Introducing NetBehave (.org)

OpenSource Tools for Network Behavior Analysis

Yves B. Desharnais, MBA, CISSP, PCIP

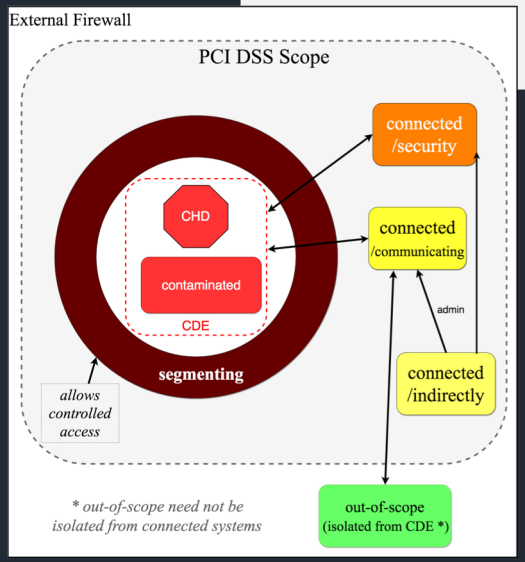
BSides Ottawa 2018 – November 8, 2018

Disclaimer

- This presentation is the result of my experience and only represents my personal views, and is not endorsed by anyone other than myself
- There is new code so some bugs may still be present (YMMV)
- The code will live on github and available at NetBehave.org
- NetBehave is containerized (Docker) for convenience, but:
  - Other security controls are required to secure the system in a production environment, for example:
    - Installing on an hardened, patch operating system (and patching it periodically)
    - Restricting accesses
    - Etc.

About Yves

- IT/InfoSec Expert generalist.
- Over 17 years professional experience (IT/Infosec)
- Background in software development, Unix/Linux administration
- Author of books on PCI DSS including
  - PCI DSS Scope methodology and approach (one of 2 public methodologies at the time) CC-BY-SA



How was this approach born?
Need of a PCI Compliance Project

From interview: Status

- 2nd year PCI DSS compliance extension (< 5 months left) – acquired company network
- flat network, mostly VMs, no patching/hardening
- few diagrams used to prepare system migrations to the new network (i.e. incomplete documentation)
- card number tokenization in use
- very competent IT staff

How to achieve compliance?

- This called for massive reduction in scope if they were to have any chance of achieving compliance
- With limited information on applications, we needed to identify what talked on the network to whom and on what protocols…
- first thought: span port and packet capture, but speaking to network friend I learned of the NetFlow protocol

NetFlow version overview

	5	9	10 (IPFix)
Owner of Standard	Cisco	Cisco	IETF
RFC	None (1996 or later)	3954 (©2004)	7011 (©2013)
Format	Fixed	Template-based	
IP version	IPv4 only	Any (v4, v6) & Ethernet (others?)	
Equipment Supporting	Older Cisco switches/routers	Newer Cisco, some other network providers	Most other providers (including VMWare)

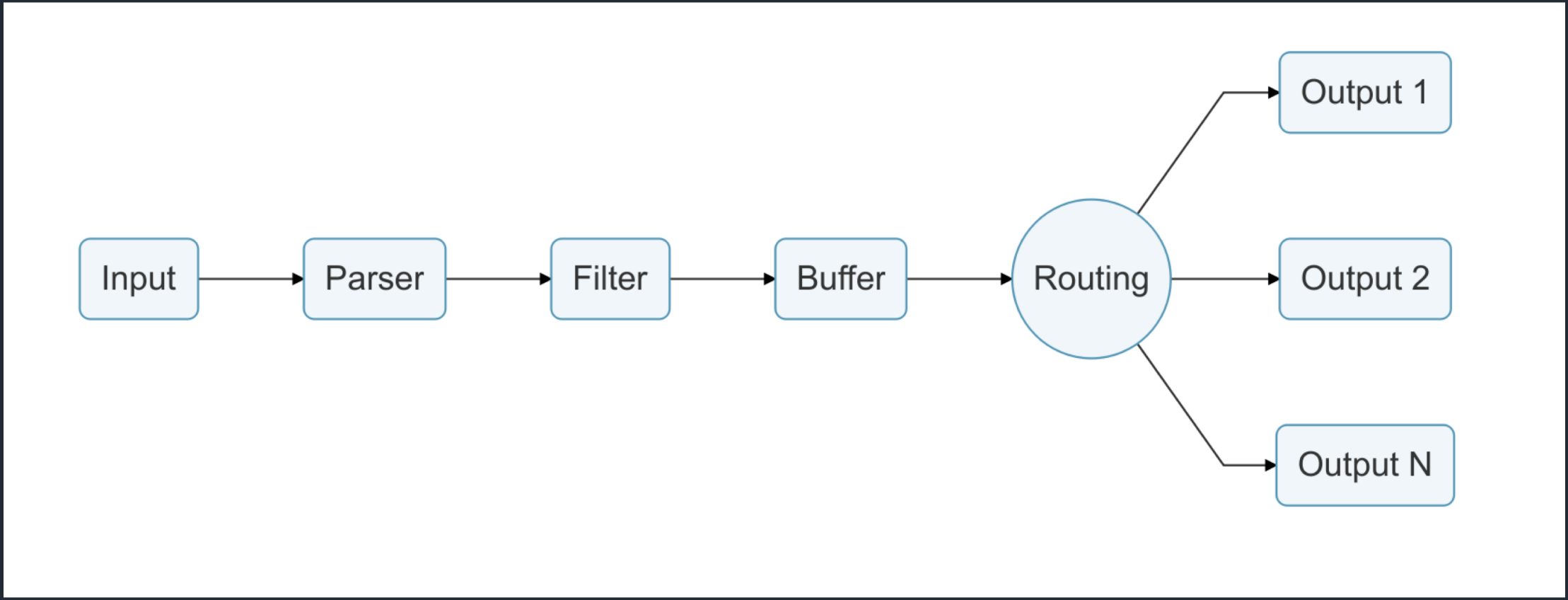
Back in 2016…
How do I capture and analyze NetFlow?

- Market research - little time, and little budget
- Many tools supported NetFlow (including NTOP, SiLK) – none met my needs for scoping
- Roll-your-own with either Logstash or Fluentd - both ruby based
- Logstash (famously the L in the ELK stack) – Ruby/J (requires JDK, more portable, more centralized framework)
- Fluentd is more community driven – Ruby/C (uses less RAM)
- Both are basically data aggregator/transformations engines using plugins, aka the *nix command line on steroids but using JSON

Fluentd : the backbone of the project

- An open source workflow-like solution written in Ruby consisting at the core of plugins in the structure:
  - Input (and parsers) => Filter(s) => Output
- Workflows initially built using tags, but now using labels
- Can support multiples “pipelines” at once (YMMV)
- Many plugins already exist (> 100), distributed ecosystem
- You can do a lot with just configs and plugins (no coding), but configs can be made simpler with plugins (which need not be complex)
- Writing plugins is done in Ruby (NetBehave includes many new plugins)

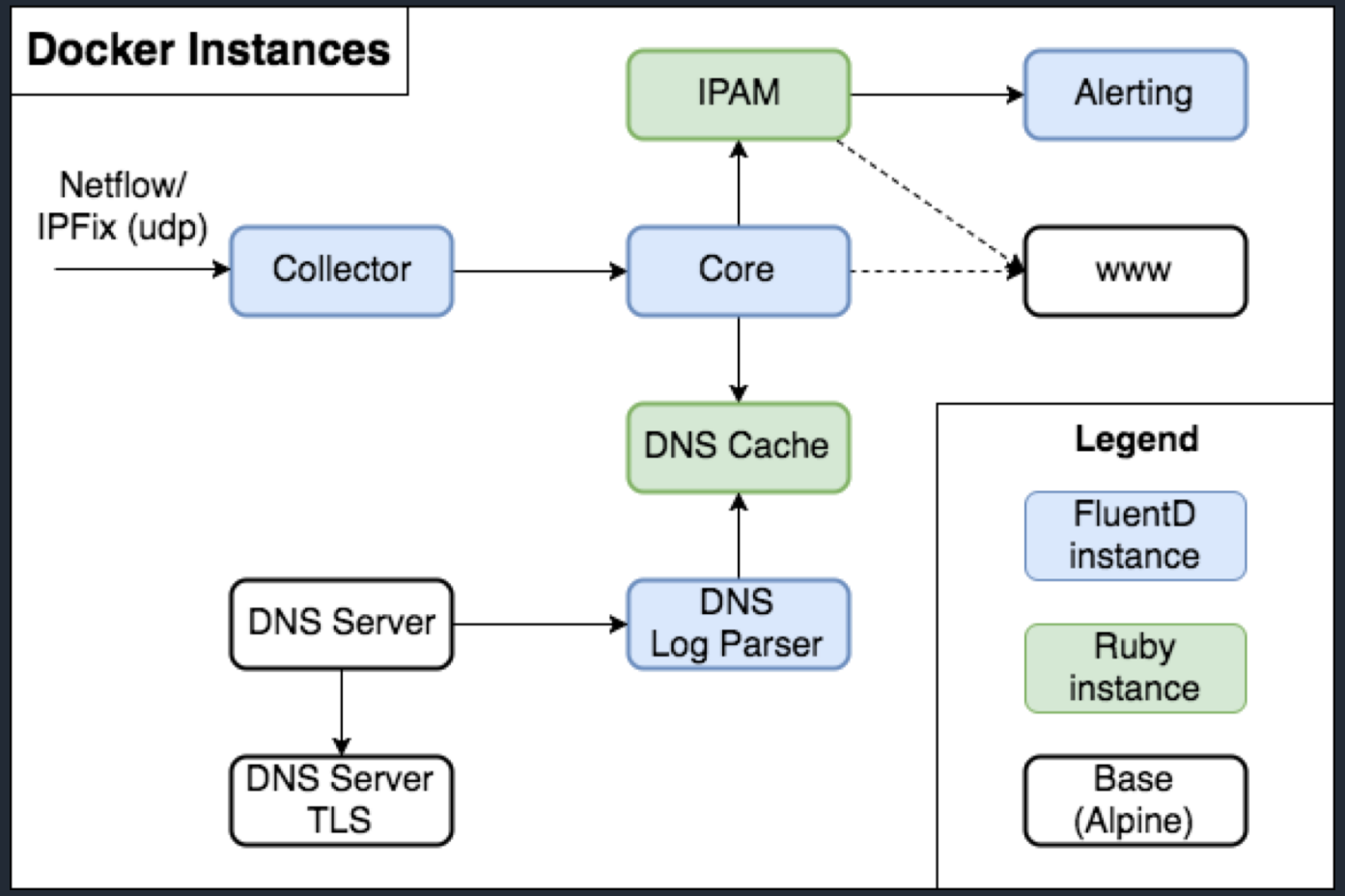
Fluentd : what a logging pipeline looks like



NetBehave solution design

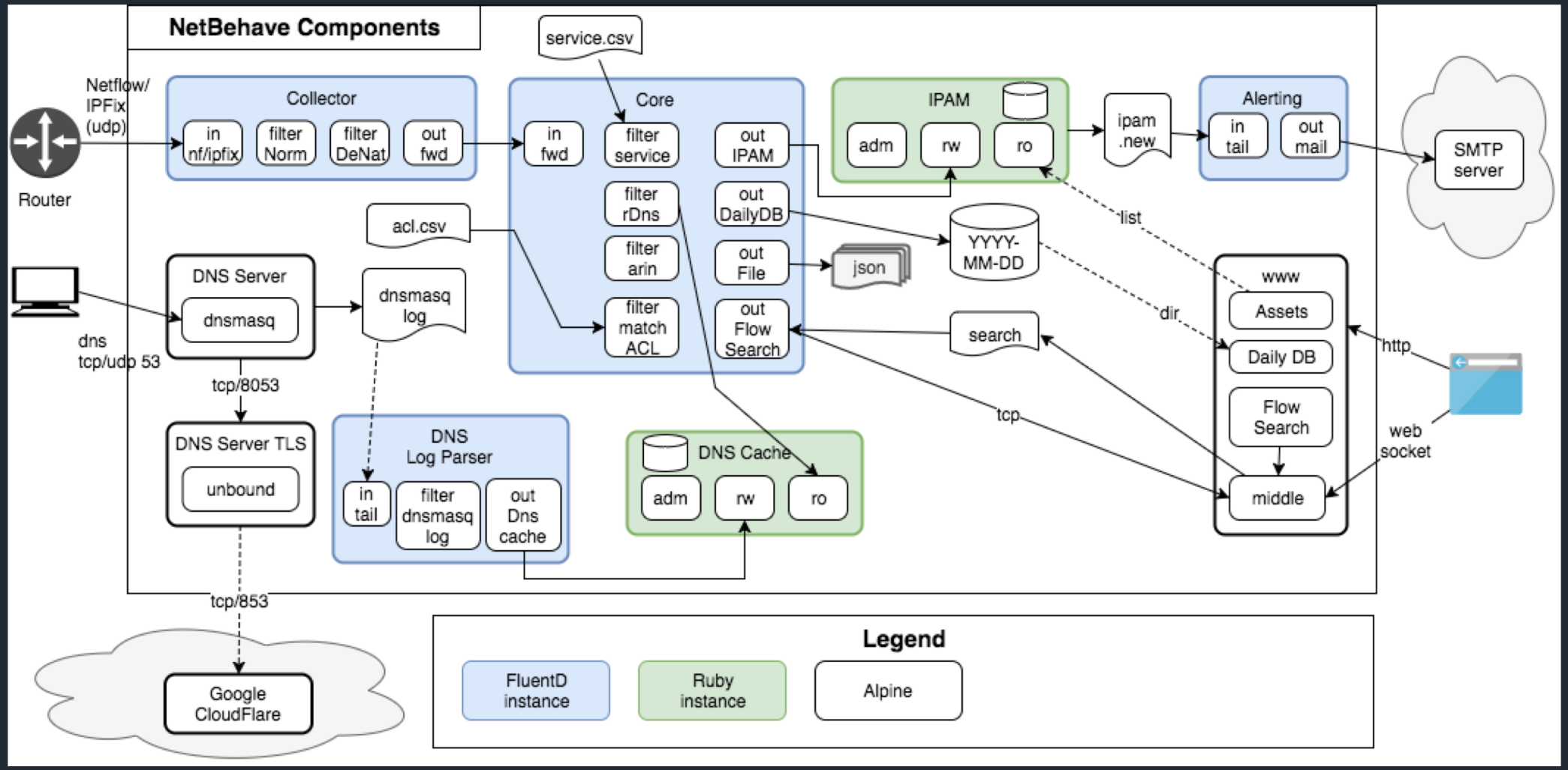
- Leverage the experience of the 2 past years
- Built on the Unix philosophy: make each program do one thing well.
- Plugins will help accomplish this task, as will Docker images
- Simplify install on various platforms using Docker images and Docker-compose to create “apps”
- This is core plumbing that may require higher level functionality tailored to each business needs (a.k.a. it’s very usable for my needs, and it is extensible)
- Allow to create various “solutions” from building blocks
- Add some non-fluentd ruby code for tasks (assets, caches, www)

NetBehave architecture



- Base (alpine + s6)
  - DnsServer (DnsMasq)
  - DnsServer-TLS (Unbound)
  - www (busybox HTTPD)
  - Ruby
    - IPAM
    - DnsCache
- Base+Fluentd
  - Alerting
  - Collector
  - Core
  - DnsLogParser

NetBehave detailed architecture



New fluentd plugins added to fluentd

- Input : Netflow/IPFix – backward compatible with fluentd Netflow plugin
- Filter : DnsmasqLogParser – parses Dnsmasq logs to feed the DNScache
- Filter: FlowNormalizer – Creates a “flow” object standard for all NetFlow/IPfix versions
- Filter: FlowDeNat – Collects outgoing connections and maps them back to natted traffic
- Filter: FlowArinWhois – perform Arin block name search
- Filter: FlowReverseDNSCache – match domain name from IP
- Filter: FlowService – match proto/port to service in csv file
- Filter: FlowMatchACL – matches flow objects to ACL in a csv file
- Output: FlowIPAM – sends new IP information to the IPAM
- Output: FlowDailyDB – outputs a daily SQLite file with all collected flows summarized (total bytes/packets)
- Output: DnsCache – sends DNS information (from Dnsmasq) to the DNScache
- Output: FlowSearch – allows for live searching of flows matching changing ACLs
- Flow* plugins are made to work with flow structure

Demo

- Demos are controlled by using saved JSON from the first conversion step of the process (2017 demos where with saved PCAP)
- Instructions, code, and this presentation will be available online after the talk

Solution Issues / Parting Thoughts

- Netflow
  - Is UDP, so risk of packet loss – keep collectors close to source (and distribute processed information) => hence the collector is separable from the core
  - Multiple devices could provide duplicate traffic information (no dedup now)
  - Different devices provide different templates, offer different possibilities
- Fluentd
  - Can be setup in hierarchy for distributed sites and high-availability
  - has other uses: Logging and Monitoring (using nxlog)
  - Can be combined with other tools (elasticsearch, etc.)
- Solution
  - May require more manual edits and tweaks
  - New plugins are not packaged using gems (Ruby packaging system)
  - It can adapt to your needs with existing and/or new plugins

Stay in touch!

- Everything is published on www.netbehave.org (hosted on github)
  - Blog Post including links to
    - Instructions / Source code
    - This presentation
- For questions regarding
  - NetBehave (Issues page on github)
  - PCI DSS (www.PCIresources.com)
  - Wholistic security, contact me at:
    - yves@yvesbd.com or LinkedIn