# Skydive

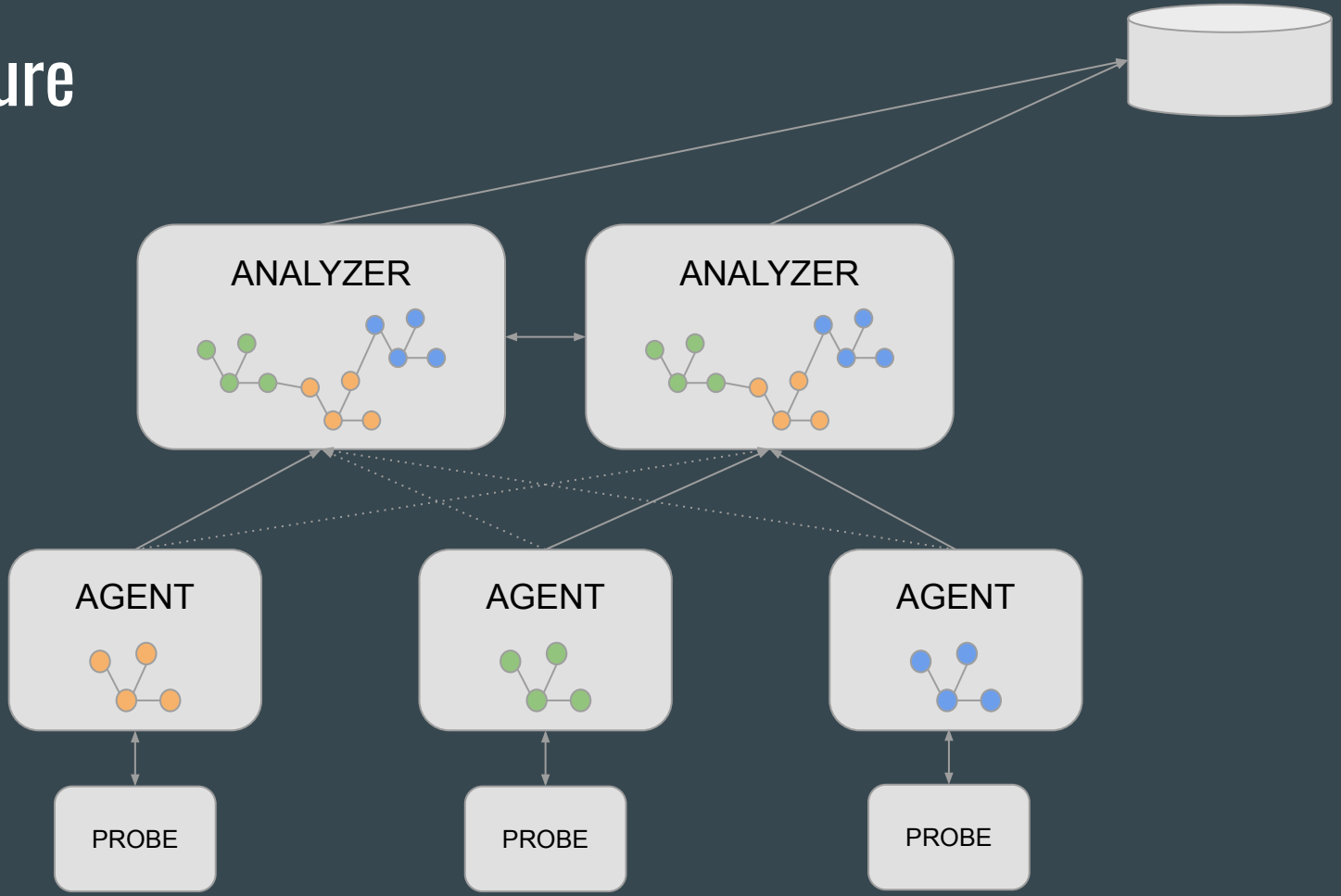
An analyzer for network topology and traffic

Sylvain Baubeau
Sylvain Afchain

# Skydive goals

- Topology exploration and visualization
- Network traffic capture
- Make network troubleshooting easier
- SDN agnostic
- Real-time / post-mortem network analysis framework
- Lightweight, easy to deploy
- Event based

# Architecture



# Network topology as a graph

- Event based
- Pub/Sub, WebSocket
- Node/Edge create, update, delete event
- Revision of every graph modification

# Topology probes

- Netlink, metrics, routing tables, etc.
- OpenvSwitch, OVN, table, metric, exploration
- VPP
- LLDP
- LibVirt
- BESS
- Docker, Runc
- Kubernetes, OpenShift
- Network Service Mesh
- OpenStack Neutron
- OpenContrail/Tungsten Fabric
- Block Devices
- Socket Info

And more....

# Distributed packet capture (1)

- Distributed packet capture
- Multiple capture methods

  AFpacket, eBPF, DPDK, sFlow, ...
- BPF Filtering
- Support L2/L3 flow tracking, tunnels supported too

  GRE, VXLAN, MPLS, etc.

# Distributed packet capture (2)

- Packet/Flow forwarding to an external endpoint

  sFlow, NetFlow, ERSPAN
- Flows and metrics stored in a time series database
- Flow bus, ex : conversion to VPC Flow Logs
- Same query method than for topology:

  G.V().Has(“Capture.Name”, “my-dockers-cap”).Flows(“Application”, “HTTPS”).Metrics()

# Distributed packet injection

- Distributed packet generator/injection
- ICMP, TCP, UDP, replay PCAP traces
- Long running injections
- Ping-mesh with RTT report

  skydive client injection create --dst 'G.V().Has("Type, "veth")' --src 'G.V().Has("Type, "veth")'
- PCAP socket probe allowing to inject remote or recorded traffic

# Wait ! There's still more..

- Alerting
- Workflows (in JavaScript)
- Open API, Rest, WebSocket
- Golang and Python client
- Ansible module
- Plugin support, ex: Collectd
- Grafana datasource

# Thanks

https://skydive.network

https://github.com/skydive-project/skydive

https://github.com/skydive-project/skydive-ui