ntopng and Suricata: Merging Network Visibility and Security

Luca Deri <deri@ntop.org>, @lucaderi
Alfredo Cardigliano <cardigliano@ntop.org>
About ntop

- ntop develops open source network traffic monitoring applications. All code is available at https://github.com/ntop

- ntop is a community: http://t.me/ntop_community

- Part of the Intel Innovator program.

- ntop is also the name of the first app we released in 1998, a web-based network monitoring application (today ntopng).

- Today our tools range from traffic monitoring (ntopng, nProbe), high-speed packet capture (PF_RING), Deep-Packet Inspection (nDPI), traffic recording (n2disk), DDoS mitigation (nScrub), IDS/IPS acceleration.
Network Visibility

- Network visibility ensures that you are able to see everything happening on a network. It includes:
  - Network performance
  - Application performance
  - Devices discovery

ntopng is a web-based open-source traffic analysis application that aims to provide full network visibility.
Uncorrelated Security Events

- Suricata, as well as other IDS systems, is commonly used to generate alarms when security threats are detected, and produce logs with suspicious network activities.

- There are many tools collecting logs produced by Suricata, and pushing them to system like ElasticSearch. The best they can do is index them and produce statistics: “Tell me how many Policy Violations we got today”.

- Threat detection is typically limited to a single session (see decode-events.c, app-layer-events.c) and it is (mostly) based on signatures matching. Suricata is basically a pure network sensor with no mechanisms for correlating information across multiple flows or hosts.
Augmented Security

- Network administrators need a clear picture of the traffic flowing into their network and place security events in the right context.
- Correlating security events with network traffic provides a better visibility of what’s going on and the root cause of threats.
- Single events that can be considered harmless when looking at them individually, could be small pieces of bigger harmful events.
ntopng Troubleshooting [1/2]

The problem is here
ntopng Troubleshooting [2/2]

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Duration</th>
<th>Severity</th>
<th>Alert Type</th>
<th>Drilldown</th>
<th>Description</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>07:31:02</td>
<td>02:32:50</td>
<td>Error</td>
<td>TCP SYN Scan</td>
<td>Host: 192.168.0.1 is under SYN Scan (127 &gt; 30 SYN received)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>07:31:02</td>
<td>02:32:50</td>
<td>Error</td>
<td>TCP SYN Scan</td>
<td>Host: 192.168.0.1 is under SYN Scan (609 &gt; 30 SYN received)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>07:31:02</td>
<td>02:32:50</td>
<td>Error</td>
<td>TCP SYN Scan</td>
<td>Host: 192.168.0.1 is under SYN Scan (609 &gt; 30 SYN received)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>07:31:02</td>
<td>02:32:50</td>
<td>Error</td>
<td>TCP SYN Scan</td>
<td>Host: 192.168.0.1 is under SYN Scan (153 &gt; 30 SYN received)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>07:31:02</td>
<td>02:32:50</td>
<td>Error</td>
<td>TCP SYN Scan</td>
<td>Host: 192.168.0.1 is under SYN Scan (42 &gt; 30 SYN received)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>07:31:02</td>
<td>02:32:50</td>
<td>Error</td>
<td>TCP SYN Scan</td>
<td>Host: 192.168.0.1 is a SYN Scan attacker (181 &gt; 30 SYN sent)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Showing 1 to 10 of 193 rows
ntopng Features and Limitations [1/2]

- Host system and containers monitoring through eBPF
- Process, container, POD and user statistics

Full path: useful for drill-down in case of security alerts
ntopng Features and Limitations [2/2]

- ntopng features:
  - Network traffic metrics
  - Anomaly detection
  - Blacklists for malware detection

- It lacks security features including:
  - Threat detection
  - Signatures support
  - File extraction
Suricata Limitations

- It does not use any DPI (Deep Packet Inspection) techniques to identify traffic regardless of the port it uses:
  - Running a service on a non-standard port might be invisible to it.
- No information about flows/protocols not dissected by Suricata.
- No encrypted traffic analysis (i.e. Cisco Joy-like technologies) beside protocol fingerprinting: the idea is to be able to decode traffic, but unencrypted traffic is becoming rare, and this has impact on visibility.
- It does not provide any facility that could help users to understand the “big picture” (e.g. ARP scan, DNS negative/positive response ratio, or too many host active flows with respect) as it focuses on per-flow analysis.

```
alert tcp $HOME_NET any -> $EXTERNAL_NET ![25,587,6666:7000,8076] (msg:"ET POLICY IRC Channel JOIN on non-standard port"
```
Motivation: Unify Visibility and Security [1/2]

• Suricata is a great tool for dissecting selected protocols, extracting key metrics, and emitting alerts based on flow content driven by external rules.

• ntopng is able to collect information from various sources (packets, NetFlow, sFlow), analyse them in a comprehensive format, and emit alerts. All in one place, with minimal requirements.

• What if we can unify these two open source tools into a single tool able to provide the best solution for complementing security and visibility? Seamlessly.
Motivation: Unify Visibility and Security [2/2]

- Benefits for the Suricata community:
  - Provide a web GUI to Suricata. Someone might say: there are many (ELK-based) tools that do that. True but they lack network visibility, require third parties DBs/tools, and are not been designed for networking/security.
  - Enhance Suricata with network metrics not reported by the tool.
  - Provide existing Suricata users with ntop features (e.g. nIndex-based efficient flow-storage or Slack-based alerts).

- Benefits for the ntop community:
  - Add the benefits of signature-based traffic analysis.
  - Merge Suricata traffic alerts with those already handled by ntopng to implement the best of both worlds.
ntopng Architecture

C++ Core Engine

Packet Processing
Flow Collection

{ REST:API }

Redis

OMQ

nDPI

influxdb

Lua

github.com/ntop
Suricata Eve

- The Suricata EVE output facility outputs events in JSON format.
- Events include:
  - Alerts (signature matches)
  - Flow records (à la Netflow)
  - Application layer metadata (HTTP, DNS, TLS, …)
  - Extracted files information
Syslog Collector Interface

- Ntopng implements Syslog-over-TCP ingestion to collect Syslog records from remote clients.
- Syslog records are processed by Lua modules based on the source application.
Alerts Ingestion

- Alerts generated by Suricata are collected through a Syslog interface.
- Binding the Syslog interface to a physical interface in ntopng we are able to:
  - Correlate events coming from Suricata with traffic processed by ntopng.
  - See network metrics and alerts (as well as other information coming from Suricata) in the same logical interface.
Configuration

- `ntopng -i eth0 -i syslog://127.0.0.1:9999`

*User’s Guide at https://www.ntop.org/guides/ntopng*
Suricata (Syslog) Interface
Flow Alerts

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Duration</th>
<th>Count</th>
<th>Severity</th>
<th>Alert Type</th>
<th>Drilldown</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00:17 ago</td>
<td>-</td>
<td>1</td>
<td>Error</td>
<td>External Alert</td>
<td></td>
<td>Detected TROJAN alert: W32/AlexSpy RAT Checkin [Emerging Threats] [Flow: 00:50:56:3B:68:67</td>
</tr>
<tr>
<td>03:07 ago</td>
<td>-</td>
<td>1</td>
<td>Warning</td>
<td>External Alert</td>
<td></td>
<td>Detected HTTP alert: unable to match response to request [Suricata] [Flow: 02:00:00:00:00:00:00:00</td>
</tr>
</tbody>
</table>
Flow Details

- **Flow Details**

- **Flow Peers [Client / Server]:**
  - Client: 172.16.255.148:1134
  - Server: 10.88:1064

- **Protocol / Application:**
  - Protocol: TCP
  - Application: Unknown (Unspecified)

- **First / Last Seen:**
  - First: 27/10/2019 14:57:15 [01:17 ago]
  - Last: 27/10/2019 14:58:05 [00:27 ago]

- **Total Traffic:**
  - Total: 2.89 KB
  - Goodput: 1.81 KB (62.7 %)

- **Client ➔ Server:**
  - 10 Pkts / 2.30 KB
  - Client ➔ Server: 10 Pkts / 600.00 Bytes

- **Round-Trip Time Breakdown:**
  - Client ➔ Server: 50.472 ms (Latency)
  - Client ➔ Server: 30.485 ms (Server)

- **Client / Server Estimated Distance:**
  - 20.278 Km
  - 12.518 Miles

- **Application Latency:**
  - 100.955 ms

- **Packet Inter-Arrival Time (Min / Avg / Max):**
  - Client ➔ Server: 100 ms / 227.5 ms / 381 ms
  - Client ➔ Server: 100 ms / 227.5 ms / 381 ms

- **TCP Packet Analysis:**
  - Lost: 1 Pkt / 1 Pkt

- **Max (Estimated) TCP Throughput:**
  - Client ➔ Server: 5.09 Mbit/s
  - Client ➔ Server: 5.09 Mbit/s

- **TCP Flags:**
  - Client ➔ Server: SYN, PUSH, ACK
  - Client ➔ Server: SYN, PUSH, ACK

- **Flow is active:**

- **Additional Flow Elements:**
  - **Community ID:** 1:IFYvcuw32ZEdzMV931x5y4Ao=
  - **Suricata Flow ID:** 64788950018662
L7 Metadata Ingestion

- Application layer metadata for selected protocols (e.g. HTTP, DNS, TLS, …) are generated by Suricata and collected through the Syslog interface.

- The Suricata protocol parser and stream reassembly engine can also be used to extract and store files to disk (e.g. from HTTP, SMTP, FTP, …).

- All metadata are ingested by ntopng and are used to compute metrics and run analysis (those natively supported) or just listed as “Additional Information”.
## HTTP & File Info

Flow reset by the client.

### HTTP
- **HTTP Method**: GET
- **URL**: www.republic.it/minify/sites/repubblica/video/config_rtv_DB.ca...
- **Response Code**: 200

### Additional Flow Elements
- **File Gaps**: No
- **File Name**: /content/nazionale/img/2016/02/21/16294456/83640f59-a515-4b7e-b088-cc859d376af7-th.jpg
- **File Size**: 8768
- **File State**: CLOSED
- **File Stored**: No
- **HTTP Content Length**: 8768
- **HTTP MimeType**: image/pjpeg
- **HTTP Protocol**: HTTP/1.1
- **HTTP Referer**: http://www.repubblica.it/sport/2016/02/21/foto_/balotelli_e_italiano_ma_ha_preso_tropp_sole_la_frase_di_berlusconi_non_sfigge_alla_scatmpa_straniera-139328856/
Flow Records Ingestion

- Suricata as a NetFlow-like flow exporter.

- Flow information generated by Suricata are collected through a Syslog interface, together with alerts.

- In this working mode, ntopng collects flows instead of processing packets.

- Drawback: ntopng cannot compute most of the Network metrics as it does not have packets visibility.
# Flows From Suricata

## Recently Active Flows

<table>
<thead>
<tr>
<th>Application</th>
<th>Protocol</th>
<th>Client</th>
<th>Server</th>
<th>Duration</th>
<th>Breakdown</th>
<th>Actual Thpt</th>
<th>Total Bytes</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>OAuth</td>
<td>UDP</td>
<td>MyMacbook:17500</td>
<td>192.168.1.255:17500</td>
<td>00:02</td>
<td>Client</td>
<td>50.04 bit/s</td>
<td>22.17 KB</td>
<td></td>
</tr>
<tr>
<td>Dropbox</td>
<td>UDP</td>
<td>MyMacbook:17500</td>
<td>Broadcast:17500</td>
<td>00:02</td>
<td>Client</td>
<td>56.38 bit/s</td>
<td>24.98 KB</td>
<td></td>
</tr>
<tr>
<td>NTP</td>
<td>UDP</td>
<td>MyMacbook:ntp</td>
<td>defns1-ntp-001.aapling.com:ntp</td>
<td>00:02</td>
<td>Client</td>
<td>2.18 bit/s</td>
<td>980 Bytes</td>
<td></td>
</tr>
<tr>
<td>DNS</td>
<td>UDP</td>
<td>fe80:3a15:c2ff:feab:720...mdns</td>
<td>402::fb:mdns</td>
<td>00:02</td>
<td>Client</td>
<td>31.79 bit/s</td>
<td>14.09 KB</td>
<td></td>
</tr>
<tr>
<td>DHCP</td>
<td>UDP</td>
<td>192.168.1.254:bootps</td>
<td>Broadcast:bootp</td>
<td>00:02</td>
<td>Client</td>
<td>35.42 bit/s</td>
<td>15.7 KB</td>
<td></td>
</tr>
<tr>
<td>Unknown</td>
<td>TCP</td>
<td>10.10.10.10:80</td>
<td>MyMacbook:56364</td>
<td>00:01</td>
<td>Server</td>
<td>4.28 bit/s</td>
<td>1.89 KB</td>
<td></td>
</tr>
<tr>
<td>Unknown</td>
<td>TCP</td>
<td>MyMacbook:56361</td>
<td>10.10.10.10:80</td>
<td>00:01</td>
<td>Client</td>
<td>3.12 bit/s</td>
<td>1.38 KB</td>
<td></td>
</tr>
<tr>
<td>ICMPv6</td>
<td>IPv6/ICMP</td>
<td>fe80::18e6:4db6:b604:df7...</td>
<td>ip6-allrouters [IPv6]</td>
<td>00:01</td>
<td>Client</td>
<td>0.74 bit/s</td>
<td>334 Bytes</td>
<td>Echo Reply</td>
</tr>
<tr>
<td>ICMPv6</td>
<td>IPv6/ICMP</td>
<td>fe80::18e6:4db6:b604:df7...</td>
<td>402::16</td>
<td>00:01</td>
<td>Client</td>
<td>0.64 bit/s</td>
<td>290 Bytes</td>
<td>Echo Reply</td>
</tr>
<tr>
<td>IMAPS</td>
<td>TCP</td>
<td>MyMacbook:56373</td>
<td>smtps.aruba.it:imaps</td>
<td>00:01</td>
<td>Client</td>
<td>23.69 bit/s</td>
<td>10.5 KB</td>
<td></td>
</tr>
</tbody>
</table>

Showing 1 to 10 of 1652 rows. Idle flows not listed.
Flow Details w/o Packets

<table>
<thead>
<tr>
<th>Additional Flow Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Community ID</td>
</tr>
<tr>
<td>Suricata Flow ID</td>
</tr>
<tr>
<td>TLS Certificate DN</td>
</tr>
<tr>
<td>TLS Certificate Not Before</td>
</tr>
<tr>
<td>TLS Certificate Subject</td>
</tr>
<tr>
<td>TLS Version</td>
</tr>
</tbody>
</table>
Flow Details w/ Packets
Ongoing Activities

Analysis and Consolidation

Containers

Packet Processing

Feeds and Blacklists

Data

NetFlow

IPFIX

sFlow

Work In Progress

eBPF

Containers

Alerts

suricata

B

S

Ongoing Activities

Analysis and Consolidation

ABUSE

github.com/ntop

Packet Processing

Alerts

sFlow

NetFlow

IPFIX

Data

Containers

Work In Progress

eBPF

GitHub

Ongoing Activities

Analysis and Consolidation

ABUSE

github.com/ntop

Packet Processing

Alerts

sFlow

NetFlow

IPFIX

Data

Containers

Work In Progress

eBPF

GitHub
Final Remarks

• Network security and visibility is now possible.

• Comprehensive merge of Suricata alerting information with ntopng traffic analysis.

• Benefits for the whole open source community, as well the ntopng and Suricata communities.

• Hopefully closer integration using nDPI into Suricata for characterising traffic unknown to Suricata.