performed for the speed ranging from 250Mbps, 500Mbps, 1.0Gpbs and 2.0Gbps. Figure 2 show the performance detection results.

This experiment demonstrated that more packets will be dropped as bandwidth increases.

E. Discussion of results

Test scenarios were designed to evaluate the SNORT performance on different computers. IDS were subject to the same tests and under the same conditions. In order to get more accurate results, all IDS in scenario 1, were tested with different transmission time frames (1ms, 4ms, 8ms, and 16ms). When, all systems were tested with number of packet (100, 1000, 5000, and 10000) in scenario 2. In scenario 3, all systems were tested with packet sizes (128, 256, 512 and 1024 bytes). Finaly, all systems were tested for the speed ranging from 250Mbps, 500Mbps, 1.0Gpbs, and 2.0Gbps.

On the basis of the results obtained from the four experiments, we noticed an increase of rejected packets number for all five environments, either with a lessening of transmission time in Scenario 1, or with an increase of packets number in Scenario 2, or increased packet size in Scenario 3, or with the increase in bandwidth in Scenario 4.

Our experiments have revealed the weaknesses of the IDS in a precise way. Mainly, the inability to process multiple packets and the propensity to deposit, without analysis, packets in high-speed networks with heavy traffic. Our work also determined the effect of a component on the entire system and the effect of hardware characteristics on the performance of an IDS within the environment SNORT4 and SNORT5. Hence, the problem is not primarily related to the physical characteristics of the deployment environment, but rather to the limitations of the IDS itself, so we have to look for a solution which will combine the IDS with other network components to improve the efficiency of our system.

IV. CONCLUSION

This research has focused on ways of determining the efficacy of the performances concept for IDS in high-speed network environments. The test scenarios employed involved the use of the widely deployed open-source IDS, namely SNORT. The results obtained have shown a number of significant limitations in the use of IDS, where both packet-handling and processing capabilities at different traffic loads were used as the primary criteria for defining system performance. We have further shown that performance is further degraded as the traffic is increased, irrespective of the host hardware used. Furthermore, we have demonstrated a number of significant

ISSN 2356-5608

TABLE IV
SAME TRANSMISSION TIME FRAME BUT DIFFERENT NUMBER OF PACKETS

	SNORT1				SNORT2				SNORT3				SNORT4				SNORT5			
	100	1000	5000	10000	100	1000	5000	10000	100	1000	5000	10000	100	1000	5000	10000	100	1000	5000	10000
PCA	103	1013	5008	10003	105	1007	5003	10006	109	1005	5011	10019	111	1010	5018	10021	104	1011	5019	10031
PAN	103	807	3508	5231	105	804	3511	5221	109	801	3541	5291	111	706	2845	4883	104	623	2549	4321
PDR	0	206	1500	4772	0	203	1492	4785	0	204	1470	4728	0	304	2173	5138	0	388	2470	5710

TABLE V
SAME TRANSMISSION TIME FRAME BUT DIFFERENT SIZE OF PACKETS

	SNORT1				SNORT2				SNORT3				SNORT4				SNORT5			
	128	256	512	1024	128	256	512	1024	128	256	512	1024	128	256	512	1024	128	256	512	1024
PCA	30002	30005	30009	30007	30011	30008	30027	30013	30017	30015	30027	30004	30018	30013	30019	30009	30021	30014	30025	30012
PAN	30002	28691	24613	20017	30011	28461	24601	19723	30017	28453	24521	19588	30002	26262	21789	14872	29721	25721	18231	9731
PDR	0	1314	5396	9990	0	1547	5426	10290	0	1562	5506	10416	16	3751	8230	15137	300	4293	11794	20281

differences in the performance characteristics of the five different environments in which SNORT was deployed.

This work has identified specific and replicable bottlenecks in commonly used implementations IDS in high-speed networks. The results obtained can be taken as a benchmark for improving the performance of these systems in future research work.

REFERENCES

[1] Khorkov, D. A. "Methods for testing network-intrusion detection systems". Scientific and Technical Information Processing, 2012, vol. 39, no 2, p. 120-126. DOI=10.3103/S0147688212020128.

[2] R. Berthier, W. H. Sanders and H. Khurana, "Intrusion Detection for Advanced Metering Infrastructures: Requirements and Architectural Directions," 2010 First IEEE International Conference on Smart Grid Communications, Gaithersburg, MD, 2010, pp. 350-355. doi: 10.1109/SMARTGRID.2010.5622068

[3] Akhlaq, M., Alserhani, F., Awan, I., Mellor, J., Cullen, A. J., & Al-Dhelaan, A. (2011). "Implementation and evaluation of network intrusion detection systems". In Network performance engineering (pp. 988-1016). Springer Berlin Heidelberg. DOI=10.1007/978-3-642-02742-0_42.

[4] Saber Mohammed, Chadli Sara, Emharraf Mohamed, EL Farissi Ilhame. Modeling and implementation approach to evaluate the intrusion detection system. Springer Lecture Notes in Computer Science (2015), Vol. 9466, pp. 513-517. DOI: 10.1007/978-3-319-26850-7_41.

[5] Shiri, F.I.; Shanmugam, B.; Idris, N.B., "A parallel technique for improving the performance of signature-based network intrusion detection system," Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on , vol., no., pp.692,696, 27-29 May 2011. DOI=10.1109/ICCSN.2011.6014986.

[6] Muhammad Asim Jamshed, Jihyung Lee, Sangwoo Moon, Insu Yun, Deokjin Kim, Sungryoul Lee, Yung Yi, and KyoungSoo Park. 2012. "Kargus: a highly-scalable software-based intrusion detection system". In Proceedings of the 2012 ACM conference on Computer and communications security (CCS'12). ACM, New York, NY, USA, 317-328. DOI=10.1145/2382196.2382232.

[7] Saber Mohammed, Chadli Sara, Emharraf Mohamed, EL Farissi Ilhame. Performance evaluation of an intrusion detection system. Springer Lecture Notes in Electrical Engineering (2016),Vol. 381, pp. 509-517. DOI: 10.1007/978-3-319-30298-0_52.

[8] Albin, E.; Rowe, N.C., "A Realistic Experimental Comparison of the Suricata and SNORT Intrusion-Detection Systems," Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on , vol., no., pp.122,127, 26-29 March 2012. DOI=10.1109/WAINA.2012.29.

[9] Xinli Wang, Alex Kordas, Lihui Hu, Matt Gaedke, and Derrick Smith. 2013. "Administrative evaluation of intrusion detection system". In Proceedings of the 2nd annual conference on Research in information technology (RIIT '13). ACM, New York, NY, USA, 47-52. DOI=10.1145/2512209.2512216.

[10] Ruinan Chi, Intrusion detection system based on Snort, in Proceedings of the 9th International Symposium on Linear Drives for Industry Applications, vol.3, Springer, Heidelberg, Berlin, 2014, pp.657664. DOI: 10.1007/978-3-642-40633-1_82.

[11] Martin Roesch. 1999. "SNORT - Lightweight Intrusion Detection for Networks". In Proceedings of the 13th USENIX conference on System administration (LISA '99). USENIX Association, Berkeley, CA, USA, 229-238.

ISSN 2356-5608