What's new in PF_RING, n2disk, nBox

Alfredo Cardigliano
cardigliano@ntop.org
PF_RING

- As of today PF_RING provides:
  - (Limited) packet capture acceleration with any adapter using Linux kernel drivers
  - XDP (Linux eXpress Data Path) acceleration with Linux drivers supporting AF_XDP
  - Best (Zero-Copy Kernel-Bypass) acceleration with PF_RING ZC drivers up to 100 Gbps with:
    - Commodity adapters from Intel, Mellanox
    - FPGA adapters from Napatech, Silicom FPGA and other vendors
NVIDIA/Mellanox Adapters

• PF_RING ZC driver for ConnectX 4/5/6
• Performance up to 100 Gbps
• Hardware packet timestamps
• Hardware packet filtering
• Load-balancing (RSS)
• Traffic duplication!
Intel Adapters

• Supported families:
  ◦ **e1000e** (8254x/8256x/8257x/8258x)
  ◦ **igb** (82575/82576/82580/1350)
  ◦ **ixgbe** (82599/X520/X540/X550)
    • **ixgbevf** (ixgbe VF)
  ◦ **i40e** (X710/XL710/XXV710)
    • **iavf** (i40e VF) **NEW**
  ◦ **ice** (E810)
  ◦ **fm10k** **DEPRECATED**
Intel with VFs

- SR-IOV Virtual Functions are virtualized instances of the physical interface (usually used by VMs)
- Traffic is steered to VFs based on MAC (and VLAN)
- i40e VFs (iavf) support **trust mode** which enables promiscuous capture (with **duplication**!)
n2disk
n2disk

• n2disk provides continuous recording: in most cases it’s not possible to predict when a network event occurs, on-demand capture is not enough

• Data retention depends on traffic rate and storage size

<table>
<thead>
<tr>
<th>Traffic rate</th>
<th>10 Gbps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data on disk (sec)</td>
<td>1,2 GB/s</td>
</tr>
<tr>
<td>Data on disk (hour)</td>
<td>4 TB/h</td>
</tr>
<tr>
<td>Data on disk (day)</td>
<td>100 TB/day</td>
</tr>
</tbody>
</table>
Saving Space

• Packet compression: save up to 5% on Internet traffic (more on LAN traffic)
• Packet slicing: good if interested in headers only
• BPF filtering: difficult to predict
• L7 filtering: good to discard or shunt unwanted traffic (e.g. encrypted, compressed, multimedia)
Not all traffic is alike

• What if our storage does not satisfy the desired data retention, even after filtering?

• Traffic matching Network events is more important than the rest of the traffic

• We want to:
  ◦ Prioritize selected traffic (e.g. security alerts)
  ◦ Delete the rest of the traffic first, when the disk is full
Smart Data Retention

• Process Network events generated by ntopng
• Use a 1st level storage to implement continuous recording with a short data retention (cache)
• Use a 2nd level storage to archive traffic for Network events with a longer data retention (archive)
nBox Appliance

• A turnkey solution for those who don't want to bother with hardware selection, software installation and tuning

nBox NetFlow

nBox Recorder
nBox UI

• Supported on Ubuntu only
• UI based on old technologies (Perl CGI)

• It's time to rewrite it from scratch!
New nBox UI

• Integrated in Cockpit, an open source web-based UI for servers sponsored by Red Hat
• Runs on most Linux distributions, including Ubuntu, Debian, CentOS
• Extensible by means of plugins (Javascript API)
• ntop plugins written in modern HTTP and Vue.js
New nBox UI

Live Preview!
Already on Github

• Contributions are welcome!
Thank you