

# Using OpenConfig Streaming Telemetry with Prometheus

Oliver Herms @ DENOG 10, <[oliver.herms@exaring.de](mailto:oliver.herms@exaring.de)>

# Agenda

- ~~1. Who am I?~~
2. Problem statement
3. What is OpenConfig Streaming Telemetry
4. How does it work?
5. How to use it with Prometheus
6. Questions

- State of whitebox network monitoring
  - ◆ CLI scraping
  - ◆ SNMP
  - ◆ Netconf
- Typical data resolution these days is fairly low

# What is OpenConfig Streaming Telemetry

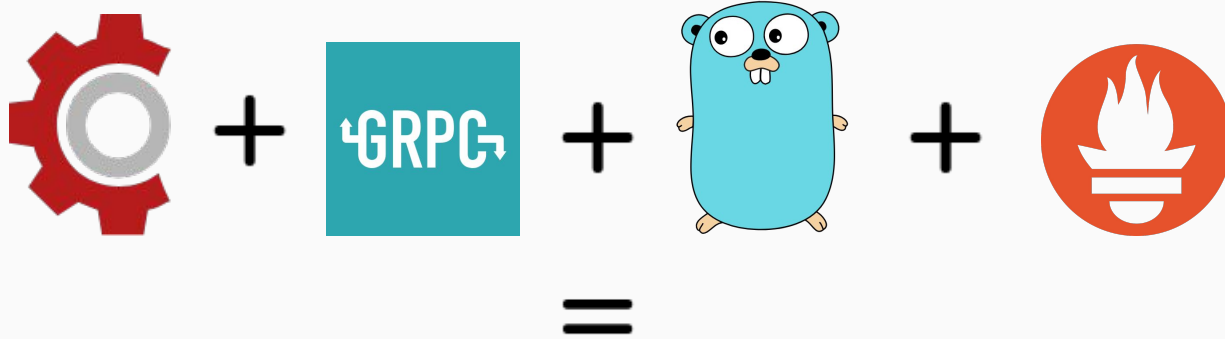
- OpenConfig is an informal working group
  - ◆ “moving our networks toward a more dynamic, programmable infrastructure by adopting software-defined networking principles”
  
- Streaming Telemetry
  - ◆ “a new approach for network monitoring in which data is streamed from devices continuously with efficient, incremental updates”
  - ◆ Uses gRPC/Protobuf for data transmission
  - ◆ Monitoring station connects to network devices and subscribes for certain metrics
  - ◆ Network devices push periodic incremental updates

# OpenConfig Streaming Telemetry - Details

```
message OpenConfigData {
  // router name:export IP address
  string system_id          = 1
  // line card / RE (slot number)
  uint32 component_id      = 2
  // PFE (if applicable)
  uint32 sub_component_id  = 3;
  // Path specification for elements of OpenConfig data models
  string path              = 4;
  // Sequence number, monotonically increasing for each
  // system_id, component_id, sub_component_id + path.
  uint64 sequence_number   = 5;
  // timestamp (milliseconds since epoch)
  uint64 timestamp        = 6;
  // List of key-value pairs
  repeated KeyValue kv    = 7;
  // For delete. If filled, it indicates delete
  repeated Delete delete   = 8;
```

...

# OpenConfig Streaming Telemetry Exporter



# OpenConfig Streaming Telemetry Exporter

- Released last week:  
[github.com/exaring/openconfig-streaming-telemetry-exporter](https://github.com/exaring/openconfig-streaming-telemetry-exporter)
- Translates path into metric names and labels into prometheus labels.
  - ◆ `"/interfaces/interface[name=xe-0/0/0]/state/mtu"`  
becomes  
`interfaces_interface_state_mtu{name="xe-0/0/0"}`
- No path specific code. Low maintenance.

Problem: Some metric values are not numeric

e.g. `/interfaces/interface[name=xe-0/0/0]/state/admin-status = "UP"`

Solution:

```
string_value_mapping:
```

```
# Path to do mappings for
```

```
/interfaces/interface/state/admin-status:
```

```
# string(DOWN) mapped to int(0)
```

```
DOWN: 0
```

```
# string(UP) mapped to int(1)
```

```
UP: 1
```



- Ever wondered how to get meaningful metadata attached to your metrics?
- The Exporter takes key=value pairs from descriptions (e.g. interfaces) and attaches them as labels to metrics
- Interface description:
  - ◆ `"rdev=core01.lej01,rif=xe-0/0/38:0,cid=ods-lej01-fra01-002.02,lpatch=A3-03-C02,role=WAN,ae=ae2"`
  - ◆ `interfaces_interface_state_mtu{name="xe-0/0/0",rdev=core01.lej01,rif=xe-0/0/38:0,cid=ods-lej01-fra01-002.02,lpatch=A3-03-C02,role=WAN,ae=ae2}`

# Thank you for your attention!

## Questions please!

[github.com/exaring/openconfig-streaming-telemetry-exporter](https://github.com/exaring/openconfig-streaming-telemetry-exporter)  
Contributions welcome!