Snort over NTOP DNA Performance Test

Silicom 10G dual port with DNA driver

Eneo Tecnologia redBorder IPS/IDS

Date: July 30, 2012

Snort over DNA Compare to af_packet and pf_ring Snort Performance Test

The intention of this test was to do some compare test between the vanila Snort based on Open Source Project
and compare the performance results with DAQ over PF_Ring and DNA solution.
We would like to thank Pablo and Jeime Nebrera from Eneo Tecnologia for providing the Traffic Generator pfsend burn test,
redBorder IPS / IDS solution, Luca Deri and Alfredo Cardigilano from ntop.org for porting DAQ to the latest pf_ring / DNA
and performance / clustering enhancements.
All tests were run on Silicom 10G cards supporting Bypass.

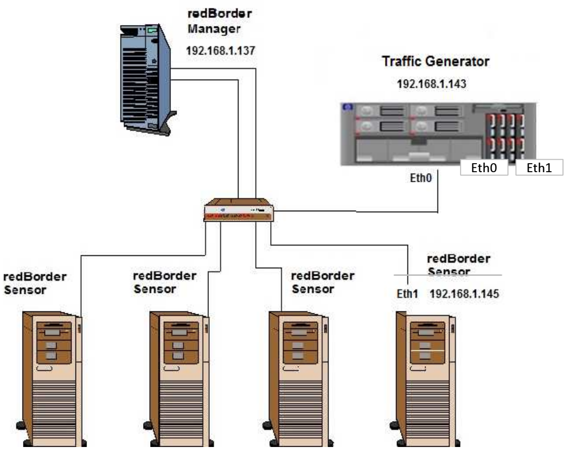
Report clear shows that with DAQ over DNA and redBorder IPS/IDS we are geting better results then the vanila Snort

Snort with DAQ over PF_Ring test configuration and setup:

Computer1 : Intel Xeon, **E3123 3200 MHz**, memory - 8 Gb - L1 Cache 64kB - L2 Cache 256 kB - L3 Cache 12 MB, 8 physical cores (Hyperthread - enabled) - 16 Logical cores
Network adapters: PE 210G2SPI9-SR, slot PCI-E x8 v. 2.0
OS: Linux Bitfrost
Software: ./battery.sh
TrafficGenerator:
Note: Computer1 was used as traffic generator

Computer2 : Intel X58 (Tylersburg), **Xeon3565 3200 MHz**, memory - 3 Gb - L1 Cache 64kB - L2 Cache 256 kB - L3 Cache 12 MB, 4 physical cores
Network adapters:
OS: Linux CentOS 6.2 x 64
Software: redBorder
Note: Computer2 was used as redBorder Manager

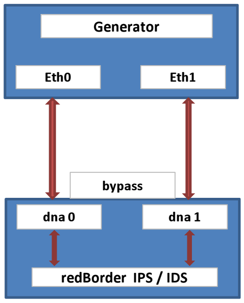
Computer3 : Intel Xeon, **XeonX5570 2900 MHz**, memory - 16 Gb - L1 Cache 64kB - L2 Cache 256 kB - L3 Cache 12 MB, 8 physical cores
Network adapters: PE 210G2BPI9-XR (SN: E296501400033), slot PCI-E x8 v. 2.0
OS: Linux CentOS 6.2 x 64
Software: redBorder
Note: Computer3 was used as redBorder Sensor



Host andTarget Platform Description:

Host: Pktgen Generator with 2 interfaces 10G (Eth0 and Eth1)

Target: redBorder IPS / IDS Snort sensor with 2 interfaces 10G (dna0 and dna1)



Software Configuration:
Linux Setup:
1. Linux Version: CentOS 6.2 64 Bit
2. Kernel Version: 2.6.32 -220.e16. x86_64

Traffic Setup:

The Computer1 has Silicom PE210G2SPI9A-SR - 2 ports 10G Ethernet card and used as Traffic Generator

The 2 x 10 GbE ports directly connected in full duplex to the redBorder Sensor installed at Computer 3

RedBorder Management system installed at Computer2 to monitor Sensor system capabilities (CPU load, RAM usage,..)

as well as Snort parameters (Mbps, Kpps, CPU, Alerts..)

The following scenarios were tested:

1 Segment Uni-Directional IPS ; 1 Segment Uni-Directional IDS; 1 Segment Bi-Directional IPS; 1 Segment Bi-Directional IDS

In each one of the scenarios we tested:

Bypass; No-rules; Connectivity, Balanced, Security traffic with 6k and 12k rules

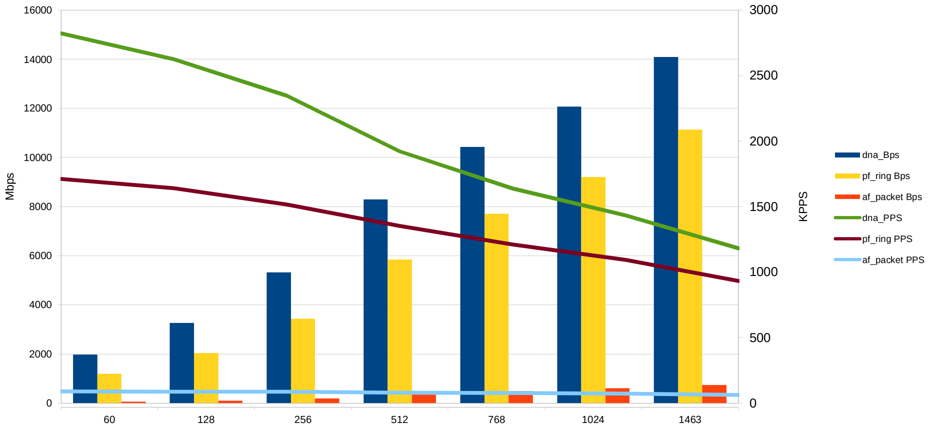
Bypass: Segment in bypass mode. Snort does not do anything, testing the generator limit.
No-rules: Snort has no rules activated. Only preprocessor works forwarding packets.
Connectivity: Snort assign rules according to the VRT connectivity policy.
Balanced: Snort assign rules according to the VRT balanced policy.
Security: Snort assign rules according to the VRT security policy.
6k: Snort assign rules according to the VRT security policy + commented rules (noisy rules) .
12k: Snort assign rules according to the all VRT rules.

1. Configure the traffic generator to send packets of a certain size until test fails
2. Change packet size to the next one (64,128, 256, 512, 1024 and 1500) again untill test fail
3. Connect the Snort device under test (DUT) and change the number of rules, restarting the service
4. Repeat all the testing
5. Execute all the tests with the listed above scenarios untill all packet sizes, all rules combination are tested
6. Compare IPS mode vs. IDS forwarding
7. Store all data in CSV file format

Results Analysis:

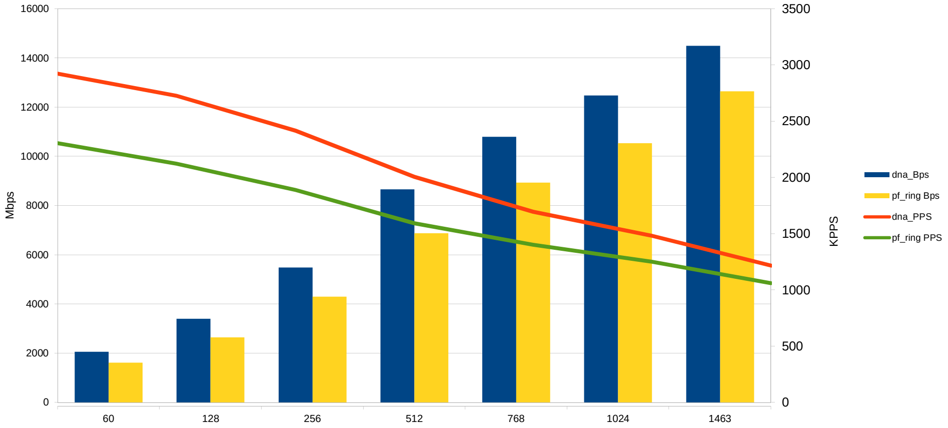
IPS 10G Bidirectional VRT Balanced								
Packet size:	**60**	**128**	**256**	**512**	**768**	**1024**	**1463**	
dna_Bps	1987.9	3276.1	5333.5	8305.2	10436.5152	12076.1	14095	36.67%
pf_ring Bps	1206.2	2048.1	3447.1	5848	7711.0112	9212.15	11144	1475.48%
af_packet Bps	63.008	111.07	199.25	348.19	500.5248	609.318	746.69	
dna_PPS	2823.7	2625.1	2347.5	1922.5	1638.9	1434.9	1181.7	47.64%
pf_ring PPS	1713.3	1641.1	1517.2	1353.7	1210.9	1094.6	934.3	1588.99%
af_packet PPS	89.5	89	87.7	80.6	78.6	72.4	62.6	

IPS 10G VRT Balanced



IDS 10G Bidirectional VRT Balanced								
Packet size (bytes):	**60**	**128**	**256**	**512**	**768**	**1024**	**1463**	
dna_Bps	2057.8	3404.5	5491.2	8668.1	10807.1328	12479.2	14502	20.65%
pf_ring Bps	1622.9	2650.6	4295.7	6886.1	8932.3936	10538.5	12657	
dna_PPS	2923	2728	2416.9	2006.5	1697.1	1482.8	1215.8	24.42%
pf_ring PPS	2305.3	2123.9	1890.7	1594	1402.7	1252.2	1061.1	

IDS 10G VRT Balanced


Mbps
KPPS
dna_Bps
pf_ring Bps
dna_PPS
pf_ring PPS