Towards A CyberScore-based Network Traffic Monitoring System

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

Introduction [1/2]

- Most cybersecurity tools have been designed only for security and have poor network visibility as their only focus is to detect threats.
- Many tools are signatures and rule-based as they have been designed to detect specific threats for which they have prior knowledge (caveat zero-day attacks are often not detected).
- The above tools have no knowledge of available network services as they are basically network sensors that have no global network knowledge and thus focus on specific network events at flow-level.

Introduction [2/2]

- Often these tools (e.g. Suricata IDS) have been used to produce signals that are used by higher-level applications to emit meaningful alerts.
- AI (Artificial Intelligence) is frequently used in these signals to implement the logic for detecting non-flow-based alerts, such as malware detection.
- Commercial tools based on AI, often require periodic vendor supervision and tuning at the customer site, practice that increases the overall price and prevents end-users from being able themselves to tune the tool they purchased.

Motivation [1/2]

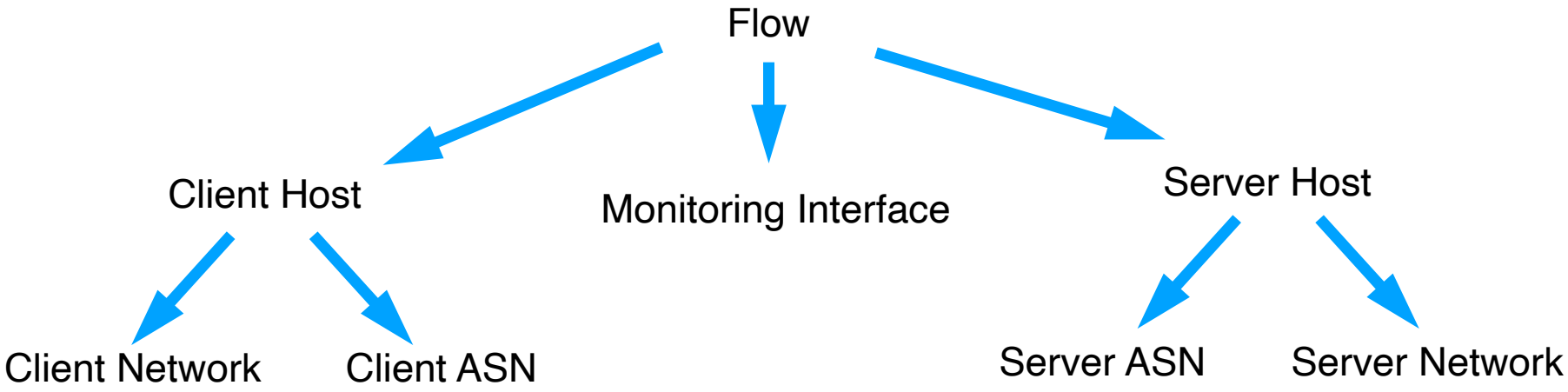
- While AI is a core technology, we believe that it has also limitations in terms of:
  - Operational costs (high on-site hardware costs, or service-based cloud service).
  - As signals used by the above tools are cybersecurity-centric, most tools are blind with respect to network activities (e.g. top talkers).
  - Zero-day attacks for which a signature or behaviour has not been defined and thus no model for this activity exists.

Motivation [2/2]

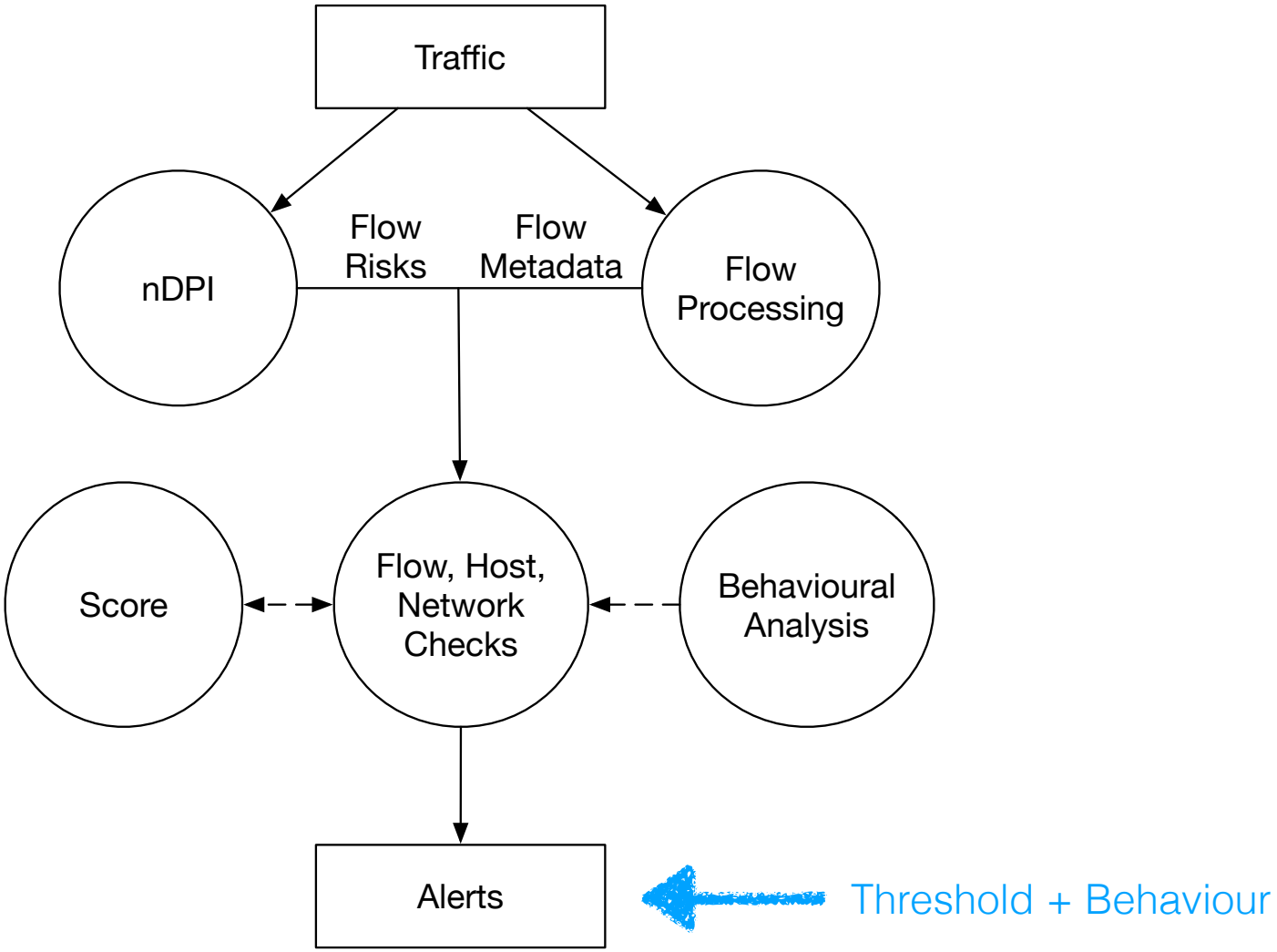
- Limitations of signature-based systems, and the complexity and cost of AI-based tools has been the motivation to develop a novel method able to:
  - Combine behavioural traffic analysis and encrypted network traffic analysis.
  - Use statistical-based methods and thus light and simple to operate in comparison to AI-based systems.
  - Release this work as open source so that every researcher can improve it, and use it as starting point for his research.

Cyberscore [1/2]

- This work is based on the cyberscore concept which is a (numerical) relevance indicator that is assigned to every observed network activity. When a cyberscore is assigned to malicious network communication, the higher the value, the higher the severity of this activity.
- Similar to multivariate time-series analysis, the cyberscore is a unique value that summarises the result of the analysis at various granularity levels.



Cyberscore [2/2]



Packet / Flow Checks [1/2]

- Suspicious Data Transfer (e.g. binary application transfer).
- Data Exfiltration (e.g. over ICMP and DNS).
- Unexpected Traffic (e.g. DNS packets larger than 512 bytes, TLS traffic with no SNI).
- Alerts based on communication with a remote host present on a third-party blacklist (e.g. Cisco Thalos or Abuse.ch).
- Suspicious Traffic (e.g. suspicious HTTP user-agent or unidirectional unicast UDP traffic).
- Elephant (i.e. large uploads/downloads) or Long-Lived Flows.
- Insecure or Obsolete Protocol versions (e.g. TLSv1 or obsolete SSH client version).

Packet / Flow Checks [2/2]

- Unexpected DNS (e.g. DNS server not allowed, invalid DNS queries, DNS data exfiltration).
- Suspicious HTTP (e.g. HTTP Numeric IP Host, HTTP Suspicious Header/URL/User-Agent).
- Web Mining (e.g. traffic to hosts known to perform cryptocurrencies mining).
- Possible Exploit Detected: communications that indicate an abnormal behaviour with respect to the host service map.
- Binary Application Transfer or other exploits (e.g. potential SQL injection).

.

Host Checks [1/2]

- Threshold-based Alerts (e.g. a non-NTP server host that contacts over 4 different NTP servers, or a host that contacts too many hosts/ASNs/ countries in the last minute).
- Network and Port Scan Detection.
- Behavioural Alerts (e.g. a host has an unusual number of client/server flows with respect to its recent past).

Host Checks [2/2]

- Lateral Movement Detection (e.g. an unexpected or new service is detected, leveraging on traffic behaviour learning and service map).
- Server Contacts (e.g. an anomalous number of domain names or NTP servers contacted).
- Flow Floods (e.g. a high number of flows generated).
- Scan Detection (e.g. a high number of connection attempts with an incomplete three-way handshake).

.

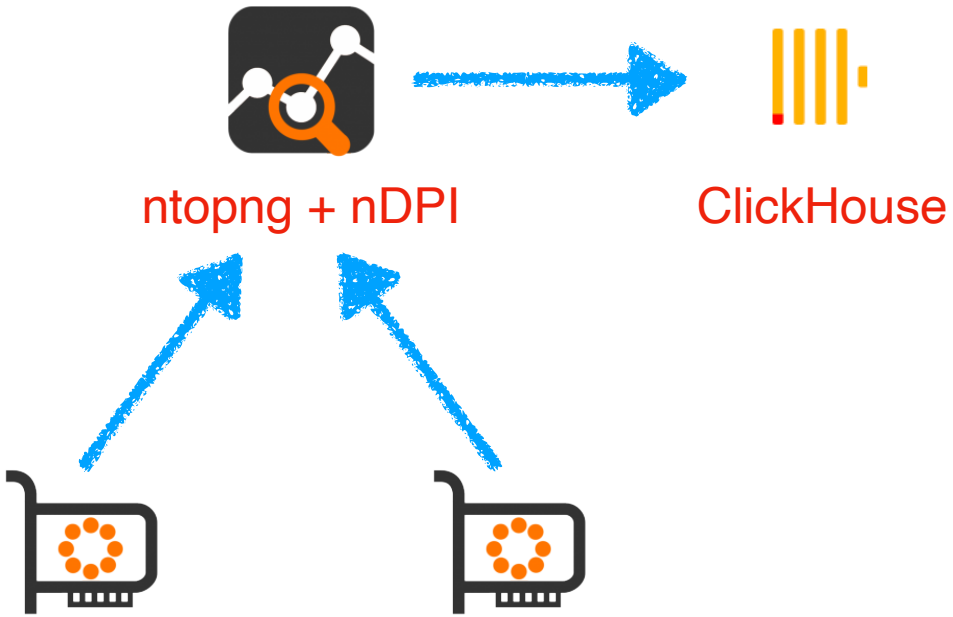
Risk Exceptions

- The cyberscore principle is effective only if there are no false positives as otherwise they can deceive detection algorithms by generating fake alerts.
- These are not false positives, but real alerts that need to be silenced (i.e. add an exception) as they are not relevant and increase cyberscore for no reason. Examples:
  - Insecure protocols/hosts that cannot be upgraded but that provide a specific service to a few clients.
  - Applications running on non standard ports (e.g. SSH server on port 2222).
  - TLS towards numeric IP address (no symbolic hostname)

Validation [1/2]

- This work has been validated in two different environments:
  - Service Provider (2.5k hosts, 100k hosted domains):
    The monitoring goal is not to tackle all the security issues, but to only identify and isolate attackers or infected systems hosted on the provider network, this to protect the network assets by remote attackers that gained control of some hosted systems.
  - Corporate Network (~2k hosts)
    Protected by modern firewall/IDS/EDR. Monitoring goal is to detect malicious communication patterns and spot potential threats due to personal/mobile devices and external/remote users that connect to the network though a VPN.

Validation [2/2]



Intel Xeon Silver 4116 CPU, 64 GB of RAM, 512 GB SSD

Note: in red open source components.

Lesson Learnt [1/4]

- With cyberscore there is no need to perform a training/ annotation as with AI-based systems, but it is necessary to silence specific checks (triage) to avoid unexpected noise that will influence the host cyberscore value.
- We based behavioural alerts on top of exponential smoothing techniques, that allowed us to easily detect when a metric deviates from its expected value (having a unique threshold for heterogeneous hosts is not possible).
- (High-quality) IP reputation blacklists have not shown false positives and hosts listed in blacklists are responsible for about 74 % of the reported alerts: blacklists are a great way to “anticipate” issues.

.

Lesson Learnt [2/4]

- Attackers Detection: partitioning assets in pools of homogeneous hosts (e.g. VPN hosts, routers/ switches, Win hosts…) allows us to enable specialised checks and thus immediately label attackers or misbehaving hosts (e.g. when a SSH to a Windows host is very suspicious).
- Triplet <IP source, IP destination, destination port> and twin <IP source, IP destination> minute counters (absolute and behavioural)… enabled us to quickly identify scanners and important hosts (through edge and rank analysis).

.

Lesson Learnt [3/4]

- Slow scanner have been identified using connection frequency vs flow size analysis (using data binning techniques), that in normal traffic has high variance, whereas limited (or with almost zero variance) with scanners.
- DNS Traffic: on the service provider 52% of DNS queries are not related to address resolution, but often used for querying host reputation services. They use DGA-like queries that need to be whitelisted in order not to trigger invalid cyberscore. Communication with IPs previously resolved to DGAs over unknown (i.e. not detected by nDPI) protocols, is a good way to increase cyberscore.

.

Lesson Learnt [4/4]

- Signature and behavioural based algorithms often are ineffective with zero-day attacks. For this reason we have implemented local service and periodicity maps in order to detect “new” communications (or a different pace).
- We believe that the main direction for future research is to keep creating (dynamically) per-host models based on network traffic and services, detect (IP and domain) popularity changes, and (anonymously) correlate data across monitored systems.

Conclusions and Original Paper Contributions

- The paper described the design and implementation of a novel technique named cyberscore. It has also shown how it has been effectively used in two popular yet very different scenarios.
- We have compared our results with a market leader AI-based tool operated on the same network and this work demonstrated to report similar results in cybersecurity, while reporting network issues not reported by the other tool that is apparently blind to non-security related communications.
- This work demonstrated that this open-source technology is mature and efficient to monitor thousands of hosts with Gbit links with a low-end commodity server.

https://github.com/ntop/