Open Source in Network Administration: the ntop Project

Luca Deri <deri@ntop.org>
Project History

• Started in 1997 as monitoring application for the Univ. of Pisa
• 1998: First public release v 0.4 (GPL2)
• 1999-2002: Registered ntop.org, created mailing lists (ntop and ntop-dev) port to several platforms and Linux distro’s.
• 2002-03: Version 2.x, added support for commercial protocols (NetFlow v5 and sFlow v2).
• 2004-05: Version 3.x, added RRD support, IPv6 (Loria) and SCSI/FibreChannel (Cisco) support, NetFlow V9/IPFIX (draft), sFlow v5, VoIP.
• 2006-08: ntop consolidation, PF_RING 3.x, n2n 1.x
What is ntop? [1/2]

ntop is a simple, open source (GPL), portable traffic measurement and monitoring tool, which supports various management activities, including network optimization and planning, and detection of network security violations.
What is ntop? [2/2]

Host Information

<table>
<thead>
<tr>
<th>Host</th>
<th>Domain</th>
<th>IP Address</th>
<th>MAC Address</th>
<th>Community</th>
<th>Other Name(s)</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.1.81</td>
<td></td>
<td>192.168.1.81</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mi.mirror.garr.it</td>
<td></td>
<td>193.206.139.34</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>151.1.245.36</td>
<td></td>
<td>151.1.245.36</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.1.1</td>
<td></td>
<td>192.168.1.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>jake.unipi.it</td>
<td></td>
<td>131.114.21.22</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>151.1.185.60</td>
<td></td>
<td>151.1.185.60</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>151.1.185.65</td>
<td></td>
<td>151.1.185.65</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.1.1</td>
<td></td>
<td>192.168.1.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>all-systems.mcast.net</td>
<td></td>
<td>224.0.0.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>224.0.0.251</td>
<td></td>
<td>224.0.0.251</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>83.103.35.4</td>
<td></td>
<td>83.103.35.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**NOTE:**
- You can define new communities.
- Click here for more information about host and domain sorting.
- Bandwidth values are the percentage of the total bytes that ntop has seen on the interface. Hover the mouse to see the actual value (rounded to the nearest full percentage point). The total of the values will NOT be 100% as local traffic will be counted TWICE (once as sent and again as received).
- The SENT bandwidth is shown as and the RECEIVED bandwidth is shown as.

Report created on Sun May 11 15:47:35 2008 [ntop uptime: 16 sec]
What can ntop do for me?

- ntop has been created to solve a real monitoring problem (no planning, case studies, market analysis).
- By the time it has been extended to satisfy user requirements.
- Portable and platform neutral: deploy it wherever you want with the same look and feel.
- Minimal requirements to leverage its use.
- Suitable for monitoring both a LAN (default) and a WAN (don’t forget to configure ntop properly).
Who is using ntop products?
Traffic Measurement

• Data sent/received: Volume and packets, classified according to network/IP protocol.
• Multicast Traffic.
• TCP Session History.
• Bandwidth Measurement and Analysis.
• VLAN/AS traffic statistics.
• VoIP (SIP, Cisco SCCP) Monitoring.
Traffic Characterization and Monitoring

• Network Flows (user configurable)
• Protocol utilization (# req, peaks/storms, positive/negative repl.) and distribution.
• Network Traffic Matrix.
• ARP, ICMP Monitoring.
• Detection of many popular P2P protocols (Caida paper)
Network Optimization and Planning

- Passive network mapping: identification of Routers and Internet Servers (DNS, Proxy).
- Traffic Distribution (Local vs. Remote).
- Service Mapping: service usage (DNS, Routing).
- Network traffic map (Graphwiz)
Network Inventory [1/2]

• Identification of routers and internet servers (DNS, NFS, proxy)
• Resource, services and OS inventory.
• Unhealthy hosts.
### Network Inventory [2/2]

#### Local Hosts Characterization

<table>
<thead>
<tr>
<th>Host</th>
<th>Unhealthy Host</th>
<th>L2 Switch Bridge</th>
<th>Gateway</th>
<th>Printer</th>
<th>NTP/DNS Server</th>
<th>SMTP/POP/IMAP Server</th>
<th>Directory/FTP/HTTP Server</th>
<th>DHCP/WINS Server</th>
<th>DHCP Client</th>
<th>P2P</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.0.0</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host059-160</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host062-160</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host053-160</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host003-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host005-160</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host029-160</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host028-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dns03_abila.net</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dns02_abila.net</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dns01_abila.net</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host119-160</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host118-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host117-160</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host074-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host073-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host066-160</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host065-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host064-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host063-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>host062-160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Host Fingerprint

Based on http://ettercap.sourceforge.net/
### Host Health

<table>
<thead>
<tr>
<th>Data Rcvd Stats</th>
<th>0 %</th>
<th>Rem 100 %</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP vs. Non-IP Rcvd</td>
<td>IP 100 %</td>
<td>Non-IP 0 %</td>
</tr>
<tr>
<td>Sent vs. Rcvd Pkts</td>
<td>Sent 51.8 %</td>
<td>Rcvd 48.2 %</td>
</tr>
<tr>
<td>Sent vs. Rcvd Data</td>
<td>Sent 33.2 %</td>
<td>Rcvd 66.8 %</td>
</tr>
</tbody>
</table>

**Host Type**
- Name Server

**Historical Data**

#### Host Healthiness (Risk Flags)
- 

1. **Unexpected packets (e.g. traffic to closed port or connection reset):**
   - [Rcvd: rejected]
   - [Rcvd: port unreac]
   - [Rcvd: hostnet unreac]

### Host Traffic Stats

<table>
<thead>
<tr>
<th>Time</th>
<th>Tot. Traffic Sent</th>
<th>% Traffic Sent</th>
<th>Tot. Traffic Rcvd</th>
<th>% Traffic Rcvd</th>
</tr>
</thead>
<tbody>
<tr>
<td>11 AM</td>
<td>13.4 MB</td>
<td>74.7 %</td>
<td>26.6 MB</td>
<td>74.0 %</td>
</tr>
<tr>
<td>10 AM</td>
<td>4.5 MB</td>
<td>25.3 %</td>
<td>9.3 MB</td>
<td>26.0 %</td>
</tr>
<tr>
<td>9 AM</td>
<td>0</td>
<td>0.0 %</td>
<td>0</td>
<td>0.0 %</td>
</tr>
<tr>
<td>8 AM</td>
<td>0</td>
<td>0.0 %</td>
<td>0</td>
<td>0.0 %</td>
</tr>
</tbody>
</table>

---

**ntop.org**

**Università di Pisa**

Free / Libre / Open Source Software Conference
VoIP Support

<table>
<thead>
<tr>
<th>Client</th>
<th>Server</th>
<th>Data Sent</th>
<th>Data Rcvd</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>130.192.225.34 :8000</td>
<td>130.192.225.44 :32854</td>
<td>58.6 KB</td>
<td>70.3 KB</td>
<td>valter called livio</td>
</tr>
<tr>
<td>130.192.225.34 :8001</td>
<td>130.192.225.44 :32855</td>
<td>224</td>
<td>146</td>
<td></td>
</tr>
<tr>
<td>stun01.sipphone.com :3478</td>
<td>130.192.225.34 :47575</td>
<td>216</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>130.192.225.34 :5060</td>
<td>bill.ipv6.polito.it :5060</td>
<td>2.8 KB</td>
<td>2.3 KB</td>
<td>valter called livio</td>
</tr>
<tr>
<td>130.192.225.44 :5060</td>
<td>bill.ipv6.polito.it :5060</td>
<td>4.5 KB</td>
<td>5.0 KB</td>
<td>valter called livio</td>
</tr>
<tr>
<td>130.192.225.44 :5060</td>
<td>130.192.225.34 :5060</td>
<td>462</td>
<td>361</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Host Type</th>
<th>VoIP Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>Known Users</td>
<td>stefano &lt;101&gt; [ VoIP ]</td>
</tr>
</tbody>
</table>
Integrating ntop Into Your Network

- You can use ntop with as a stand-alone application (via web) or as a traffic measurement server.
- Ntop can export traffic data in several ways:
  - Via the embedded SNMP agent (ntop MIB)
  - XML
  - RRD files
  - PHP/Perl/Python/JSON data export
- Ntop, by means of the rrd-alarm companion application, allow users to emit alarms based on some traffic conditions.
Introduction to Cisco NetFlow

• What is NetFlow? A Cisco-proprietary IP statistics collection feature that collects information on IP flows passing through a router.

• NetFlow Version 9 is a flexible and extensible means to carry NetFlow records from a network node to a collector.
Introduction to InMon sFlow

• Ntop is part of the sflow.or consortium.
• Similar to NetFlow: probes send traffic flows to collectors over UDP in sFlow format (RFC 3176).
• A sFlow probe is basically a sniffer that captures packets at X rate (1:400 is default) and sends them coded in sFlow format. The more flows are captured, the more precise are the statistics. Tuning sample rate allows probes to capture at Gb speeds and above.

• sFlow in a nutshell:
  • Embedded in every switch port
  • Monitors traffic flow for all network ports
  • Effective at gigabit speeds
  • Does not impact network performance
  • Continuous monitoring
  • Robust under all network conditions
  • All devices = L2 – L7 flows end-end
  • Real-time and historical, detailed data
Ntop and NetFlow/sFlow

- Ntop supports both NetFlow (v1/5/7/9)/IPFIX and sFlow (v2/5).
- Ntop collects flows on virtual interfaces user-defined.
- Multiple interfaces can be defined independently. Ntop can simultaneously monitor netflow and sflow and pcap in interfaces.
- All the various interfaces have the same look and feel with little differences mainly due to the lack of payload access (NetFlow) hence inability to support packet decode (e.g. for P2P detection).
NetFlow Monitoring: State of the Art

- Cisco NetFlow is a commercial standard for network monitoring and accounting.
- Many companies (e.g. Cisco, Juniper, Extreme) ship appliances with embedded NetFlow probes.
- Most commercial probes perform very poorly (~7-10’000 pkt/sec).
- Several collectors available (both commercial and Open Source).
- Very little offering in the probe side.
- NetFlow monitoring cannot cope with Gbit speeds and above hence new mechanisms (e.g. sampled NetFlow) have been used to overcome this problem.
Solution: nProbe+ntop

- The community needed an open source probe able to bring NetFlow both into small and large networks.
- Ability to run at wire speed (at least until 1 Gb) with no need to sample traffic.
- Complete open source solution for both flow generation (nProbe) and collection (ntop)
nProbe: Main Features

- Ability to keep up with Gbit speeds on Ethernet networks handling thousand of packets per second without packet sampling on commodity hardware.
- Support for major OS including Unix, Windows and MacOS X.
- Resource (both CPU and memory) savvy, efficient, designed for environments with limited resources.
- Source code available under GNU GPL.
- nProbe v4 new features:
  - Full NetFlow v9/IPFIX support
  - V9 extensions: payload, network/application latency, SIP/RTP.
  - Ability to extend the probe with user-written plugins.
- nProbe v5 will be released later this summer.
Packet Capture: Open Issues

- Monitoring low speed (100 Mbit) networks is already possible using commodity hardware and tools based on libpcap.

- Sometimes even at 100 Mbit there is some (severe) packet loss: we have to shift from thinking in term of speed to number of packets/second that can be captured analyzed.

- Problem statement: monitor high speed (1 Gbit and above) networks with common PCs (64 bit/66 Mhz PCI/X/Express bus) without the need to purchase custom capture cards or measurement boxes.

- Challenge: how to improve packet capture performance without having to buy dedicated/costly network cards?
Packet Filter Ring (PF_RING)

Application A

Socket (ring)
Read Index

mmap()
Outgoing Packets

Application Z

Socket (ring)
Write Index

Incoming Packets

Network Adapter

Userspace

Kernel

PF_RING
PF_RING: Benefits

- It creates a straight path for incoming packets in order to make them first-class citizens.
- No need to use custom network cards: any card is supported.
- Transparent to applications: legacy applications need to be recompiled in order to use it.
- No kernel or low-level programming is required.
- Developers familiar with network applications can immediately take advantage of it without having to learn new APIs.
## PF_RING: Performance Evaluation

<table>
<thead>
<tr>
<th>Pkt Size</th>
<th>Kpps</th>
<th>Mpps</th>
<th>% CPU Idle</th>
<th>Wire-Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>250</td>
<td>259.23</td>
<td>518</td>
<td>&gt; 90%</td>
<td>Yes</td>
</tr>
<tr>
<td>250</td>
<td>462.9</td>
<td>925.9</td>
<td>88%</td>
<td>Yes</td>
</tr>
<tr>
<td>128</td>
<td>355.1</td>
<td>363.6</td>
<td>86%</td>
<td>Yes</td>
</tr>
<tr>
<td>128</td>
<td>844.6</td>
<td>864.8</td>
<td>82%</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Test setup: pcount, full packet size, 3.2 GHz Celeron (single-core) - IXIA 400 Traffic Generator
PF_RING on Embedded Devices

http://nst.sourceforge.net/nst/docs/user/ch09s02.html
n2n: Private Overlay for Nw Administration

Motivation

• NAT devices mask the user’s IP identity and limit peers accessibility.
• No control over the connection configuration, totally managed by ISPs.
• Firewall greatly reduce the possibility of a user being contacted by a direct session opened elsewhere over the Internet.

Vision

• The internet should be a “transparent” IP-based transport for users, not a geographical/ISP constrain.
• Users should control/create their community networks (today network administrators do).
N2N Architecture
N2N Features

- A n2n network is an encrypted L2 P2P-VPN.
- Unlike Skype/Hamachi, encryption is performed on edge nodes using open protocols with user-defined encryption keys.
- n2n users can simultaneously belong to multiple networks.
- Ability to cross NAT and firewalls in the reverse traffic direction (i.e. from outside to inside) so that n2n nodes are reachable even if running on a private network.
- n2n networks are not meant to be self-contained, but it is possible to route traffic across n2n and non-n2n networks.
Conclusions

Over the past 10 years the ntop project has produced:

• Ntop: a mature passive traffic monitoring application able to be integrated into industrial environments.

• nProbe: a fast and extensible NetFlow probe able to use ntop as a central console and to measure traffic using NetFlow even on networks where there aren’t NetFlow-enabled routers.

• PF_RING: Linux packet capture acceleration able to run on embedded systems and high-speed SMP servers.

• n2n: layer 2, peer-to-peer VPN for remote system connectivity and administration.