



ntopng

A Web-based Network Traffic Monitoring Application

New York City, NY

June 14th, 2017

Simone Mainardi <mainardi@ntop.org>
[linkedin.com/in/simonemainardi](https://www.linkedin.com/in/simonemainardi)

Agenda

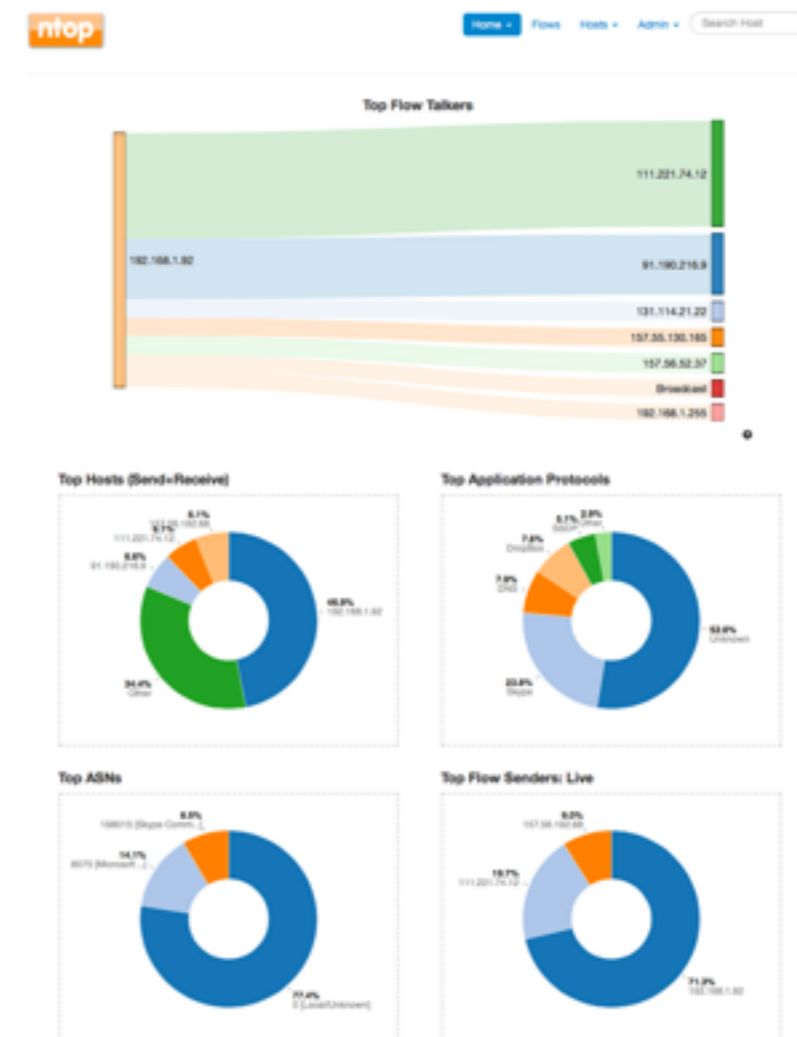
- About ntop
- Network traffic monitoring using ntopng
 - Motivation
 - ntopng architectural design and interfaces
- Integrations with third-party software
 - Grafana
 - Logstash
- Summary

About ntop



About ntop

- Private company devoted to development of Open Source network traffic monitoring applications.
- R&D Italy, Sales Switzerland.
- ntop (circa 1998) is the first app we released and it is a web-based network monitoring application.



Product Lines

- Open Source

- ntopng: Web-based monitoring application
- PF_RING: Accelerated RX/TX on Linux
- nDPI: Deep Packet Inspection Toolkit

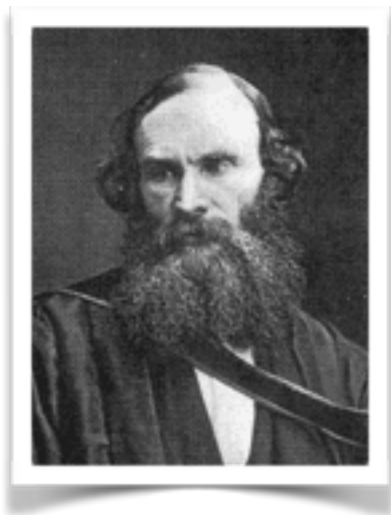
- Proprietary

- PF_RING ZC: 1/10/40/100 Gbit Line rate.
- nProbe: 10G NetFlow/IPFIX Probe
- nProbe Cento: flows+packets+security
- n2disk/disk2n Network-to-disk and disk-to-network.
- nScrub: Software DDoS Mitigation

Network Traffic Monitoring Using ntopng



“To measure is to know”



**“If you can not measure it,
you can not improve it”**

Lord William Thomson
(aka Lord Kelvin)

What Happens in Our Network?

- Do we have control over our network?
- It's not possible to imagine a healthy network without a clear understanding of traffic flowing on our network
- Knowledge is the first step towards evaluation of potential network security issues
- Event correlation can provide us timely information about our network health

Packets Never Lie

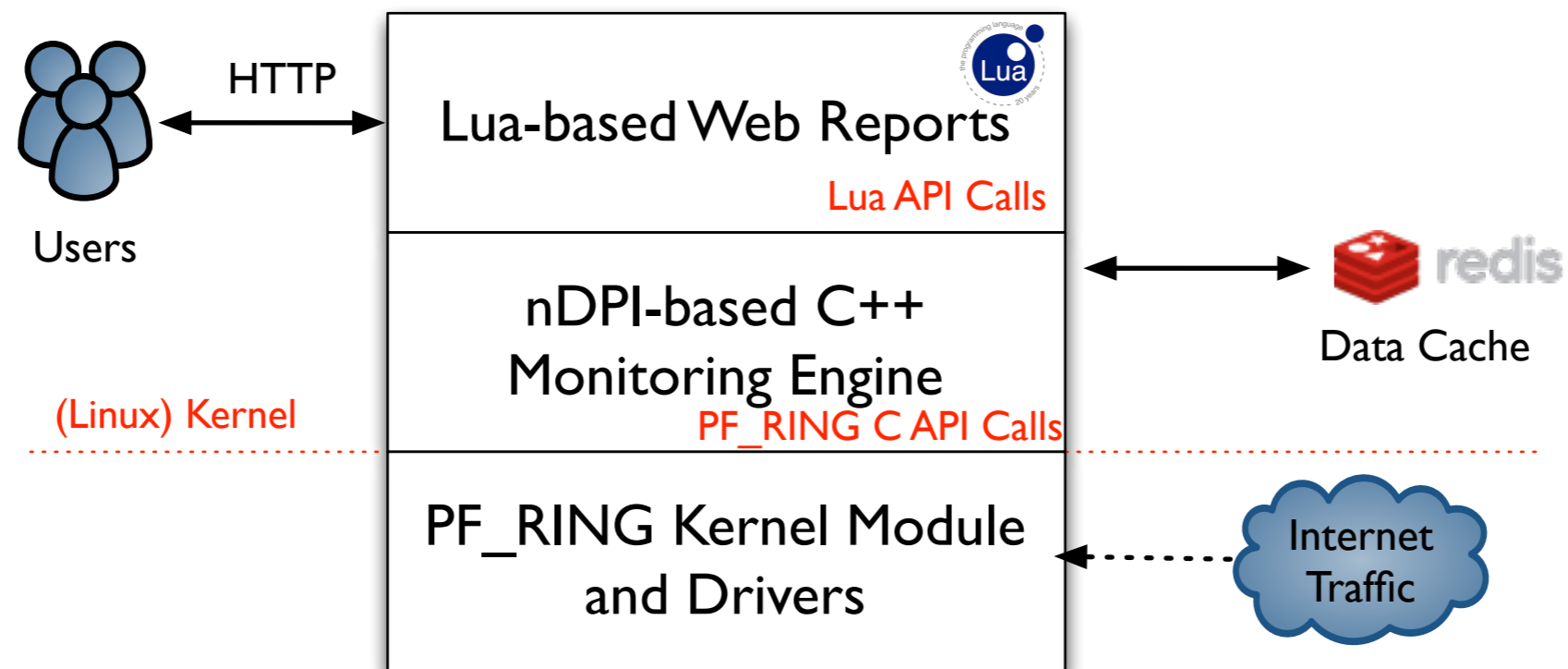
- Packet analysis provides useful information for understanding
 - Network traffic issues
 - Network usage not compliant with network policies (note: firewalls cannot help here)
 - Performances less than expected
 - Potential security flaws
 - Ongoing (latent) attacks
 - Data breach

Before We Start: ntopng Installation

- Source code
<https://github.com/ntop/ntopng>
- Binary Packages (stable and nightly)
<http://packages.ntop.org> (Debian, Ubuntu, CentOS, OSX, Raspbian (ARM), FreeBSD, Windows)

ntopng Architecture

- Three different and self-contained components, communicating with clean API calls.



Lua-based ntopng Scriptability [1/3]

- A design principle of ntopng has been the clean separation of the GUI from engine (in ntop it was all mixed)
- This means that ntopng can (also) be used (via HTTP) to feed data into third party apps such as Grafana and Nagios, just to name a few
- All data export from the engine happens via Lua
- Lua methods invoke the ntopng C++ API in order to interact with the monitoring engine

Lua-based ntopng Scriptability [2/3]

- `/scripts/callback/` scripts are executed periodically to perform specific actions.
- `/scripts/lua/` scripts are executed only by the web GUI.
- **Example:**

`http://ntopng:3000/lua/flow_stats.lua`

| Name | Date Modified | Size |
|------------------------------|-----------------------|-----------|
| ▼ callbacks | Sep 30, 2013 2:15 PM | -- |
| daily.lua | Apr 17, 2013 1:55 PM | 29 bytes |
| hourly.lua | Apr 17, 2013 1:55 PM | 29 bytes |
| minute.lua | Sep 30, 2013 2:15 PM | 5 KB |
| nprobe-collector.lua | Sep 30, 2013 2:15 PM | 4 KB |
| second.lua | Sep 30, 2013 2:15 PM | 2 KB |
| ▼ lua | Today 3:58 PM | -- |
| about.lua | Jun 30, 2013 10:27 PM | 2 KB |
| ▶ admin | Jun 26, 2013 11:24 PM | -- |
| aggregated_host_details.lua | Sep 30, 2013 2:15 PM | 6 KB |
| aggregated_host_stats.lua | Aug 15, 2013 4:37 PM | 442 bytes |
| aggregated_hosts_stats.lua | Sep 30, 2013 2:15 PM | 1 KB |
| db.lua | Aug 12, 2013 7:48 PM | 320 bytes |
| do_export_data.lua | Sep 30, 2013 2:15 PM | 765 bytes |
| export_data.lua | Sep 4, 2013 7:49 PM | 1 KB |
| find_host.lua | Sep 4, 2013 7:49 PM | 2 KB |
| flow_details.lua | Sep 30, 2013 2:15 PM | 7 KB |
| flow_stats.lua | Aug 15, 2013 4:37 PM | 1 KB |
| flows_stats.lua | Aug 15, 2013 4:37 PM | 2 KB |
| get_aggregated_host_info.lua | Aug 15, 2013 4:37 PM | 857 bytes |
| get_flows_data.lua | Sep 4, 2013 7:49 PM | 6 KB |
| get_geo_hosts.lua | Sep 4, 2013 7:49 PM | 2 KB |
| get_host_activitymap.lua | Sep 30, 2013 2:15 PM | 505 bytes |
| get_host_traffic.lua | Sep 4, 2013 7:49 PM | 399 bytes |
| get_hosts_data.lua | Sep 30, 2013 2:15 PM | 6 KB |
| get_hosts_interaction.lua | Sep 30, 2013 2:15 PM | 2 KB |

Lua-based ntopng Scriptability [3/3]

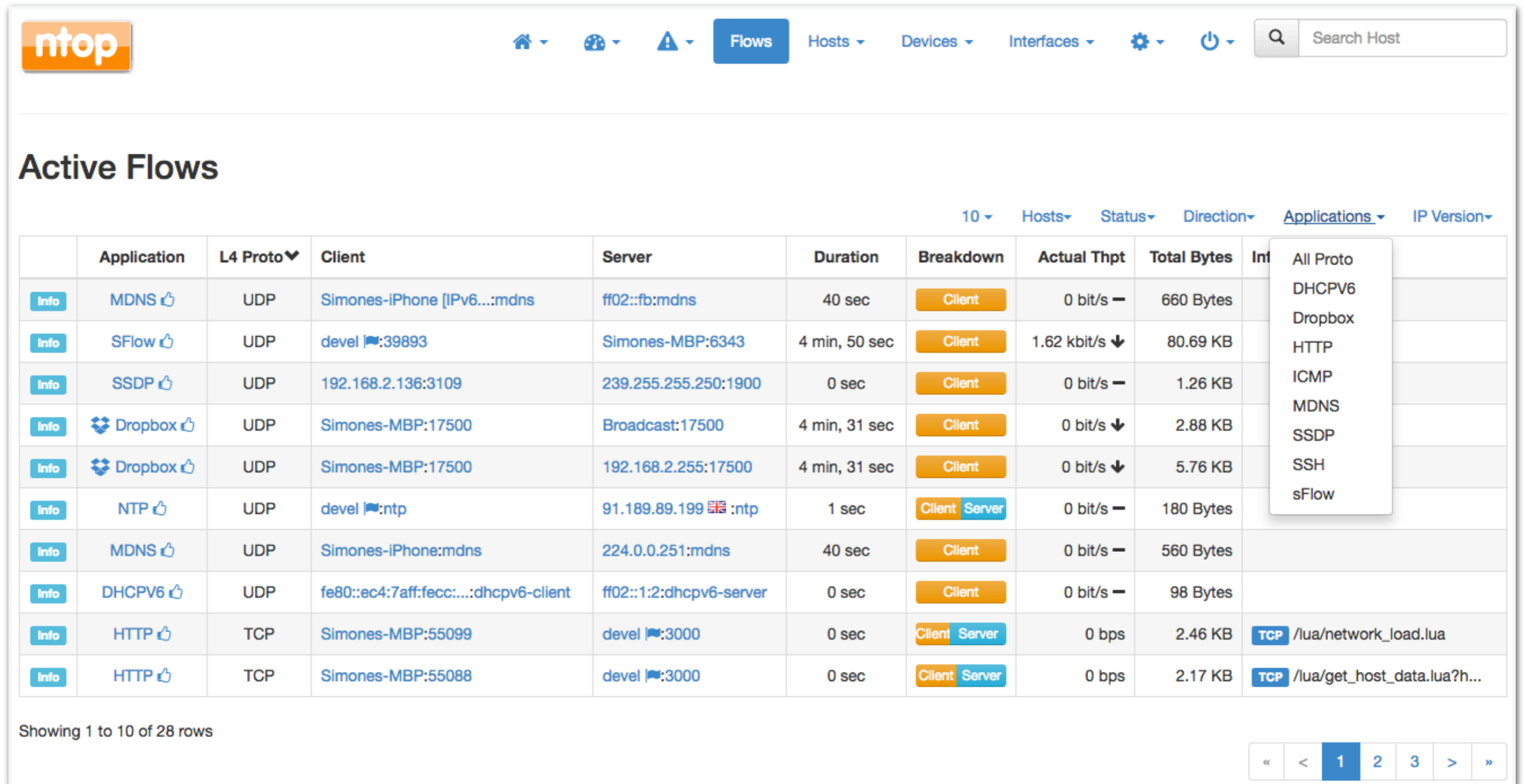
- ntopng defines (in C++) two Lua classes:
 - `interface`
 - Hook to objects that describe **flows** and **hosts**.
 - Access to live monitoring data.
 - `ntop`
 - General functions used to interact with ntopng configuration.
- **Examples**
 - `interface.getHostsInfo()`
 - `interface.getFlowsInfo()`

ntopng Interfaces



Web UI: Active Flows [1/2]

http://localhost:3000/lua/flows_stats.lua



The screenshot shows the ntop web interface. At the top, there is a navigation bar with the ntop logo, a search bar, and several menu items: Home, Alerts, Flows (selected), Hosts, Devices, Interfaces, Settings, and Power. Below the navigation bar, the main content area is titled "Active Flows". There are several dropdown menus for filtering: "10" (rows per page), "Hosts", "Status", "Direction", "Applications" (with a dropdown menu open), and "IP Version". The "Applications" dropdown menu is open, showing a list of protocols: All Proto, DHCPV6, Dropbox, HTTP, ICMP, MDNS, SSDP, SSH, and sFlow. The main table displays 10 rows of active flows. Each row includes an "Info" button, the Application name, L4 Protocol, Client and Server addresses, Duration, Breakdown (Client or Server), Actual Thpt, Total Bytes, and Interface. The last two rows show HTTP flows to the local host.

| | Application | L4 Proto | Client | Server | Duration | Breakdown | Actual Thpt | Total Bytes | Int |
|------|-------------|----------|---------------------------------------|-------------------------|---------------|---------------|-------------|-------------|---------------------------------|
| Info | MDNS | UDP | Simones-iPhone [IPv6...:mdns | ff02::fb:mdns | 40 sec | Client | 0 bit/s | 660 Bytes | |
| Info | SFlow | UDP | devel :39893 | Simones-MBP:6343 | 4 min, 50 sec | Client | 1.62 kbit/s | 80.69 KB | |
| Info | SSDP | UDP | 192.168.2.136:3109 | 239.255.255.250:1900 | 0 sec | Client | 0 bit/s | 1.26 KB | |
| Info | Dropbox | UDP | Simones-MBP:17500 | Broadcast:17500 | 4 min, 31 sec | Client | 0 bit/s | 2.88 KB | |
| Info | Dropbox | UDP | Simones-MBP:17500 | 192.168.2.255:17500 | 4 min, 31 sec | Client | 0 bit/s | 5.76 KB | |
| Info | NTP | UDP | devel :ntp | 91.189.89.199 :ntp | 1 sec | Client Server | 0 bit/s | 180 Bytes | |
| Info | MDNS | UDP | Simones-iPhone:mdns | 224.0.0.251:mdns | 40 sec | Client | 0 bit/s | 560 Bytes | |
| Info | DHCPV6 | UDP | fe80::ec4:7aff:fecc:...:dhcpv6-client | ff02::1:2:dhcpv6-server | 0 sec | Client | 0 bit/s | 98 Bytes | |
| Info | HTTP | TCP | Simones-MBP:55099 | devel :3000 | 0 sec | Client Server | 0 bps | 2.46 KB | TCP /lua/network_load.lua |
| Info | HTTP | TCP | Simones-MBP:55088 | devel :3000 | 0 sec | Client Server | 0 bps | 2.17 KB | TCP /lua/get_host_data.lua?h... |

Showing 1 to 10 of 28 rows

« < 1 2 3 > »

Web UI: Active Flows [2/2]

http://localhost:3000/lua/flow_details.lua

The screenshot displays the ntop web interface for an active flow. The top navigation bar includes the ntop logo, a search bar, and menu items for Home, Flows (selected), Hosts, Devices, Interfaces, Settings, and Power. The main content area shows the flow details for 'localhost:50269 ↔ localhost:3000' in an 'Overview' tab. The flow is identified as TCP/HTTP (7) and was first/last seen on 14/06/2017 at 16:01:36. Total traffic is 69.11 KB, with 37.91 KB (54.86%) in goodput. The flow is completed and will expire soon. The HTTP method is GET, the server name is localhost, and the response code is 200. A 'Dump Flow Traffic' checkbox is present at the bottom.

| | | |
|--|--|--|
| Flow Peers [Client / Server] | localhost 50269 ↔ localhost 3000 | |
| Protocol | TCP / HTTP (7) | |
| First / Last Seen | 14/06/2017 16:01:36 [5 sec ago] | 14/06/2017 16:01:36 [5 sec ago] |
| Total Traffic | Total: 69.11 KB | Goodput: 37.91 KB (54.86 %) |
| | Client → Server: 285 Pkts / 16.04 KB | Client ← Server: 285 Pkts / 53.07 KB |
| Application Latency | 41.224 ms | |
| Packet Inter-Arrival Time [Min / Avg / Max] | Client → Server: < 1 ms / < 1 ms / 41 ms | Client ← Server: < 1 ms / < 1 ms / 41 ms |
| TCP Flags | Client → Server: FIN SYN PUSH ACK | Client ← Server: FIN SYN PUSH ACK |
| Flow Status | Normal | |
| Actual / Peak Throughput | 0 bit/s / 0 bit/s | |
| HTTP | HTTP Method | GET |
| | Server Name | localhost |
| | URL | /lua/flows_stats.lua?app... |
| | Response Code | 200 |
| Dump Flow Traffic | <input type="checkbox"/> | |

Lua: Active Flows

All the flows currently active on interface eth0

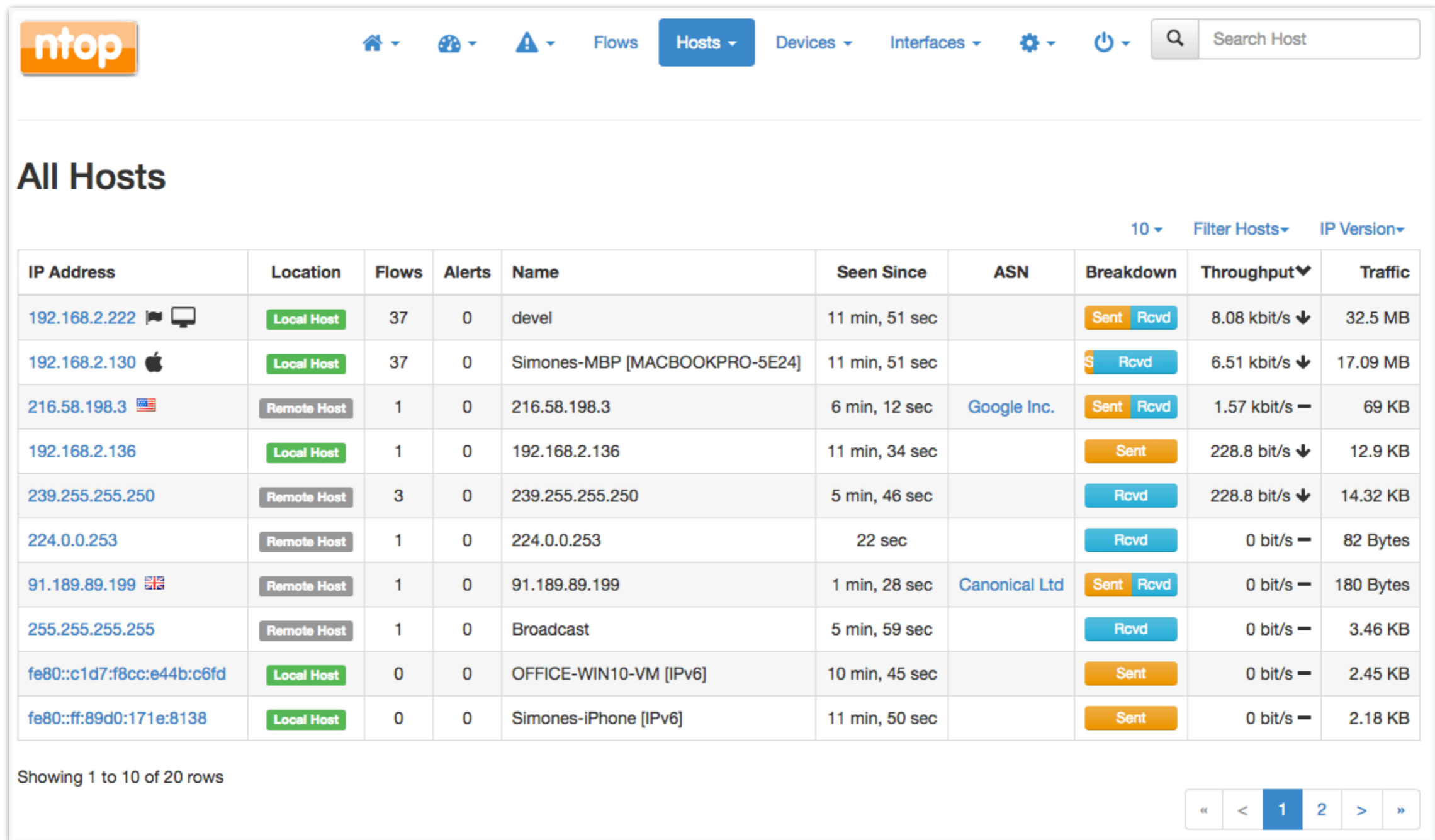
All the active flows having this host as one endpoint

Extra filtering criteria (e.g., application protocol, sort order, number of results)

```
interface.select("eth0")
local flows = interface.getFlowsInfo("192.168.1.2", pageinfo)
flows = flows["flows"]
local total = 0
local not_established = 0
for i, fl in ipairs(flows) do
    if fl["proto.14"] == "TCP" then
        total = total + 1
        if not fl["tcp_established"] then
            not_established = not_established + 1
        end
    end
end
end
if not_established / total > 0.1 then
    tprint("Too many flows not established")
end
```

Web UI: Active Hosts [1/2]

http://localhost:3000/lua/hosts_stats.lua



The screenshot shows the ntop web interface. At the top, there is a navigation bar with the ntop logo, a search bar labeled "Search Host", and several menu items: Home, Alerts, Flows, Hosts (selected), Devices, Interfaces, Settings, and Power. Below the navigation bar, the page title is "All Hosts". There are controls for "10" items per page, "Filter Hosts", and "IP Version". The main content is a table with the following columns: IP Address, Location, Flows, Alerts, Name, Seen Since, ASN, Breakdown, Throughput, and Traffic. The table lists 10 hosts, including local hosts like "devel" and "Simones-MBP", and remote hosts like "216.58.198.3" (Google Inc.) and "91.189.89.199" (Canonical Ltd). At the bottom, it says "Showing 1 to 10 of 20 rows" and has a pagination control showing page 1 of 2.

| IP Address | Location | Flows | Alerts | Name | Seen Since | ASN | Breakdown | Throughput | Traffic |
|---------------------------|-------------|-------|--------|-------------------------------|----------------|---------------|-----------|---------------|-----------|
| 192.168.2.222 | Local Host | 37 | 0 | devel | 11 min, 51 sec | | Sent Rcvd | 8.08 kbit/s ↓ | 32.5 MB |
| 192.168.2.130 | Local Host | 37 | 0 | Simones-MBP [MACBOOKPRO-5E24] | 11 min, 51 sec | | S Rcvd | 6.51 kbit/s ↓ | 17.09 MB |
| 216.58.198.3 | Remote Host | 1 | 0 | 216.58.198.3 | 6 min, 12 sec | Google Inc. | Sent Rcvd | 1.57 kbit/s — | 69 KB |
| 192.168.2.136 | Local Host | 1 | 0 | 192.168.2.136 | 11 min, 34 sec | | Sent | 228.8 bit/s ↓ | 12.9 KB |
| 239.255.255.250 | Remote Host | 3 | 0 | 239.255.255.250 | 5 min, 46 sec | | Rcvd | 228.8 bit/s ↓ | 14.32 KB |
| 224.0.0.253 | Remote Host | 1 | 0 | 224.0.0.253 | 22 sec | | Rcvd | 0 bit/s — | 82 Bytes |
| 91.189.89.199 | Remote Host | 1 | 0 | 91.189.89.199 | 1 min, 28 sec | Canonical Ltd | Sent Rcvd | 0 bit/s — | 180 Bytes |
| 255.255.255.255 | Remote Host | 1 | 0 | Broadcast | 5 min, 59 sec | | Rcvd | 0 bit/s — | 3.46 KB |
| fe80::c1d7:f8cc:e44b:c6fd | Local Host | 0 | 0 | OFFICE-WIN10-VM [IPv6] | 10 min, 45 sec | | Sent | 0 bit/s — | 2.45 KB |
| fe80::ff:89d0:171e:8138 | Local Host | 0 | 0 | Simones-iPhone [IPv6] | 11 min, 50 sec | | Sent | 0 bit/s — | 2.18 KB |

Web UI: Active Hosts [2/2]

http://localhost:3000/lua/host_details.lua

The screenshot shows the ntop web interface for host details. The top navigation bar includes the ntop logo, a search bar, and menu items for Home, Flows, Hosts (selected), Devices, Interfaces, Settings, and Power. Below the navigation bar, the host IP is 192.168.2.222, and various tabs like Traffic, Packets, Ports, Peers, ICMP, Protocols, Activity, DNS, HTTP (with a notification), Flows, SNMP, and Talkers are visible. The main content area displays a table of host details:

| | | |
|--|---|--|
| (Router/AccessPoint) MAC Address | SuperMic_D4:CC:F9 (00:25:90:D4:CC:F9) | |
| Host SNMP Localization ↻ NOTE: Hosts are located in SNMP devices using the Bridge MIB . | SNMP Device | Device Port |
| | 192.168.2.169 | 3 Idx 3 |
| IP Address | 192.168.2.222 ↻ [192.168.2.0/24] | Host Pool: office ⚙ |
| Name | devel ↗ Local Host System IP 🖥 | |
| First / Last Seen | 18/05/2017 12:01:51 [12 min, 23 sec ago] | 18/05/2017 12:14:11 [3 sec ago] |
| Sent vs Received Traffic Breakdown | <div style="display: flex; justify-content: space-around;"><div style="width: 50%; background-color: orange; text-align: center;">Sent</div><div style="width: 50%; background-color: blue; text-align: center;">Rcvd</div></div> | |
| Traffic Sent / Received | 22,823 Pkts / 16.02 MB ↑ | 33,267 Pkts / 16.69 MB ↑ |
| Active Flows / Total Active / Low Goodput | 'As Client' | 'As Server' |
| | 35 - / 157 - / 0 - | 37 - / 1,144 ↑ / 0 - |
| TCP Packets Sent Analysis | Retransmissions | 6 Pkts - |
| | Out of Order | 3 Pkts - |
| | Lost | 0 Pkts - |
| Further Host Names/Information | 192.168.2.222 | |
| JSON | 📄 Download | |

Lua:Active Hosts

```
function getTopInterfaceHosts (howmany)
  local hosts_stats = interface.getHostsInfo ()
  hosts_stats = hosts_stats["hosts"]
  local ret = {}
  local sortTable = {}

  for k,v in pairs(hosts_stats) do
    sortTable[k] = v["bytes.sent"] + v["bytes.rcvd"]
  end

  local n = 0
  for _v,k in pairsByValues(sortTable, rev) do
    if(n < howmany) then
      ret[_v] = hosts_stats[_v]
      n = n+1
    else
      break
    end
  end

  return(ret)
end
```



All the hosts currently active on the selected interface

ntopng

Data Integrations with Third-Party Tools



Supported Third-Party Tools

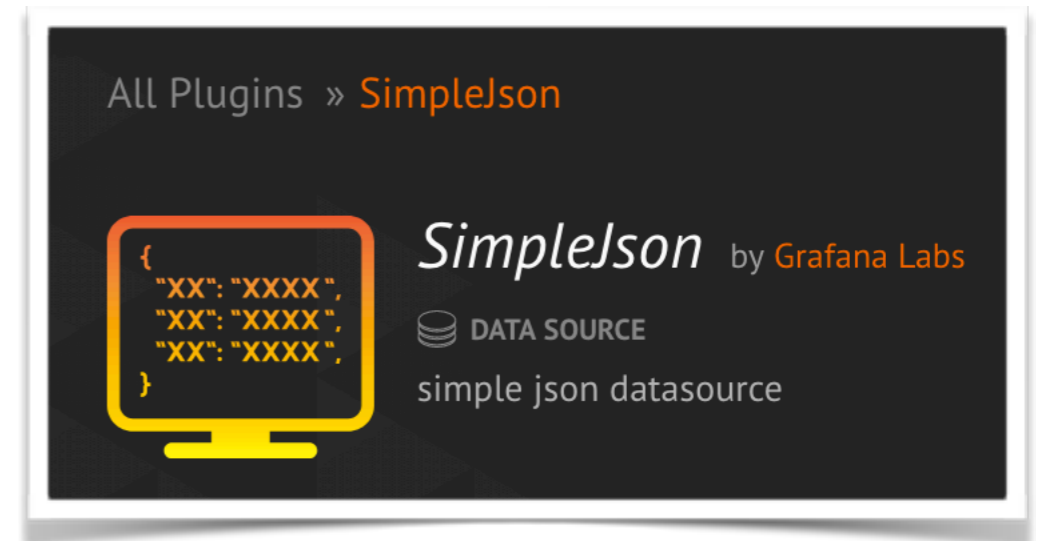
- Timeseries
 - RRDs
 - Grafana
 - Prometheus (wip)
- Flows
 - MySQL/MariaDB
 - ElasticSearch
 - Logstash



- Platform for monitor (and alert on) metrics
- Storage-platform agnostic
 - Abstractions with Datasource plugins
- Flexible data visualizations
 - Custom (exportable) dashboards
 - Reusable dashboard building blocks (Panel plugins)
- Support for 30+ storage platforms
 - Influxdb
 - Prometheus
 - SNAP
 - etc.

ntopng + Grafana

- ntopng exposes interface and host metrics to Grafana
 - Throughput (bps and pps)
 - Application protocols (Facebook, Youtube, etc)
- Compatible with Simplejson plugin by Grafana Labs
- Simplejson plugin extended to support basic authentication
 - Fork: <https://github.com/simonemainardi/simple-json-datasource>
 - PR: <https://github.com/grafana/simple-json-datasource/pull/66>



ntopng Grafana Exposed Metrics

- **Traffic**

- interface_<interface name>_traffic_bps
- interface_<interface name>_traffic_total_bytes
- interface_<interface name>_traffic_pps
- interface_<interface name>_traffic_total_packets
- host_<host ip>_interface_<interface name>_traffic_bps
- host_<host ip>_interface_<interface name>_traffic_total_bytes

- **Interface Layer-7 Application Protocols**

- interface_<interface_name>_allprotocols_bps
- host_<host ip>_interface_<interface_name>_allprotocols_bps

- **Examples**

- interface_eth0_allprotocols_bps
- host_192.168.1.2_interface_eth0_traffic_bps

Grafana ntopng Datasource

Just a name to identify the datasource

Datasource plugin is SimpleJson

http endpoint is
<ntopng host>:<port>/lua/modules/grafana

A user/password pair as created in ntopng

The screenshot shows the 'Edit data source' configuration page in Grafana. The interface is dark-themed. At the top, there's a search bar and a 'Data Sources' dropdown menu. The main content area is titled 'Edit data source'. Below the title, there are several sections:

- Name:** 'ntopngDatasource' with an information icon and a 'Default' checkbox that is checked.
- Type:** 'SimpleJson' with a dropdown arrow.
- Http settings:**
 - Url:** 'http://127.0.0.1:3000/lua/modules/grafana' with an information icon.
 - Access:** 'direct' with a dropdown arrow and an information icon.
- Http Auth:**
 - Basic Auth:** A radio button that is checked.
 - With Credentials:** A radio button that is unchecked.
- Basic Auth Details:**
 - User:** 'admin'
 - Password:** '.....'

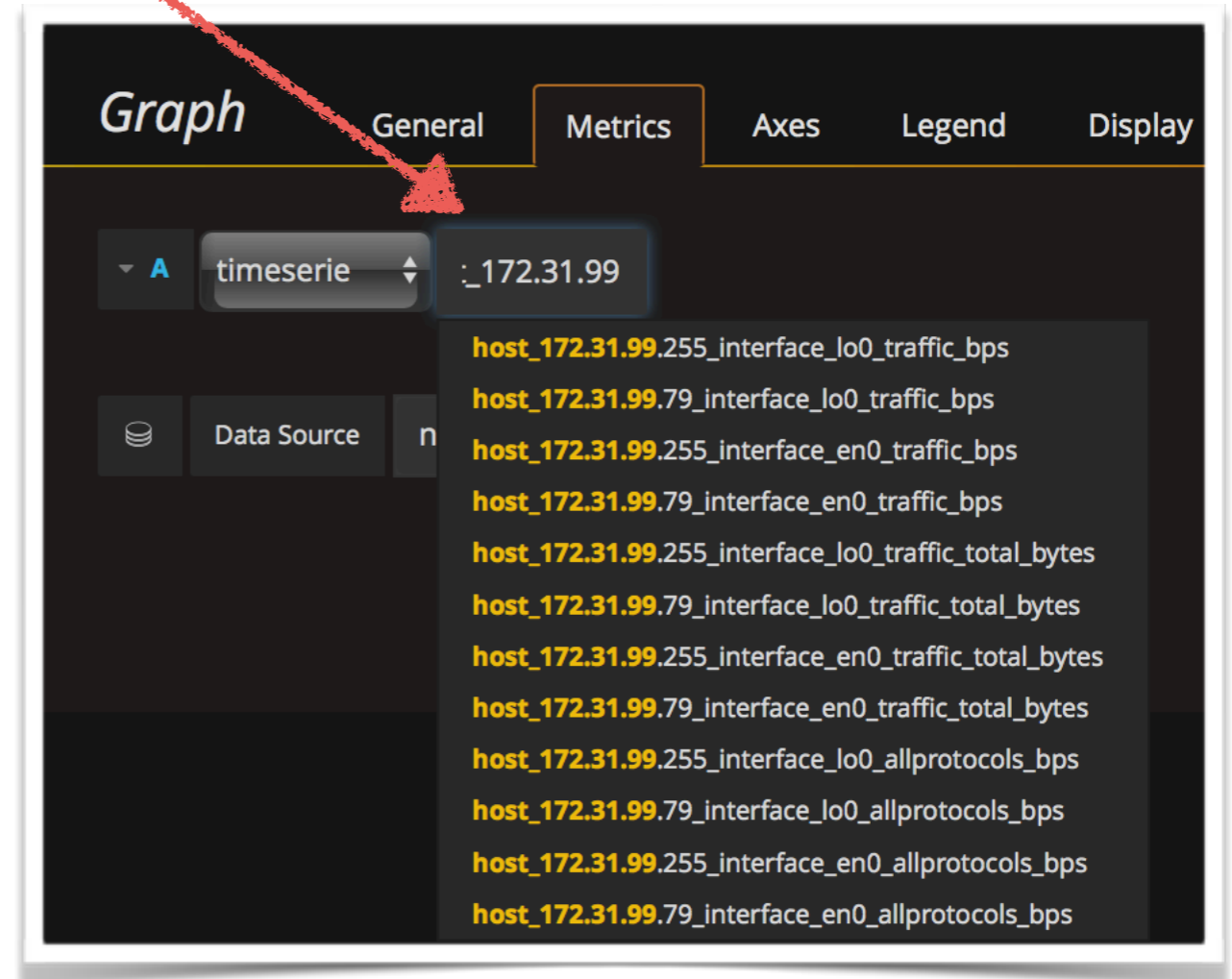
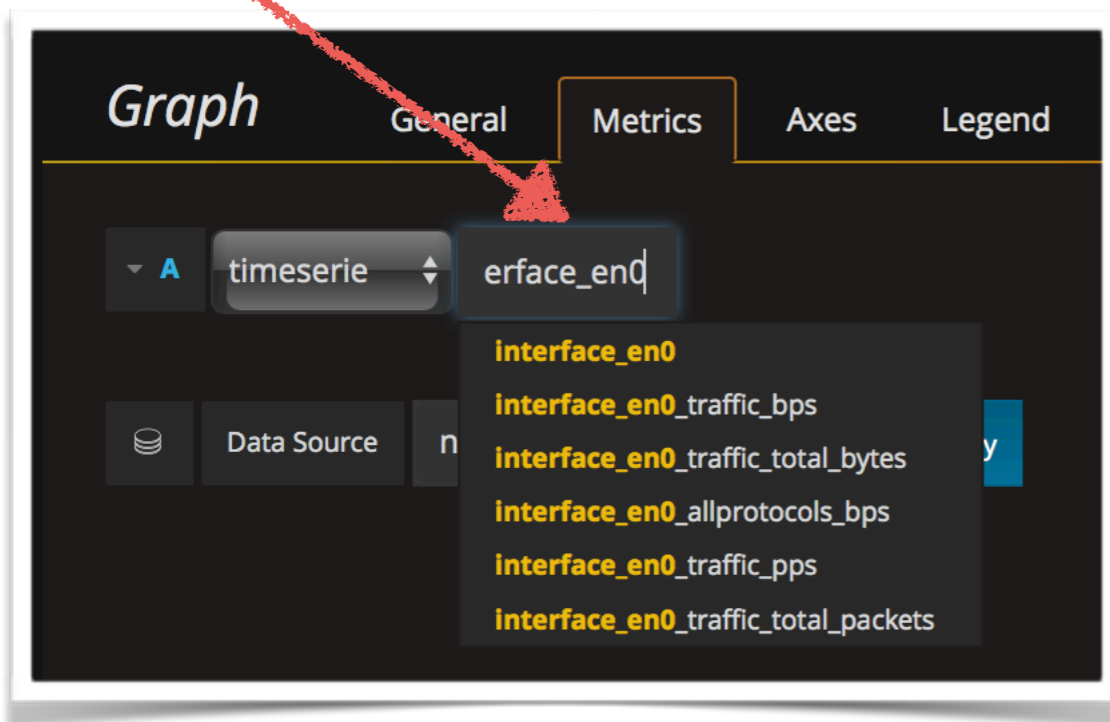
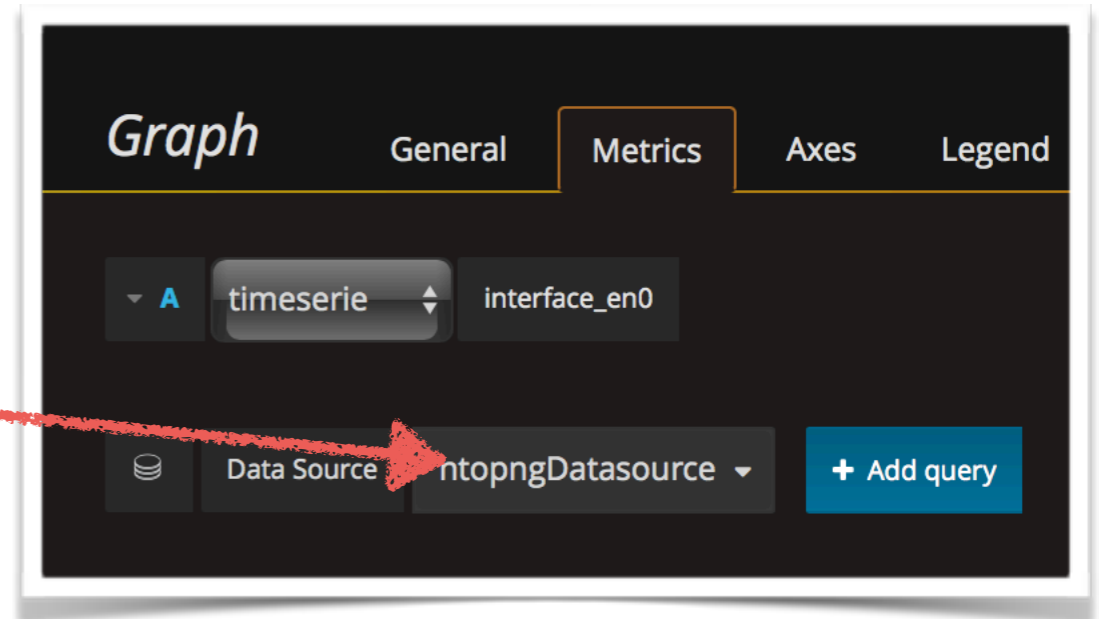
At the bottom, there is a green success message: 'Success Data source is working'. Below that are three buttons: 'Save & Test' (green), 'Delete' (orange), and 'Cancel' (grey).

Red arrows from the text boxes on the left point to the corresponding fields in the screenshot: 'Name', 'Type', 'Url', and 'User'.

Grafana Dashboard Graph Panel

Select the ntopng datasource

Start typing to get the list of available metrics



Setup of a Traffic Dashboard Graph Panel

Graph General Metrics **Axes** Legend Display Alert Time range

Left Y Show

Unit |

- packets/sec
- bits/sec
- bytes/sec
- kilobits/sec
- kilobytes/sec
- megabits/sec
- megabytes/sec
- gigabytes/sec
- gigabits/sec

Right Y Show

X-Axis Show

Mode Time

Pick the metric and set the proper unit of measure (bps == bits/sec)

Graph General **Metrics** Axes Legend

timeserie interface_en0_traffic_bps

Data Source ntopngDatasource + Add query

Graph General Metrics Axes Legend Display Alert Time range

Info

| | |
|-------------|--|
| Title | Traffic |
| Description | The traffic rate, expressed in bits per second, of the network interface |

Dimensions

| | |
|-------------|--------------------------|
| Span | 12 |
| Height | 100px |
| Transparent | <input type="checkbox"/> |

Drilldown / detail link ?

+ Add link

Set a name and a description for the chart

A Traffic Dashboard Graph Panel



Interface: en0



Packets

Protocols

Chart

Flows

Talkers

Protocols

Timeseries



Timeframe:

1m

5m

10m

1h

3h

6h

12h

1d

1w

2w

1M

6M

1Y

Hi
si

8.68 Mbit/s

8 Mbit/s

7 Mbit/s

6 Mbit/s

5 Mbit/s

4 Mbit/s

3 Mbit/s

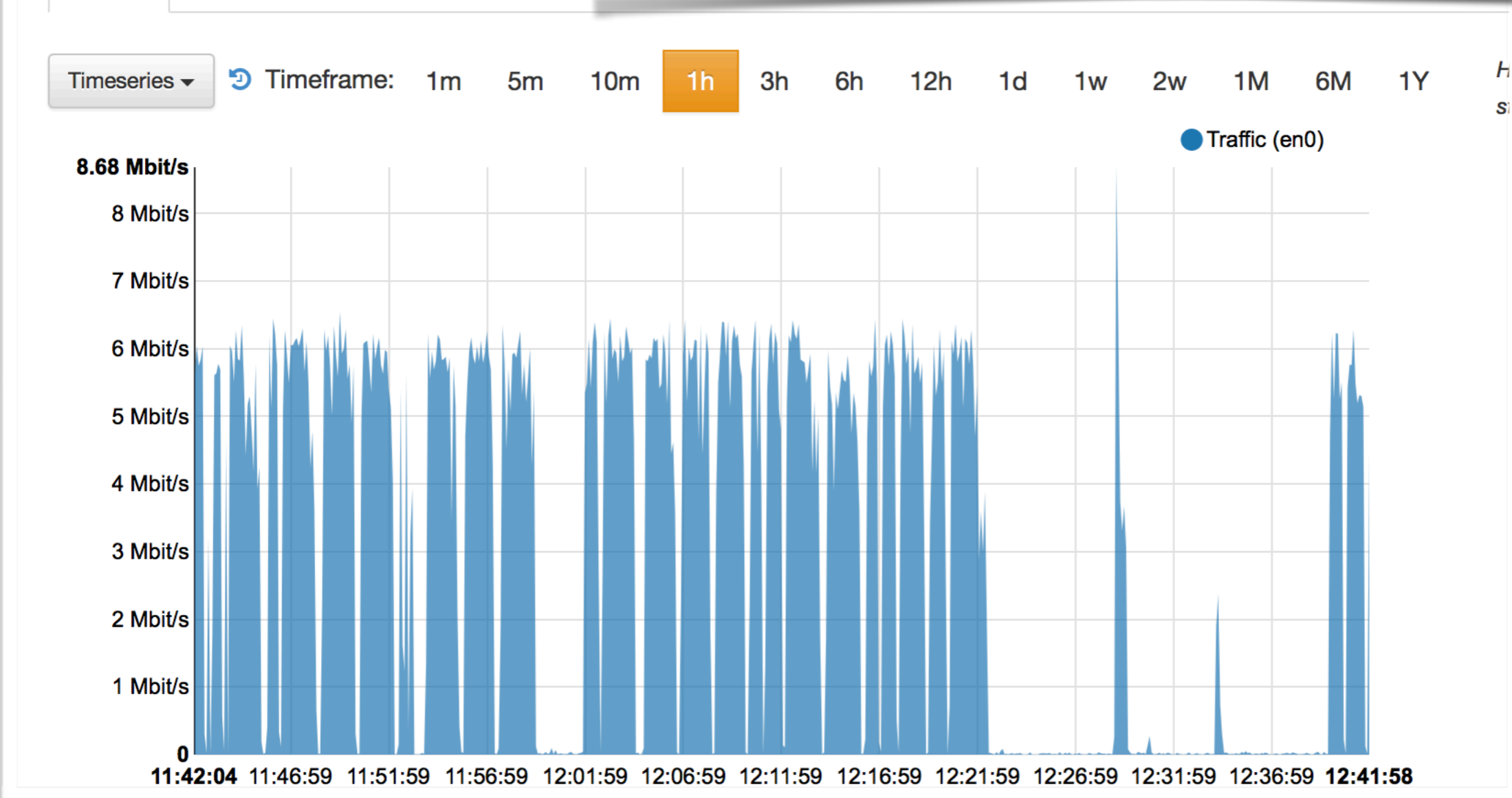
2 Mbit/s

1 Mbit/s

0

● Traffic (en0)

11:42:04 11:46:59 11:51:59 11:56:59 12:01:59 12:06:59 12:11:59 12:16:59 12:21:59 12:26:59 12:31:59 12:36:59 12:41:58



Setup of an All Protocols Traffic Graph Panel

Graph General Metrics **Axes** Legend Display Alert

Left Y Right Y

Show

Unit |

Scale none

Y-Min currency

Label data (IEC)

data (Metric)

data rate

throughput

length

velocity

volume

energy

packets/s

bits/sec

bytes/sec

kilobits/sec

kilobytes/sec

megabits/sec

megabytes/sec

gigabytes/sec

gigabits/sec

Graph General Metrics **Axes** Legend Display Alert Time range

timeserie interface_en0_all

interface_en0_allprotocols_bps

Data Source ntopngDatasource + Add query

Pick the metric and set the proper unit of measure (bps == bits/sec)

Stack the timeseries to have a more meaningful y-axis

Graph General Metrics Axes Legend **Display** Alert Time range

Draw options

Series overrides (0)

Thresholds (0)

Draw Modes

Bars

Lines

Points

Mode Options

Fill 9

Line Width 1

Staircase

Hover tooltip

Mode All series

Sort order None

Stacked value individual

Stacking & Null value

Stack

Percent

Null value null

Stack the timeseries to have a more meaningful y-axis

An All Protocols Traffic Graph Panel



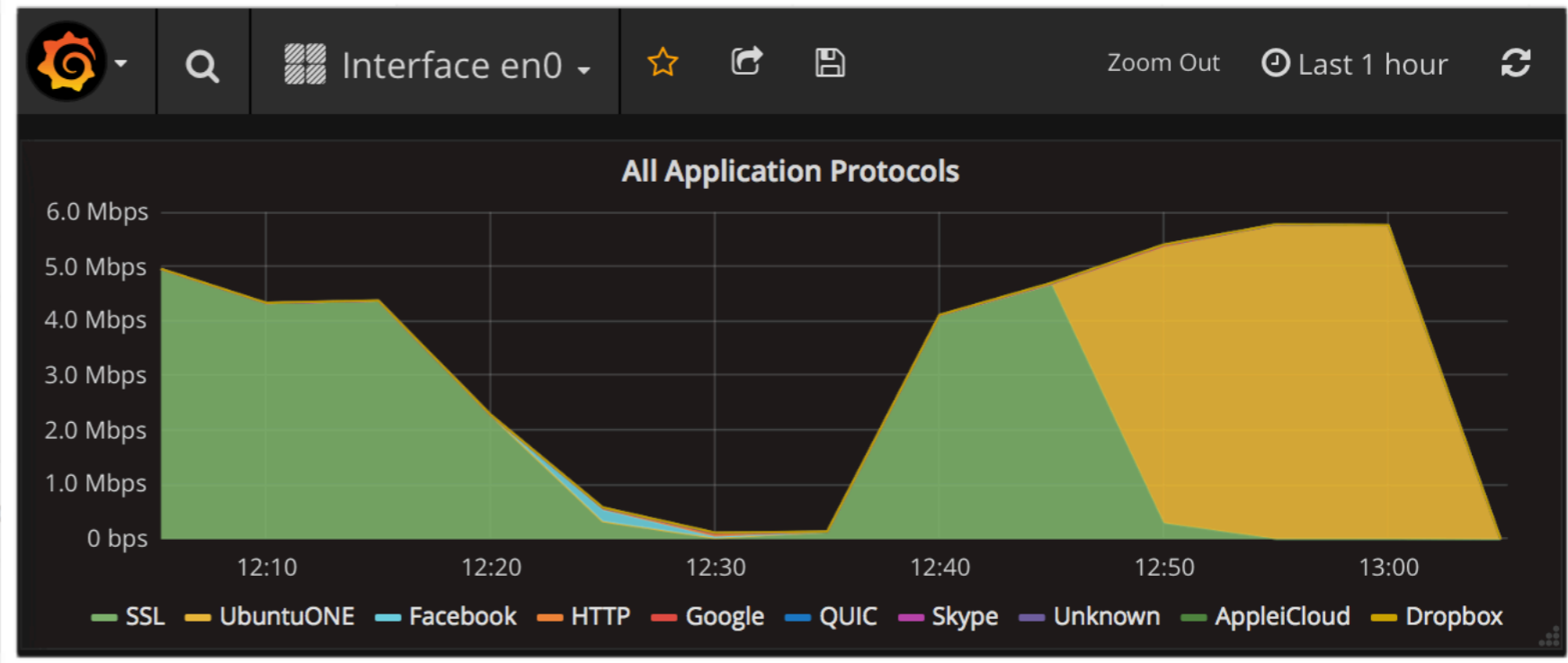
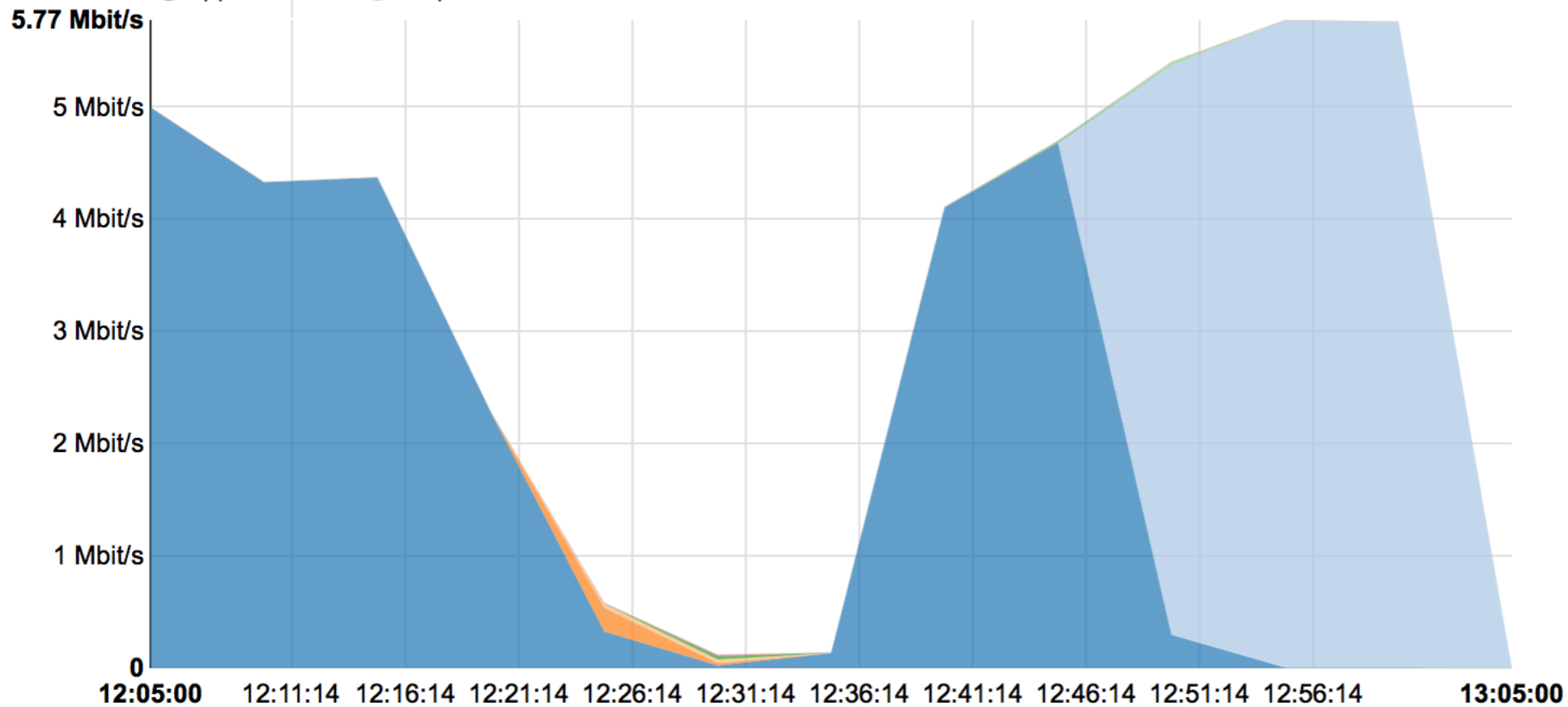
Interface: en0 ⌂ Packets Protocols

Chart Flows Talkers Protocols

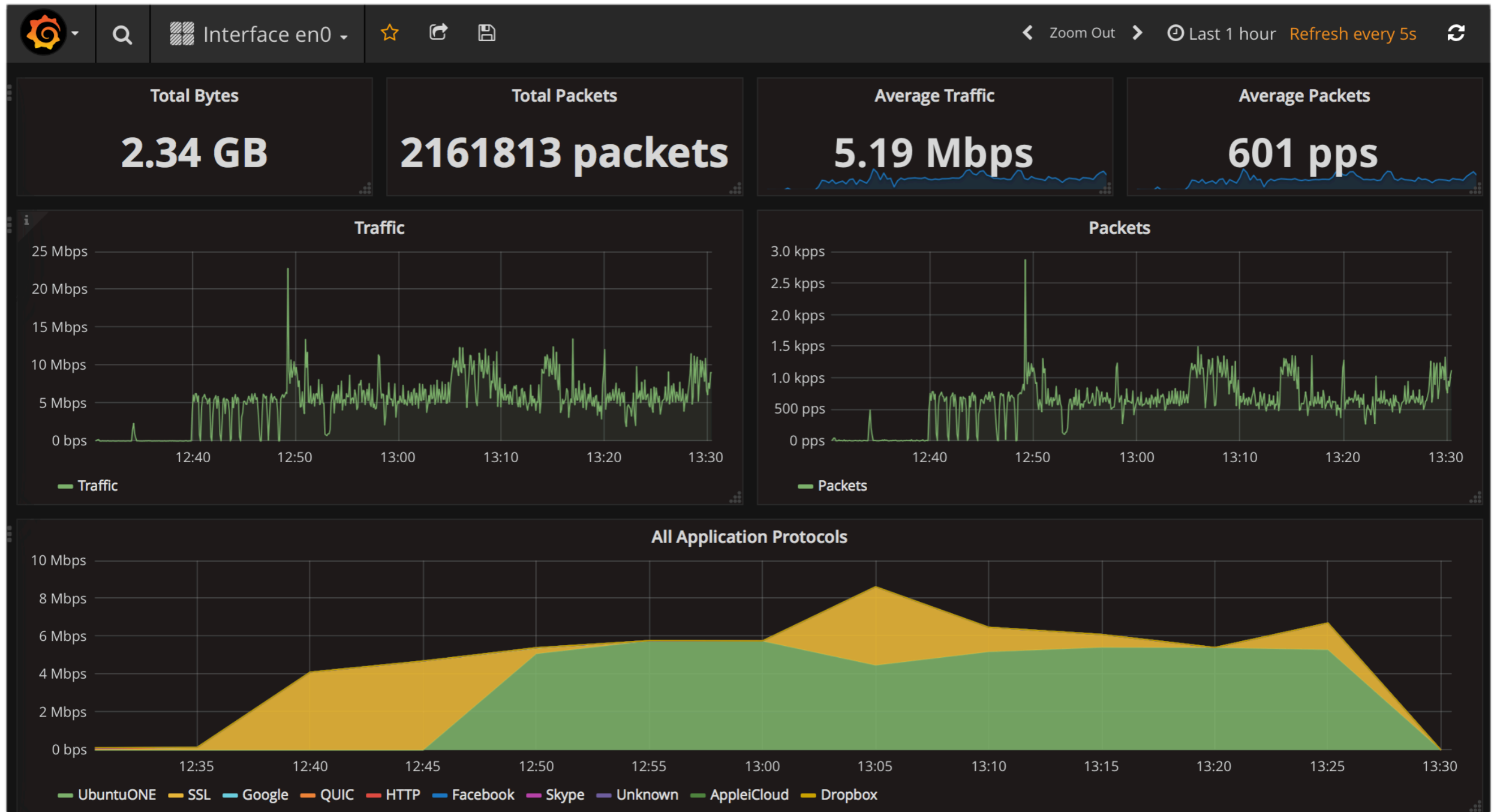
Timeseries ▾

Timeframe: 5m 10m **1h** 3h 6h 12h 1d 1w 2w 1M 6M 1Y

- SSL
- UbuntuONE
- Facebook
- HTTP
- Google
- QUIC
- Skype
- Unknown
- AppleiCloud
- Dropbox



A Full ntopng Grafana Dashboard





- Pipeline to ingest, parse, enrich, transform, and stash data
- Flexible ingestion thanks to the input plugins
 - Kafka
 - Graphite
 - Varnish
 - Syslog
 - etc.
- Enrichments and aggregations thanks to filter and codec plugins
 - Pattern matching
 - Geolocation
- Downstream route of data
 - ElasticSearch
 - HDFS
 - MongoDB
 - etc.

ntopng + logstash

- ntopng asynchronously delivers network flows to logstash
 - Source and destination ip address
 - Source and destination ports
 - Layer-7 application protocol
 - etc.
- Data encoded as JSON

A Basic Logstash Pipeline Configuration

```
input {  
  tcp {  
    host => "192.168.20.42"  
    port => 5510  
    codec => json  
    type => "ntopng-ls"  
  }  
}
```

Expected protocol

Listen address and port

Expected data encoding

```
# The filter part of this file is commented out to indicate that it is  
# optional.  
# filter {  
#  
# }
```

```
output {  
  if [type] == "ntopng-ls" {  
    stdout { codec => rubydebug }  
  }  
}
```

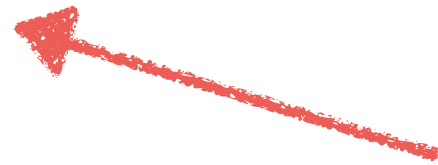
Just a dummy write-to-stdout

ntopng + Logstash: Starting Logstash

```
/usr/share/logstash$ sudo ./bin/logstash \  
-f /home/simone/logstash.conf \  
--path.settings /etc/logstash
```



Pipeline configuration file



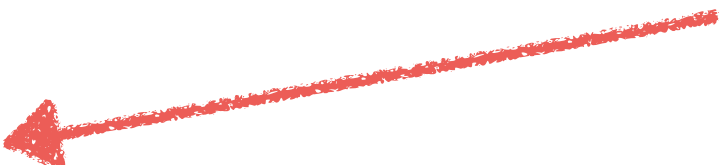
Path to the other settings files (logstash.yml, jvm.options, etc)

```
netstat -polenta | grep 5510  
tcp6          0          0 192.168.20.42:5510      :::*  
LISTEN        0          195348      9822/java              off (0.00/0/0)
```

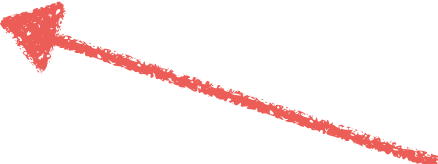
ntopng + Logstash: Starting ntopng

```
ntopng -i en0 -i lo0 \  
-F"logstash;192.168.20.42;tcp;5510"
```

Monitored interfaces



Logstash address, protocol,
and port



Logstash Output Data

```
{
    "type" => "ntopng-ls",
    "L7_PROTO" => 7,
    "OUT_PKTS" => 16,
    "HTTP_URL" => "/lua/modules/grafana/query",
    "NTOPNG_INSTANCE_NAME" => "Simones-MBP",
    "INTERFACE" => "lo0",
    "@version" => "1",
    "host" => "192.168.20.39",
    "TCP_FLAGS" => 27,
    "CLIENT_NW_LATENCY_MS" => 0.009,
    "HTTP_RET_CODE" => 200,
    "PROTOCOL" => 6,
    "LAST_SWITCHED" => 1497300991,
    "IPV4_DST_ADDR" => "127.0.0.1",
    "OUT_BYTES" => 3617,
    "L7_PROTO_NAME" => "HTTP",
    "HTTP_HOST" => "127.0.0.1",
    "L4_DST_PORT" => 3000,
    "SERVER_NW_LATENCY_MS" => 0.035,
    "@timestamp" => 2017-06-12T20:56:39.074Z,
    "ntop_timestamp" => "2017-06-12T20:56:31.0Z",
    "IPV4_SRC_ADDR" => "127.0.0.1",
    "IN_PKTS" => 16,
    "IN_BYTES" => 1880,
    "port" => 63212,
    "L4_SRC_PORT" => 63321,
    "HTTP_METHOD" => "POST",
    "FIRST_SWITCHED" => 1497300991
}
```

Take Home

- ntopng is Lua-scriptable network monitoring tool
 - Continuously monitors hosts and flows
 - Saves/exports metrics and flows
- Several third-party integrations including
 - Grafana
 - Logstash
 - Prometheus and SNAP under development
- Scriptability and ability to communicate over a network pave the way for visually unlimited third-party integrations