



# Prometheus et InfluxDB pour le monitoring réseau

La convergence des forces entre les outils.

Alexis LAMEIRE

FRnOG 36

16 septembre 2022

# Who am I

- Alexis Lameire
- Du système, de l'automatisation mais surtout du réseau
- Un couteau suisse, mais pas de Prisunic
- Chez Cegedim.cloud depuis 9 mois



# De Nagios aux TSDB



# Les Nagios-like, en bref

- Un script = un état de santé (ok, warning, critical);

# Les Nagios-like, en bref

- Un script = un état de santé (ok, warning, critical);
- monitoring centré sur l'host;

# Les Nagios-like, en bref

- Un script = un état de santé (ok, warning, critical);
- monitoring centré sur l'host;
- le graphing est un ajout maladroit via les performance metrics.

# Nagios



Zabbix

ZABBIX

# Zabbix en bref

- un monitoring centré sur la métrique;

# Zabbix en bref

- un monitoring centré sur la métrique;
- la métrique est collectée via divers mécanismes;

# Zabbix en bref

- un monitoring centré sur la métrique;
- la métrique est collectée via divers mécanismes;
- pas mal de capacités de graphing, énormément en alerting;

# Zabbix en bref

- un monitoring centré sur la métrique;
- la métrique est collectée via divers mécanismes;
- pas mal de capacités de graphing, énormément en alerting;
- mais il est très compliqué de sortir de l'approche centrée sur la machine.

# Zabbix





# Les NMS

- Simples à mettre en place ;

# Les NMS

- Simples à mettre en place ;
- automagique, énormément d'infos rapidement ;

# Les NMS

- Simples à mettre en place ;
- automagique, énormément d'infos rapidement ;
- mais difficile de trouver l'information pertinente ;

# Les NMS

- Simples à mettre en place ;
- automagique, énormément d'infos rapidement ;
- mais difficile de trouver l'information pertinente ;
- souvent centrés sur l'host.

# Les NMS





*Il est illusoire de croire  
qu'on peut se contenter du standard  
pour son monitoring*



# Recette d'un bon outil de monitoring



# Recette d'un bon outil de monitoring

- Centré sur la métrique;
- une méthode de collecte standard;
- aucune adhérence à l'host;
- les graphes avant les alertes.

# Time Serie Database

# Les TSDB



Prometheus



*influxdb*

# Les TSDB

Prometheus :

- pull based;
- orienté métrique (possibilité de jointure);
- format standard (HTTP + format texte);
- faible rétention;
- alerting bien intégré.

# Les TSDB

## Prometheus :

- pull based;
- orienté métrique (possibilité de jointure);
- format standard (HTTP + format texte);
- faible rétention;
- alerting bien intégré.

## InfluxDB

- push based;
- orienté column (jointure complexe voire impossible);
- API dans de multiples langages;
- forte rétention;
- alerting moins bien intégré.

# Monitoring des équipements réseau

« *Le SNMP c'est vieux, c'est lourd...*  
*Mais les alternatives ne marchent pas* »

# SNMP faute de mieux

La télémétrie existe mais :

- n'est pas disponible pour tous les équipements ;
- est parfois soumis à licence ;
- est buggée, faute de réel intérêt.

Pour tenter de garder une cohérence, SNMP s'impose toujours !

# La collecte du SNMP

☰ README.md

## 🔗 Prometheus SNMP Exporter

This exporter is the recommended way to expose SNMP data in a format which Prometheus can ingest.

To simply get started, it's recommended to use the `if_mib` module with switches, access points, or routers.

### Concepts

While SNMP uses a hierarchical data structure and Prometheus uses an n-dimensional matrix, the two systems map perfectly, and without the need to walk through data by hand. `snmp_exporter` maps the data for you.

### Prometheus

Prometheus is able to map SNMP index instances to labels. For example, the `ifEntry` specifies an INDEX of `ifIndex`. This becomes the `ifIndex` label in Prometheus.

If an SNMP entry has multiple index values, each value is mapped to a separate Prometheus label.

### SNMP

## Telegraf

Telegraf is the open source server agent to help you collect metrics from your stacks, sensors, and systems.

[Get Started](#)

# prometheus-snmp-exporter

- se base sur les MIB pour générer sa configuration ;
- extrêmement rigide à l'utilisation ;
- la collecte SNMP est faite sur l'équipement lors du scrapping ;
- impossible de filtrer ou renommer les métriques.

# telegraf

- traite le SNMP en tant que table ;
- impossibilité d'insérer plusieurs tables au sein d'un measurement ;
- impossibilité de filtrer les données.



*Les exporters SNMP ont été écrits...  
Par des gens qui n'ont jamais utilisé SNMP.*



# Genèse d'un projet libre

# Écriture d'un exporteur Prometheus

Petits extraits choisis des recommandations Prometheus :

- Metric names should never be procedurally generated, except when writing a custom collector or exporter ;
- Each exporter should monitor exactly one instance application ;
- Metrics should only be pulled from the application when Prometheus scrapes them.

« *Les conventions sont faites  
pour être brisées.* »



# Cahier des charges d'un bon exporter SNMP

- Un exporter pour de multiples ressources ;

# Cahier des charges d'un bon exporter SNMP

- Un exporter pour de multiples ressources ;
- chaque métrique est récupérée avec un intervalle dédié ;

# Cahier des charges d'un bon exporter SNMP

- Un exporter pour de multiples ressources ;
- chaque métrique est récupérée avec un intervalle dédié ;
- personnalisable dans la nomenclature, parce qu'il n'y a aucun standard ;

# Cahier des charges d'un bon exporter SNMP

- Un exporter pour de multiples ressources ;
- chaque métrique est récupérée avec un intervalle dédié ;
- personnalisable dans la nomenclature, parce qu'il n'y a aucun standard ;
- le moins lié possible à la structure des MIB ;

# Cahier des charges d'un bon exporter SNMP

- Un exporter pour de multiples ressources ;
- chaque métrique est récupérée avec un intervalle dédié ;
- personnalisable dans la nomenclature, parce qu'il n'y a aucun standard ;
- le moins lié possible à la structure des MIB ;
- avec l'intégration des transformations de données.

# Prometheus-enhanced-snmp-exporter

# Premetheus Enhanced SNMP exporter

This projects provides a new implementation of SNMP exporter for prometheus.

This project *is not production ready* it's currently under dev and should be used with car

This project is provided with theses feature

- Multithreading
- Metrics caching : allow the collection of many metrics and equipments
- Label conversion methods
- VRF aware quering (when community should be edited to set the requested vrf)

## usage

```
$ ./prometheus-enhanced-snmp-exporter --help
usage: prometheus-enhanced-snmp-exporter [-h] [-f FILENAME] [-l {debug,info,warning,erro}] [--lister
```

Prometheus SNMP exporter

optional arguments:

- h, --help show this help message and exit
- f FILENAME, --filename FILENAME configuration file to parse
- l {debug,info,warning,erro}, --log-level {debug,info,warning,erro} log level
- c, --check simply check config and exit

# La IFMIB tu scraperas

```
driver:  
  name: prometheus  
hosts:  
- hostname: myswitch  
  version: v2c  
  community: strong  
  modules:  
    - ifmib  
  static_labels:  
    hostname: __hostname  
    rack: 101  
    role: core
```

# La IFMIB tu scraperas

```
modules:  
  ifmib:  
    labels:  
      iface:  
        type: walk  
        every: 5m  
        mappings:  
          name: IF-MIB::ifDescr  
          description: IF-MIB::ifAlias  
    metrics:  
      - type: walk  
        every: 1m  
        mappings:  
          network_iface_in_bytes: IF-MIB::ifHCInOctets  
          network_iface_out_bytes: IF-MIB::ifHCOctets  
          network_iface_speed: IF-MIB::ifHighSpeed  
          network_iface_operational_status: IF-MIB::ifOperStatus  
        append_tags:  
          if
```

# La IFMIB tu scraper

## description:

**network\_iface\_in\_bytes:**

**type:** counter

**description:** The total number of octets received on the interface

**network\_iface\_out\_bytes:**

**type:** counter

**description:** The total number of octets transmitted out of the interface

**network\_iface\_speed:**

**type:** counter

**description:** An estimate of the interface's current bandwidth in units of 1

**network\_iface\_operational\_status:**

**type:** gauge

**description:** The current operational state of the interface

# Gérer les VRFs

```
modules:  
  arista_bgp_vrf:  
    template_labels:  
      vrf:  
        type: walk  
        every: 5m  
        community_template: "{community}@{template}"  
        mapping: ARISTA-VRF-MIB::aristaVrfState  
        store_method: subtree-as-string  
    labels:  
      bgp_peer:  
        type: walk  
        evry: 1m  
        template_label: vrf  
        mappings:  
          local_addr:  
            oid: ARISTA-BGP4V2-MIB::aristaBgp4V2PeerLocalAddr  
            store_method: hex-as-ip
```

# Gérer les VRFs

```
metrics:  
  - type: walk  
    every: 1m  
    template_label: vrf  
    mappings:  
      remote_addr:  
        oid: ARISTA-BGP4V2-MIB::aristaBgp4V2PeerLocalAddr  
        store_method: subtree-as-ip  
    append_tags:  
      - .bgp_peer  
      - __template_label
```

# Les transformations de données

L'attribut `store_method` permet de modifier la manière dont une MIB est stockée.

`subtree-as-string` convertit le sous chemin en chaîne de caractères

`subtree-as-ip` convertit le sous chemin en IPv4

`value` utilise la valeur comme tel

`hex-as-ip` convertit la valeur hexadecimale en IPv4

`hex-as-mac` convertit la valeur hexadecimale en MAC

`extract_realm` récupère le royaume depuis un email/login PPP

`milli` divise par 1000 le résultat

# Récupérer la consommation électrique

C'est de loin le besoin qui m'a demandé le plus de features :

- la fonctionnalité de filtres de données ;
- les jointures ;
- la récupération des paramètres lors des filtres.

# Consommation électrique : setup d'InfluxDB

```
driver:  
  name: InfluxDB  
  config:  
    hostname: infludbx  
    db: power  
    username: power  
    password: NotARealPassword  
hosts:  
  - hostname: myIosXRRouter  
    version: v2c  
    community: public  
    modules:  
      - power_asr
```

# Consommation électrique : jointure

```
modules:  
  power_asr_9001:  
    labels:  
      I:  
        type: walk  
        every: 5m  
        mappings:  
          name:  
            oid: ENTITY-MIB::entPhysicalName  
            filter: "current 0/PS0/M([0-9])/SP"  
      V:  
        type: walk  
        every: 5m  
        mappings:  
          name:  
            oid: ENTITY-MIB::entPhysicalName  
            filter: "voltage 0/PS0/M([0-9])/SP"
```

# Consommation électrique : jointure

```
W:  
  type: join  
  every: 1m  
  mappings:  
    I: name  
    V: name
```

# Consommation électrique : métriques

```
metrics:  
  - type: walk  
    every: 1m  
    mappings:  
      power_current:  
        oid: CISCO-ENTITY-SENSOR-MIB::entSensorValue  
        store_method: milli  
    append_tags:  
      - .W.I  
  - type: walk  
    every: 1m  
    mappings:  
      power_voltage:  
        oid: CISCO-ENTITY-SENSOR-MIB::entSensorValue  
        store_method: milli  
    append_tags:  
      - .W.V
```

# Consommation électrique : intégration à InfluxDB

```
description:  
  power_current:  
    type: power  
    description: current  
  power_voltage:  
    type: power  
    description: voltage
```

# Questions ?



Le projet



Les impressions 3D