summarizing traffic for observability and DDoS mitigation

Shannon Weyrick  •  VP Research @ NS1  •  Office of CTO
sweyrick@ns1.com
pktvisor.com • IDS 2021
1. pktvisor in 15 mins

2. Deeper Dive

3. The Future: Orb
pktvisor in 15 mins
What is pktvisor?

- Open Source observability Agent
- *Taps into* pcap and (soon) DNSTAP streams
- *Summarizes* critical data from streams
- Provides both *Local* and *Global* visibility
What is pktvisor not?

- A full packet capture system
- A query audit log system
- A database
- Resource heavy
Why pktvisor?

- Deep L7 analysis with streaming algorithms
- Not based on flow/sampling
- Small data, big information
pktvisor extracts signal and produces summaries

- “Signal” is critical Net and DNS info
- Summarizes into live + 1 minute buckets
- JSON output is ~4kb per bucket
- ...regardless of input throughput!
DNS signal extraction

L7 Dissection
- Top Queries
- Top Query Types
- Top Result Codes
- Rate Percentiles
- Top Sources
- Counters

1 minute summaries

RAW TEXT END
DNS signal extraction

Low DNS Traffic

Top Queries
Top Query Types
Top Result Codes
Rate Percentiles
Top Sources
Counters
...

1 minute summaries

flow

pktvisor.com ∙ IDS 2021

Shannon Weyrick ∙ sweyrick@ns1.com
DNS signal extraction

DDoS Traffic

Information we may need to mitigate an attack!

Top Queries
Top Query Types
Top Result Codes
Rate Percentiles
Top Sources
Counters
...

1 minute summaries

1 minute summaries

flow
Command Line UI (think “dns top”) Updates display once / sec
<table>
<thead>
<tr>
<th>Top QName 2</th>
<th>Top QName 3</th>
<th>Top RCode</th>
<th>Top SRVFAILS</th>
<th>Top REFUSED</th>
<th>Top GeoLoc</th>
<th>Top ASN</th>
</tr>
</thead>
<tbody>
<tr>
<td>.google.com</td>
<td>.com.akadns.net</td>
<td>20 ( 4.80)</td>
<td>db.dns-sd._udp.0.1.168.192.in-addr.arpa</td>
<td>IPv4 192.168.0.189</td>
<td>1318 (76.2%)</td>
<td>1320 (76.38)</td>
</tr>
<tr>
<td>.apple.com</td>
<td>.192.in-addr.arpa</td>
<td>18 ( 4.38)</td>
<td>tmsns-cequntvzwidml.com</td>
<td>IPv4 192.168.0.114</td>
<td>222 (12.8%)</td>
<td>15169/GOOGLE</td>
</tr>
<tr>
<td>.akadns.net</td>
<td>.play.google.com</td>
<td>18 ( 4.38)</td>
<td>lmsns-cequntvzwidml.com</td>
<td>IPv4 35.190.20.61</td>
<td>1318 (76.2%)</td>
<td>15169/GOOGLE</td>
</tr>
<tr>
<td>.googleapis.com</td>
<td>weather.data.apple.com</td>
<td>12</td>
<td>db.dns-sd._udp.0.1.168.192.in-addr.arpa</td>
<td>IPv6 ff02::1:2</td>
<td>22 ( 1.18)</td>
<td>21342/Akamai International B.V.</td>
</tr>
<tr>
<td>.in-addr.arpa</td>
<td>.fe.apple-dns.net</td>
<td>12</td>
<td>db.dns-sd._udp.0.1.168.192.in-addr.arpa</td>
<td>IPv6 ff02::12</td>
<td>22 ( 1.18)</td>
<td>8075/MICROSOFT-CORP-MSN-AS-BLOCK</td>
</tr>
<tr>
<td>.microsoft.com</td>
<td>calendar.google.com</td>
<td>10</td>
<td>db.dns-sd._udp.0.1.168.192.in-addr.arpa</td>
<td>IPv6 35.224.179.84</td>
<td>10</td>
<td>41231/Canonical Group Limited</td>
</tr>
<tr>
<td>.office.com</td>
<td>.g.apoling.com</td>
<td>10</td>
<td>db.dns-sd._udp.0.1.168.192.in-addr.arpa</td>
<td>IPv6 35.190.20.61</td>
<td>10</td>
<td>8068/MICROSOFT-CORP-MSN-AS-BLOCK</td>
</tr>
</tbody>
</table>

**DNS Wire Pkts**: 416 (24.0%) | **Rates Total**: 0/0/0/0/0 | **UDP 416 (100.0%)** | **TCP 0 (0.0%)** | **IPV4 413 (99.3%)** | **IPV6 3 (0.7%)** | **Query 211 (50.7%)** | **Response 205 (49.3%)** | **DNS Xacts 205** | **Timed Out 2** | **In 101 (49.3%)** | **Out 104 (50.7%)** | **In 18.2/44.1/419.4 ms** | **Out 19.3/78.9/110.9/243.9 ms** | **Q Name Card 115** | **DNS NOERROR 185 ** | **SRVFAIL 0 (0.0%)** | **NXDOMAIN 20 (9.8%)** | **REFUSED 0 (0.0%)** | **Time Window 4:35PM to 4:40PM** | **Period 296s**

- **A**: 310 (74.5%) | **HTTPS**: 52 (12.5%) | **PTR**: 38 (9.1%) | **AAA**: 12 | **SOA**: 4
- **IPv4**: 192.168.0.189, 192.168.0.114, 35.190.20.61, 91.189.88.185, 239.255.255.250, 216.239.32.10, 35.224.179.84
- **IPv6**: ff02::1:2, ff02::12, 35.190.20.61, 91.189.88.185

---

pktvisor.com ∙ IDS 2021
How many unique IPs have been seen in the time window?
<table>
<thead>
<tr>
<th>Qname Card.</th>
<th>115</th>
</tr>
</thead>
</table>

**How many unique Qnames have been seen in the time window?**

### Top Qnames

<table>
<thead>
<tr>
<th>Qname</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>google.com</td>
<td>48</td>
<td>11.5%</td>
</tr>
<tr>
<td>.com.akadsn.com</td>
<td>4</td>
<td>1.0%</td>
</tr>
<tr>
<td>.in-addr.arpa</td>
<td>3</td>
<td>0.7%</td>
</tr>
<tr>
<td>.network</td>
<td>2</td>
<td>0.5%</td>
</tr>
</tbody>
</table>

### Top QTypes

<table>
<thead>
<tr>
<th>Type</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>310</td>
<td>74.5%</td>
</tr>
<tr>
<td>HTTPS</td>
<td>52</td>
<td>12.5%</td>
</tr>
<tr>
<td>PTR</td>
<td>38</td>
<td>9.1%</td>
</tr>
<tr>
<td>AAAA</td>
<td>12</td>
<td>2.8%</td>
</tr>
<tr>
<td>SOA</td>
<td>4</td>
<td>1.0%</td>
</tr>
</tbody>
</table>

### Top RCodes

<table>
<thead>
<tr>
<th>RCode</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOERROR</td>
<td>185</td>
<td>90.2%</td>
</tr>
<tr>
<td>NXDOMAIN</td>
<td>20</td>
<td>9.8%</td>
</tr>
</tbody>
</table>

### Top SRVFAILS

<table>
<thead>
<tr>
<th>SRVFAIL</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>_dns-sd._udp.0.1.168.192.in-addr.arpa</td>
<td>3</td>
<td>0.2%</td>
</tr>
<tr>
<td>_msns-cequentvzwiml.com</td>
<td>3</td>
<td>0.2%</td>
</tr>
<tr>
<td>_msns-cequentvzwiml.com</td>
<td>3</td>
<td>0.2%</td>
</tr>
<tr>
<td>_msns-cequentvzwiml.com</td>
<td>3</td>
<td>0.2%</td>
</tr>
</tbody>
</table>

### Top REFUSED

<table>
<thead>
<tr>
<th>REFUSED</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>_a</td>
<td>1175</td>
<td>67.9%</td>
</tr>
<tr>
<td>_a</td>
<td>118</td>
<td>6.8%</td>
</tr>
<tr>
<td>_a</td>
<td>108</td>
<td>6.1%</td>
</tr>
</tbody>
</table>

### Top GeoLoc

<table>
<thead>
<tr>
<th>Country/Region</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown</td>
<td>1318</td>
<td>76.2%</td>
</tr>
<tr>
<td>NA/United States</td>
<td>222</td>
<td>12.8%</td>
</tr>
<tr>
<td>NA/United States/CA/Mountain View</td>
<td>21342</td>
<td>2.1%</td>
</tr>
<tr>
<td>EU/United Kingdom/ENG/London</td>
<td>25</td>
<td>0.2%</td>
</tr>
</tbody>
</table>

### Top ASN

<table>
<thead>
<tr>
<th>Entity</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1320</td>
<td>1320</td>
<td>76.3%</td>
</tr>
<tr>
<td>15169/GOOGLE</td>
<td>154</td>
<td>8.9%</td>
</tr>
<tr>
<td>21342/Akamai International B.V.</td>
<td>21342</td>
<td>2.1%</td>
</tr>
<tr>
<td>8875/MICROSOFT-CORP-MSN-AS-BLOCK</td>
<td>30</td>
<td>0.2%</td>
</tr>
</tbody>
</table>

### DNS Wire

- Pkts: 1730 | UDP: 443 (25.6%) | TCP: 1229 (71.0%) | Other: 58 (3.4%) | IPv4: 1666 (96.3%) | IPv6: 6 (0.3%) | In: 848 (51.0%) | Out: 816 (49.0%) | Deep Samples: 1730 (100.0%) |

### DNS Xacts

- 205 (100.0%) | Timed Out: 2 | In: 101 (50.7%) | Out: 104 (50.7%) | In: 18.2/84.4/134.1/194.4 ms | Out: 19.3/78.9/110.9/243.9 ms | Time Window: 4:35PM to 4:40PM, Period: 296s
<table>
<thead>
<tr>
<th>Top QName 2</th>
<th>Top QName 3</th>
<th>Top QName 4</th>
<th>Top QName 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>.google.com</td>
<td>.com.akadns.net</td>
<td>db_dns_sd_udp.0.1.168.192.in-addr.arpa</td>
<td>db_dns_sd_udp.0.1.168.192.in-addr.arpa</td>
</tr>
<tr>
<td>.apple.com</td>
<td>.in-addr.arpa</td>
<td>tmsns.cequintvzwidml.com</td>
<td>lb_dns_sd_udp.0.1.168.192.in-addr.arpa</td>
</tr>
<tr>
<td>.akadns.net</td>
<td>.weather-data.arpa</td>
<td>play.google.com</td>
<td>lb_dns_sd_udp.0.1.168.192.in-addr.arpa</td>
</tr>
<tr>
<td>.googleapis.com</td>
<td>.home-apple-dns.net</td>
<td>192.in-addr.arpa</td>
<td>lb_dns_sd_udp.len</td>
</tr>
<tr>
<td>.in-addr.arpa</td>
<td>.calendar.google.com</td>
<td>.g.aapling.com</td>
<td>lb_dns_sd_udp.0.253.16.172.in-addr.arpa</td>
</tr>
<tr>
<td>.microsoft.com</td>
<td>.office.com</td>
<td>db_dns_sd_udp.0.253.16.172.in-addr.arpa</td>
<td>db_dns_sd_udp.0.253.16.172.in-addr.arpa</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Top QTypes</th>
<th>Top RCodes</th>
<th>Top SRVFAILS</th>
<th>Top REFUSED</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>NOERROR</td>
<td>185 (90.9%)</td>
<td>IPv4: 192.168.0.189</td>
</tr>
<tr>
<td>HTTPS</td>
<td>NXDOMAIN</td>
<td>20 (9.8%)</td>
<td>192.168.0.114</td>
</tr>
<tr>
<td>PTR</td>
<td>38 (9.1%)</td>
<td>1175 (67.9%)</td>
<td>35.190.20.61</td>
</tr>
<tr>
<td>AAAA</td>
<td>12</td>
<td>118 (6.8%)</td>
<td>91.188.98.185</td>
</tr>
<tr>
<td>SOA</td>
<td>4</td>
<td>108 (6.2%)</td>
<td>239.255.255.250</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Top GeoLoc</th>
<th>Top ASN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown</td>
<td>1318 (76.2%)</td>
</tr>
<tr>
<td>NA/United States</td>
<td>222 (12.8%)</td>
</tr>
<tr>
<td>NA/United States/CA/Mountain View</td>
<td>21342/Akamai International B.V.</td>
</tr>
<tr>
<td>NA/United States/CA/Mountain View</td>
<td>21342/Akamai International B.V.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Top DNS UDP Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>5353</td>
</tr>
<tr>
<td>53839</td>
</tr>
<tr>
<td>15061</td>
</tr>
<tr>
<td>9066</td>
</tr>
<tr>
<td>63047</td>
</tr>
<tr>
<td>61078</td>
</tr>
<tr>
<td>64187</td>
</tr>
</tbody>
</table>
Central Collection

- Provides a *Global* view of distributed Agents
- Metric database agnostic
- Tools for Prometheus and Elasticsearch
- Grafana Dashboard
Central Collection

- Database agnostic
- Scrape or Push
- Small Data
Grafana Dashboard: Prometheus

DNS Packets (In+Out):

DNS Transactions:

DNS Errors:

Top DNS Names:

Top QName:

pktvisor.com ∙ IDS 2021
3.2.0
weyrick released this on Apr 16

New Features

- Introduce native Prometheus support into pktvisor with `--prometheus` flag, which will expose Prometheus compatible metrics at `/metrics` endpoint. Also see `--prom-instance`
- Add a new docker container for easily collecting and sending Prometheus compatible metrics, see docker hub
- Add a new Grafana dashboard for Prometheus, both to the repo and to Grafana dashboard community
- Begin building and distributing an AppImage (static Linux binary) which includes pktvisor, pktvisor-cli, and pktvisor-pcap
- Ability to daemonize pktvisor with the `-d` flag
- Ability to send pktvisor logs to either an output file (`--log-file`), or to syslog (`--syslog`)

Other Improvements

- CI and build improvements including better use of Conan and automatic dependency installation
- Improved documentation and READMEs

Bug Fixes

- #447 Fix live rates in pktvisor-cli

Assets 3

pktvisor-x86_64-3.2.0.Appimage 8.96 MB
Source code (zip)
Easy Install

pull the image

root@dnshost:~$ docker pull ns1labs/pktvisor

start the agent

root@dnshost:~$ docker run --net=host -d ns1labs/pktvisor pktvisorord eth0

run the command line UI

root@dnshost:~$ docker run -it --rm --net=host ns1labs/pktvisor pktvisor-cli

Shannon Weyrick ∙ sweyrick@ns1.com
download the binary, make executable

root@dnshost:~$ curl -L http://pktvisor.com/download -o pktvisor-x86_64.AppImage
root@dnshost:~$ chmod +x pktvisor-x86_64.AppImage

start the agent

root@dnshost:~$ sudo ./pktvisor-x86_64.AppImage pktvisord eth0

run the command line UI

root@dnshost:~$ ./pktvisor-x86_64.AppImage pktvisor-cli
Easily Plugin To Prometheus

pktvisor + centralized Prometheus collection

This container combines pktvisor with the Grafana Agent for collecting and sending metrics to Prometheus through remote write, including to cloud providers like Grafana Cloud.

There is a sample Grafana dashboard which provides a good starting point for visualizing pktvisor metrics. You can also find it online via the Grafana community dashboards, allowing you to import easily into any Grafana installation (ID 14221).

Example:

docker pull nulllabs/pktvisor-prom-write

docker run -d --net=host --env PKTVISOR_ARGS="--prom-instance <INSTANCE> --INTERFACE " \--env REMOTE_URL="https://<REMOTEHOST>/api/prom/push" --env USER_NAME="<USERNAME>" \--env PASSWORD="<PASSWORD>" nulllabs/pktvisor-prom-write

Example with Geo enabled (assuming files are located in /usr/local/geo):  

docker pull nulllabs/pktvisor-prom-write

docker run -d --mount type=bind,source=/usr/local/geo,target=/geo --net=host --env PKTVISOR_ARGS="--prom-instance <INSTANCE> --geo-city /geo/GeoPZ-City.mdb --geo-asn /geo/GeoPZ-IISP.mdb --INTERFACE " \--env REMOTE_URL="https://<REMOTEHOST>/api/prom/push" --env USER_NAME="<USERNAME>" --env PASSWORD="<PASSWORD>" nulllabs/pktvisor-prom-write

There are several pieces of information you need to substitute above:

- `<INSTANCE>`: The prometheus “instance” label for all metrics, e.g. “myhost”
- `<INTERFACE>`: The ethernet interface to capture on, e.g. “eth0”
- `<REMOTEHOST>`: The remote host to remote write the prometheus metric to
- `<USERNAME>`: If required by your prometheus setup, the user name to connect. If not required, leave off this environment variable.
- `<PASSWORD>`: If required by your prometheus setup, the password to connect. If not required, leave off this environment variable.

Other pktvisor arguments may be passed in the PKTVISOR_ARGS environment variable.
Easily Plugin To Elasticsearch

Metrics Collection

Metrics from the REST API

The metrics are available from the agent in JSON format via the REST API.

For most use cases, you will want to collect the most recent full 1-minute bucket, once per minute:

```
curl localhost:10853/api/v1/metrics/bucket/1
```

This can be done with tools like telegraf and the standard HTTP plugin. Example telegraf config snippet:

```
[inputs]
[[inputs.http]]
  urls = ["http://127.0.0.1:10853/api/v1/metrics/bucket/1"],
  interval = "50s"
  data_format = "json"
  json_query = "1m"
  json_time_key = "period_start_ts"
  json_time_format = "unix"
  json_string_fields = ["dns_*", "packets_*"]

[inputs.http.tags]
t = "pktvisor"
  interval = "50s"
```
Install Grafana Dashboard

All dashboards » pktvisor - prometheus

pktvisor - prometheus by ns1labs

A dashboard for pktvisor observability tool (https://github.com/ns1labs/pktvisor), showcasing Network and DNS metrics.
Last updated: a month ago

Start with Grafana Cloud and the new FREE tier. Includes 10K series Prometheus or Graphite Metrics and 50gb Loki Logs

pktvisor summarizes network data streams in real time. It can capture Network, DNS, and other metrics via packet capture, dnstap, sflow, and other input methods.

This dashboard can be used as a starting point to visualize pktvisor metrics. See the Github page for information on how to deploy and collect these metrics.
Deeper Dive
History

- pktvisor v1 2014 (forked netsniff-ng, remains open source)
- operations, debugging, DDoS visibility
- essentially simple DNS “top”
- deficiencies
  - central collection was a hack
  - resource usage
  - missing IPv6 and TCP support
  - did not track transactions (query/reply pair)
Rewrite

- move to Agent paradigm
- fix deficiencies
- modularize: inputs, dissectors, analyzers, sinks
- parallelize
- summarize with stream processing techniques (DataSketches)
- API first: built-in HTTP control plane
Sliding time window, JSON interface

- maintain mergeable 1m buckets of metrics to provide summary across full window
- always-on Agent supplies information to CLI UI and central collection via HTTP
- both merged and individual buckets are available for collection in REST API
  - CLI UI uses the merged window
  - Central collector gathers a single minute, once a minute
Under The Hood

- agent written in modern C++
- CLI UI is written in Go
- PcapPlusPlus abstraction for pcap input + custom AF_PACKET
- Apache Data Sketches
- optional MaxMind support for GeoIP and ASN
- HTTP(S) API, JSON + native Prometheus output
- Linux, OSX. Windows?
Data Sketches

- fast, probabilistic data structures designed for streaming
- results are approximate but within well defined error bounds
- provide cardinality, heavy hitters (frequent items), quantiles
- designed to be merged, which is how we support time window
- possible to expose raw binary sketch data via API so that it can be merged across hosts and data centers
The Future: Orb

IoT Inspired Cloud Control Plane for Fleet of pktvisor Agents
Orb

1. Agent fleet
2. IoT control plane
3. Data sinks
Orb

1. Agent fleet
2. IoT control plane
3. Data sinks
IoT Control Plane

API based configuration management

Agents connect via MQTT
Exploring Edge Data with Dynamic Datasets

- apply multiple layered policies per Agent to extract different dimensions of Signal
- separate datasets for each policy
- filter out unwanted upstream data
- choose which summary data to collect
- choose where to send the data (built-in TSDB, S3 bucket, etc)
Orb Project Goals

- open source, vendor neutral, cloud native (microservices, k8s)
- orchestrate fleet of pktvisor Agents
- single pane of glass dashboarding
- create and explore Signal data sets in real time
- central analysis and alerting
Thank You!

Questions?

Shannon Weyrick  •  VP Research @ NS1  •  Office of CTO

sweyrick@ns1.com