Vagando por las lindes de la Telemetria (con OpenNTI-new)

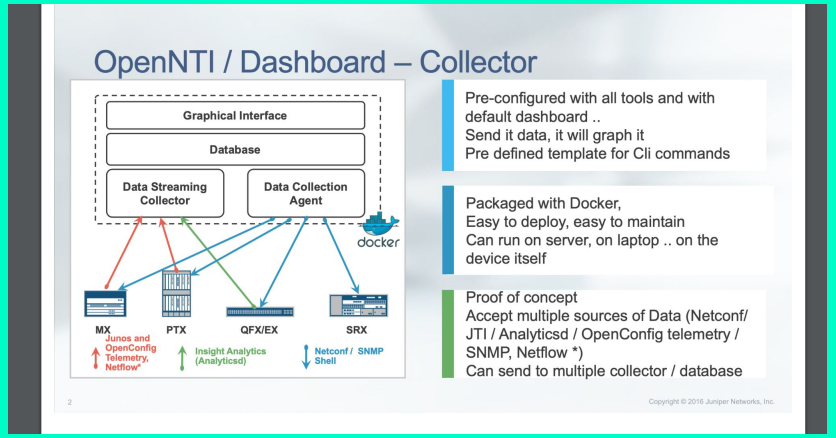
Juan P. Cerezo

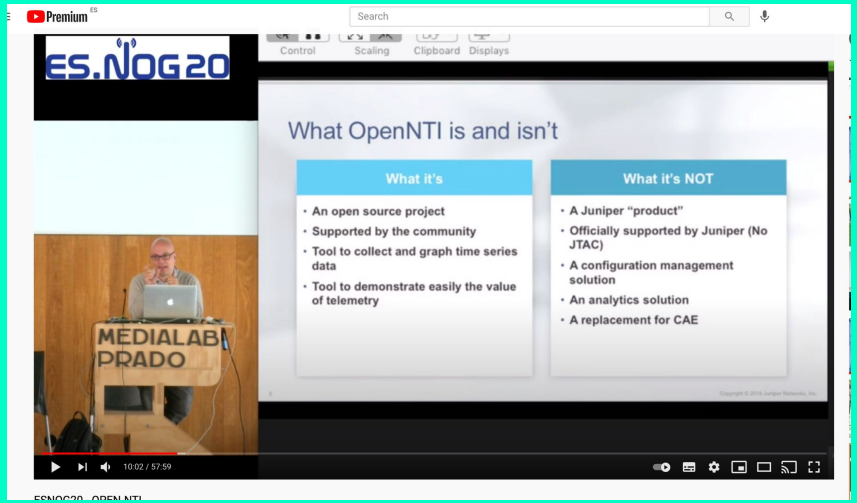
¿Posibles usos de una herramienta open en un entorno real?
Es rentable?
Es Util?

Antecedentes

EN EPISODIOS ANTERIORES DE ESNOG...

OPEN-NTI en el GORE-20





Que ha cambiado?

- La idea original se mantiene
- actualización de versiones
- nuevas funcionalidades
- mismo soporte
- mejoras en el despliegue y usabilidad?

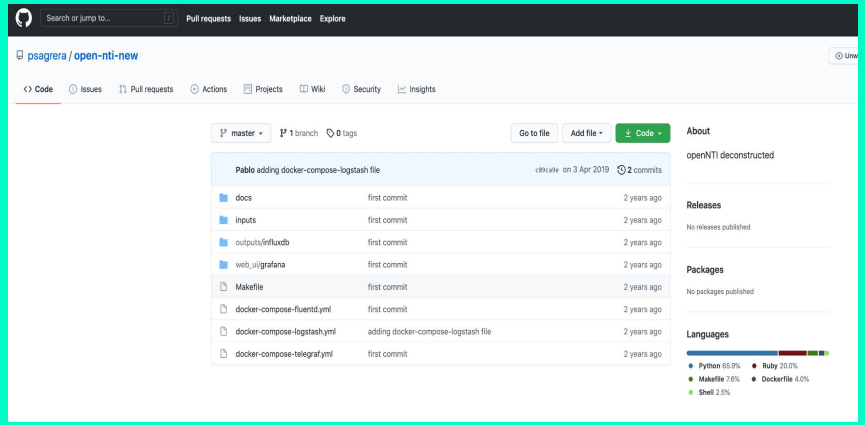
Deconstrucción

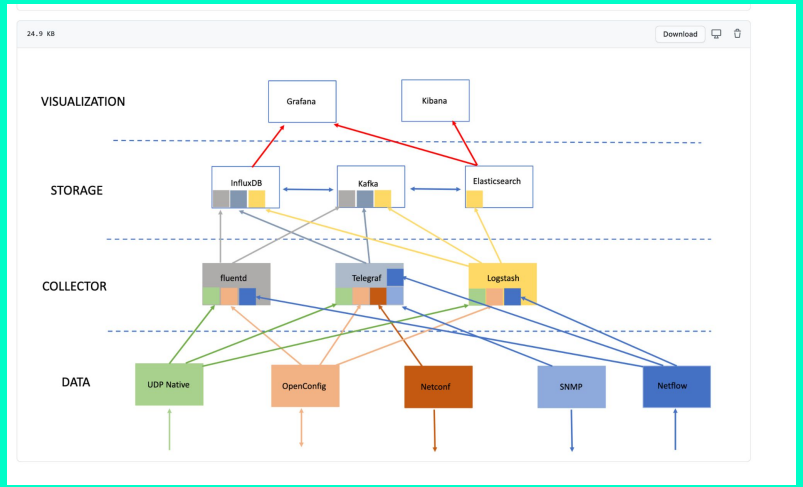
open-nti-new

- proyecto personal
- modularizar los componentes
- optimizar la seleccion de escenarios
- mismo soporte (best effort)
- mejoras en el despliegue y usabilidad (SI!)

Deconstrucción

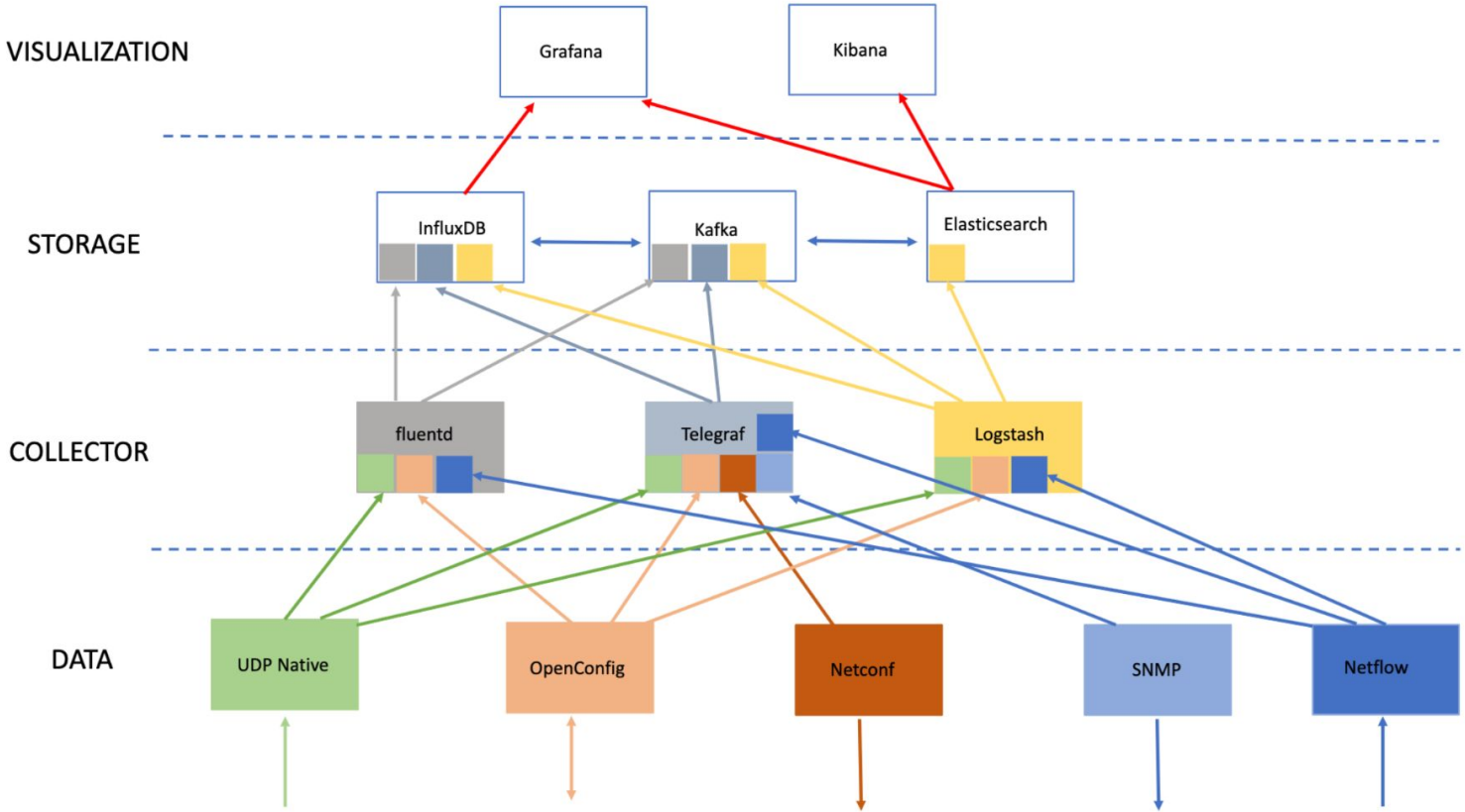
[open-nti-new]





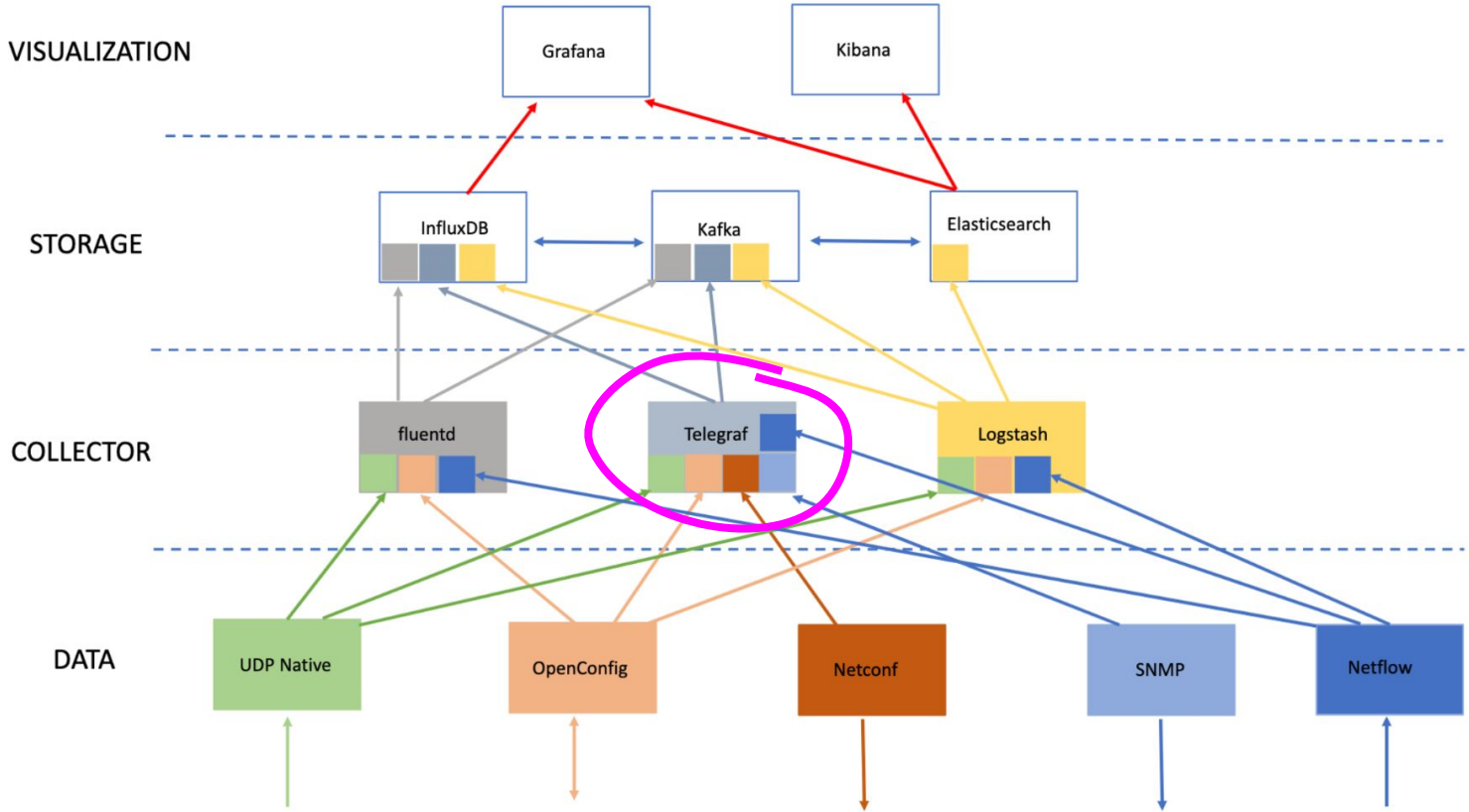
24.9 KB

Download



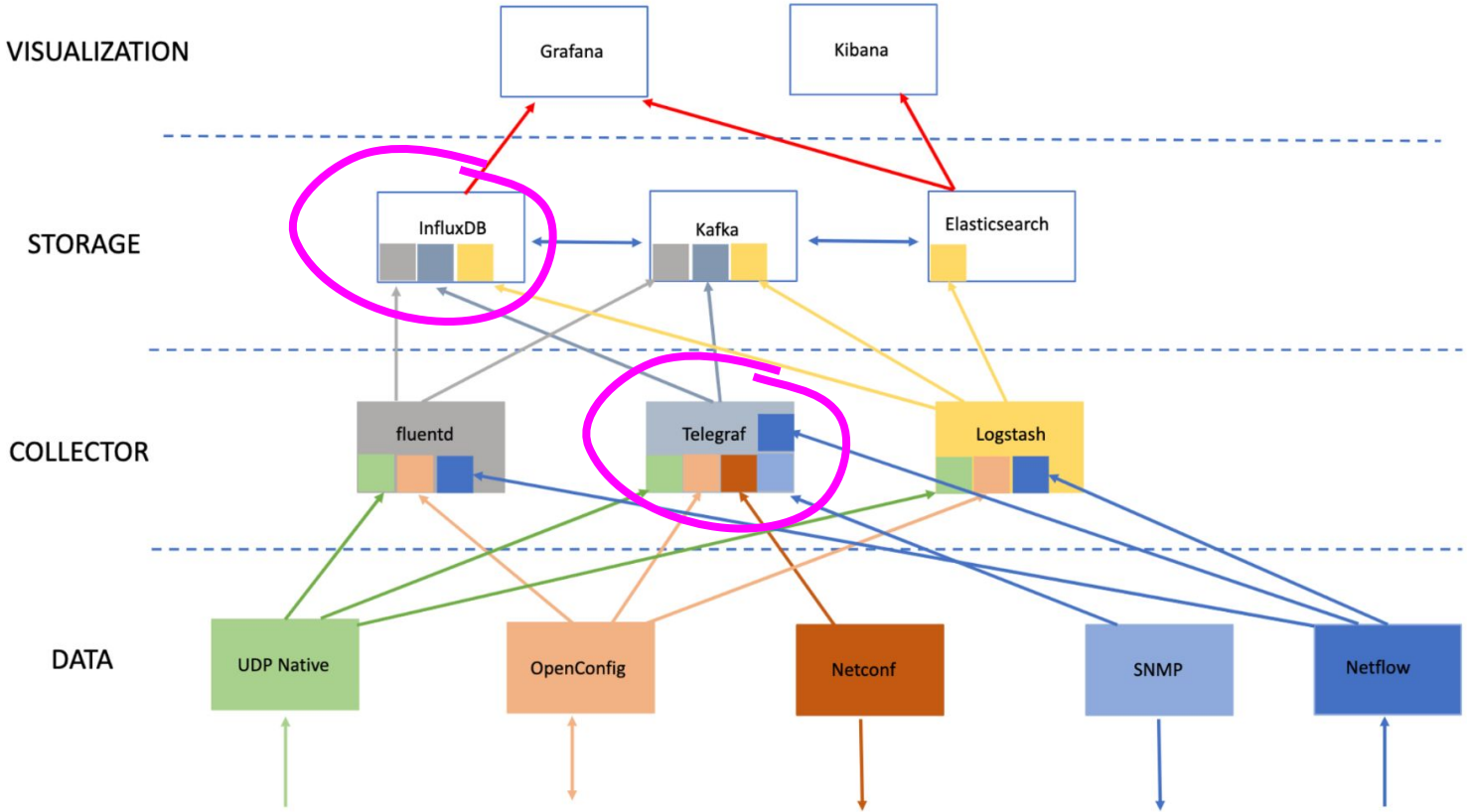
24.9 KB

Download



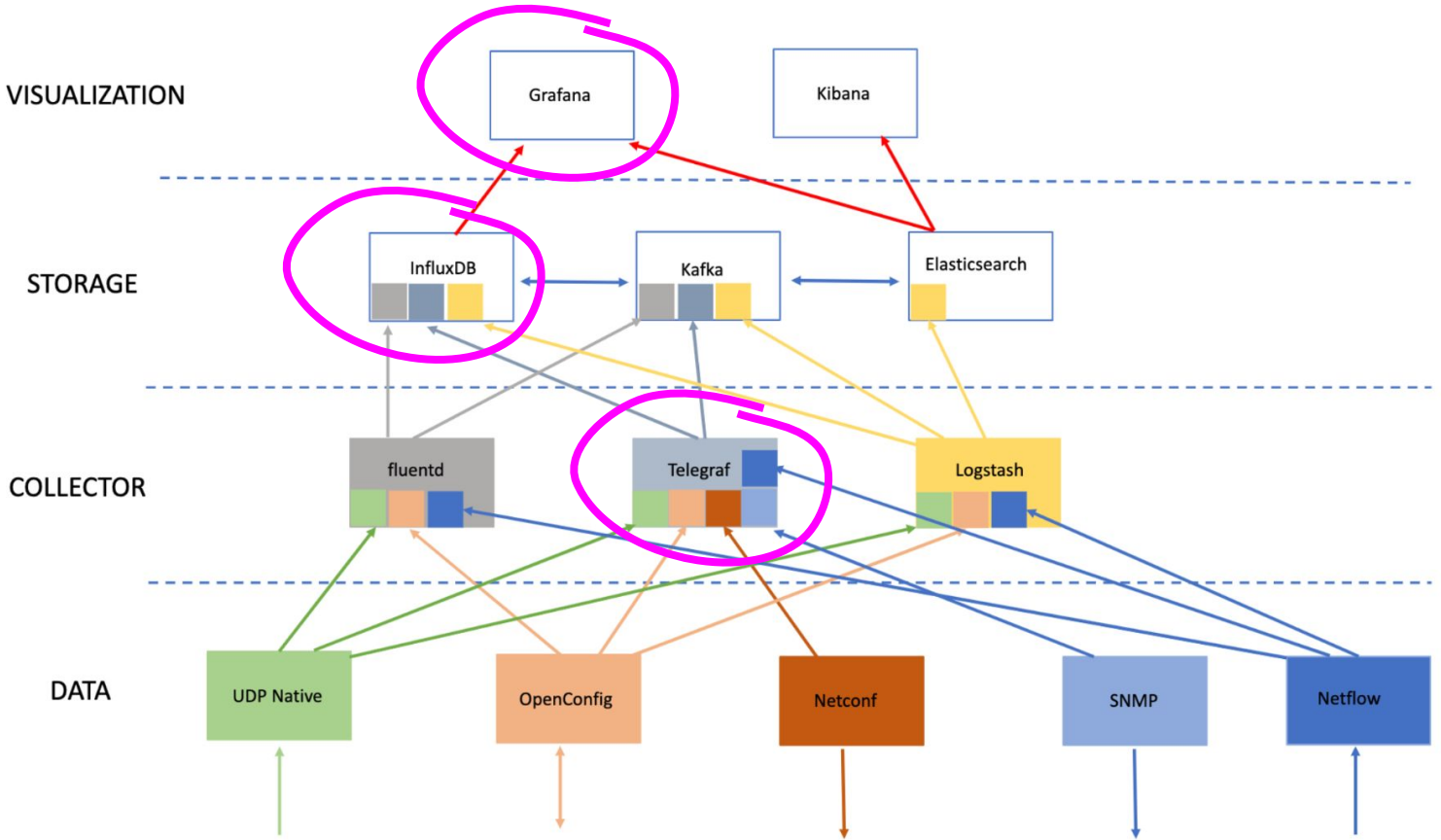
24.9 KB

Download



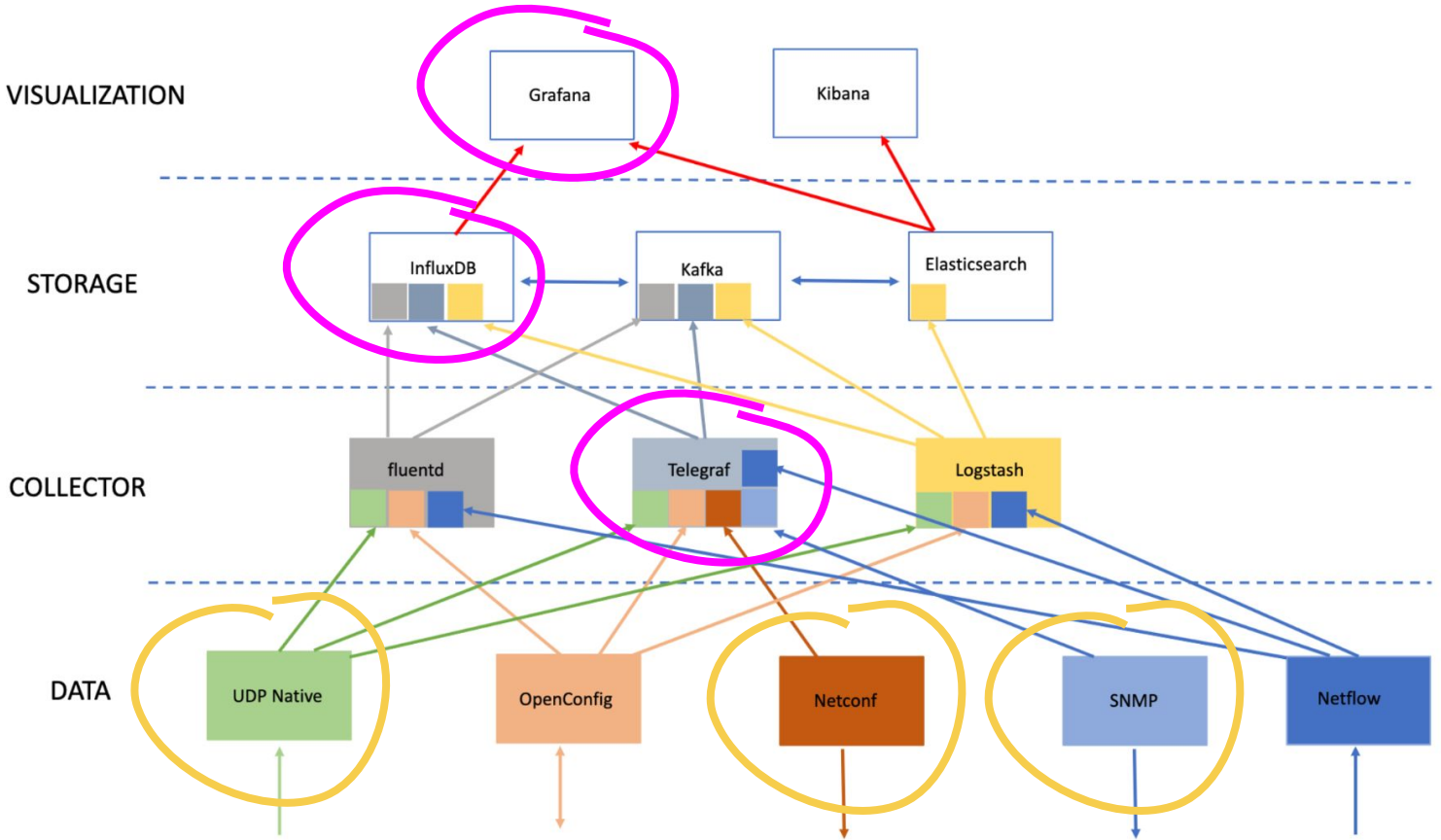
24.9 KB

Download



24.9 KB

Download



Puesta en Marcha

Material

Lo tienes todo en casa

- una red de equipos (de red)
- docker
- git, python
- code editor & test
- un equipo sobre que trabajar (preferiblemente ubuntu)

Hipótesis

Requerimientos

1. Red ISP de unos 150 equipos (Juniper) de red
2. Login (2 diferentes)
3. Diferentes funcionalidades a monitorizar: **BNG, CG-NAT, IXP, chassis & RE**
4. Posibilidad de realizar un chequeo de red permanente (**HealthCheck**) durante periodos específicos
5. Ortogonal a otras mediciones o servicios de monitorización

Instalacion
y
puesta en marcha

ADAPTACIONES

Personalización de la base de equipos

Usando los formatos habituales, configurando hosts, grupos, autentificación… para el tipo de colector/ingestión de datos seleccionado

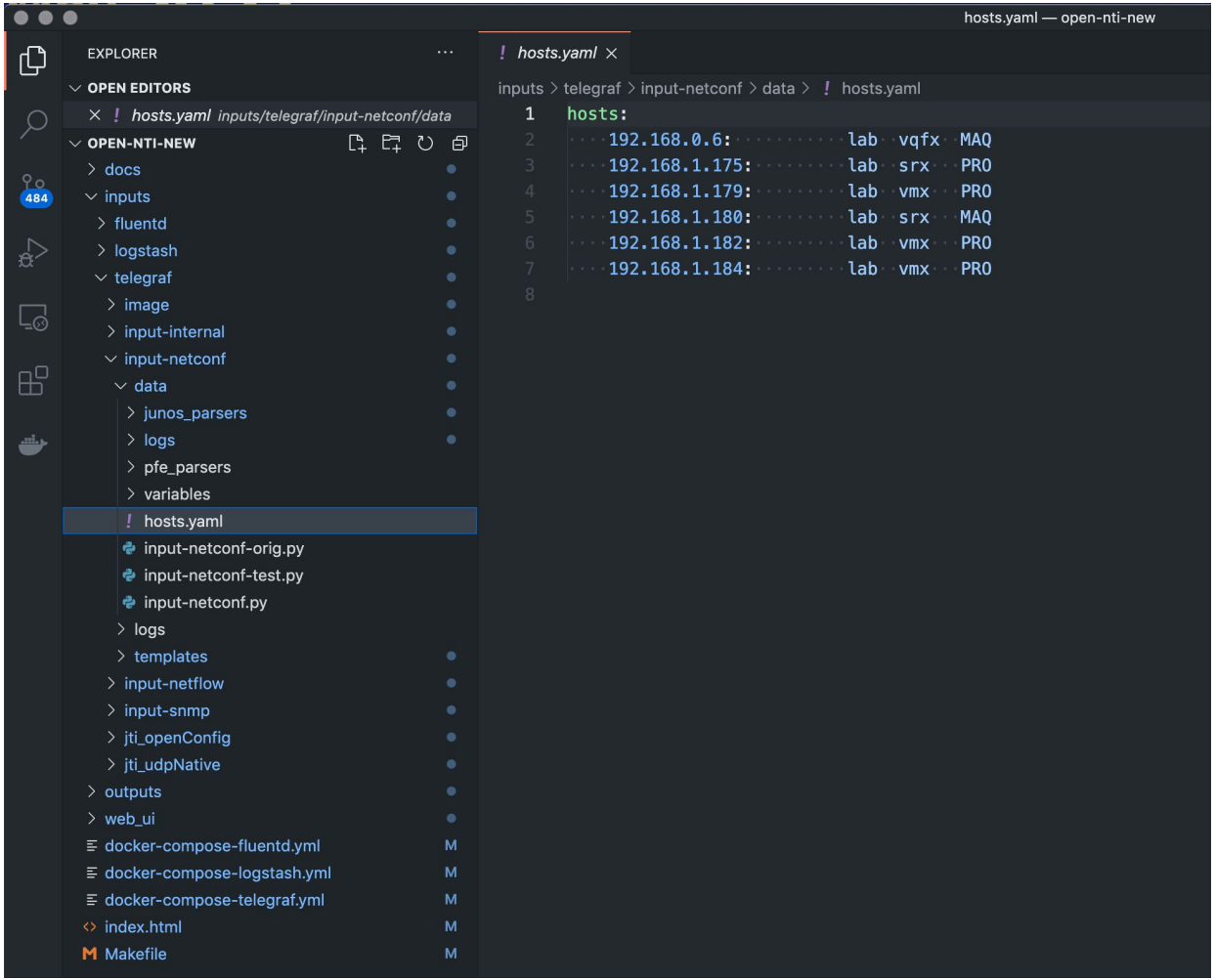
Personalización de los KPIs

- Parámetros genéricos de red a monitorizar
- Parámetros específicos para cada grupo de funcionalidades: IXP/routing, BNG, CG-NAT, babysitting

Ejemplo basado en Netconf, proceso similar para jti, snmp, openconf

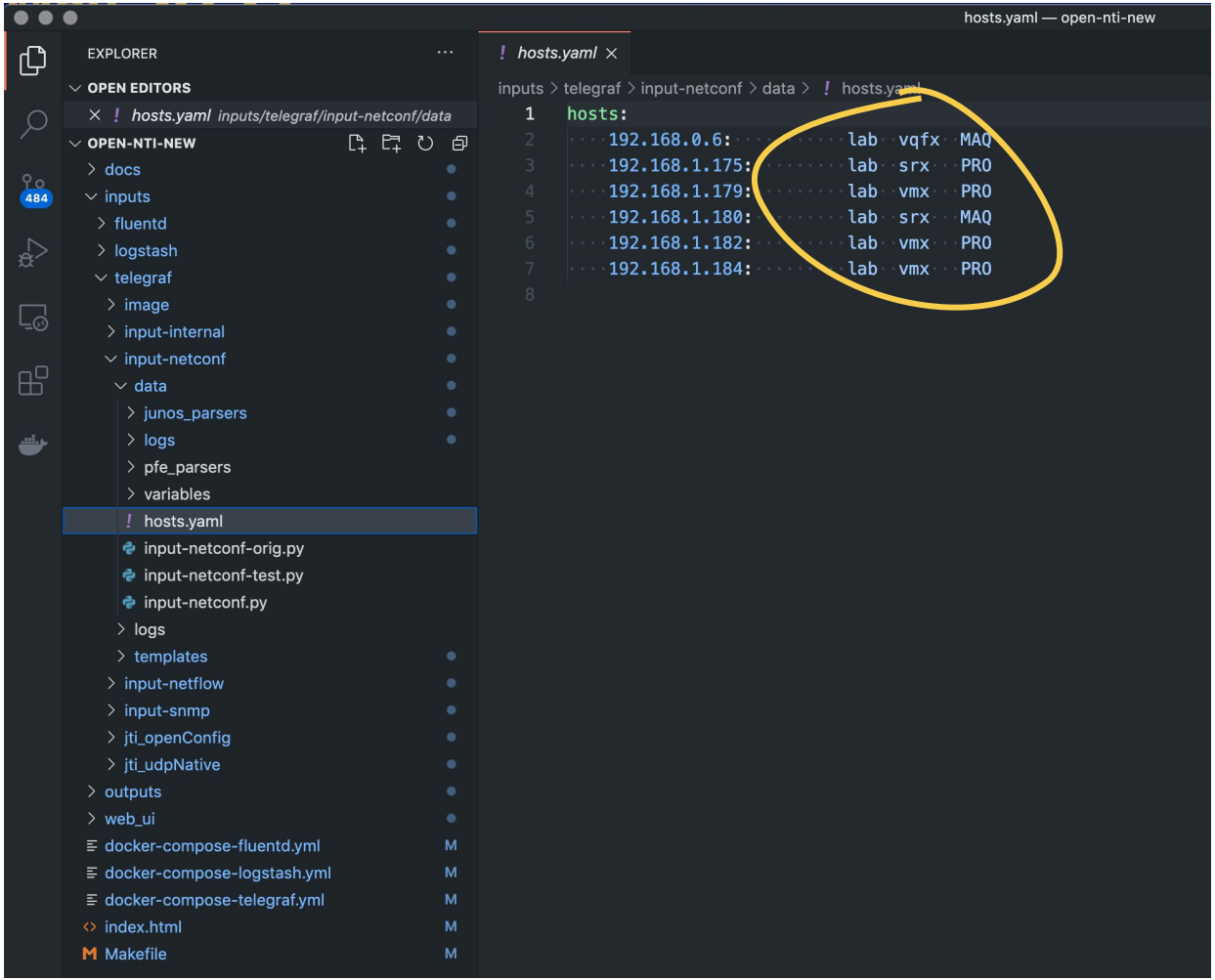
ADAPTACIONES

HOSTS



ADAPTACIONES

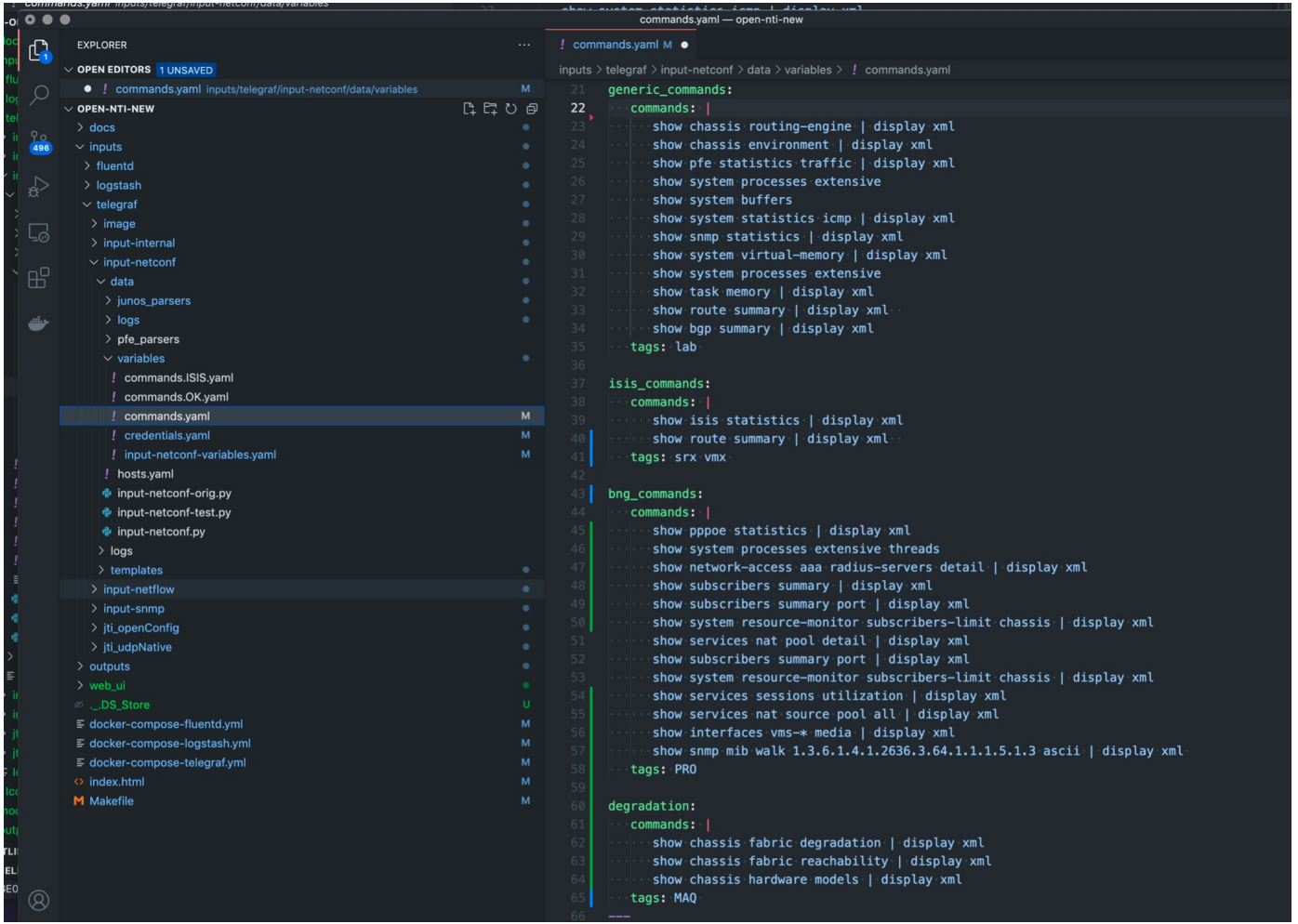
HOSTS



ADAPTACIONES

HOSTS

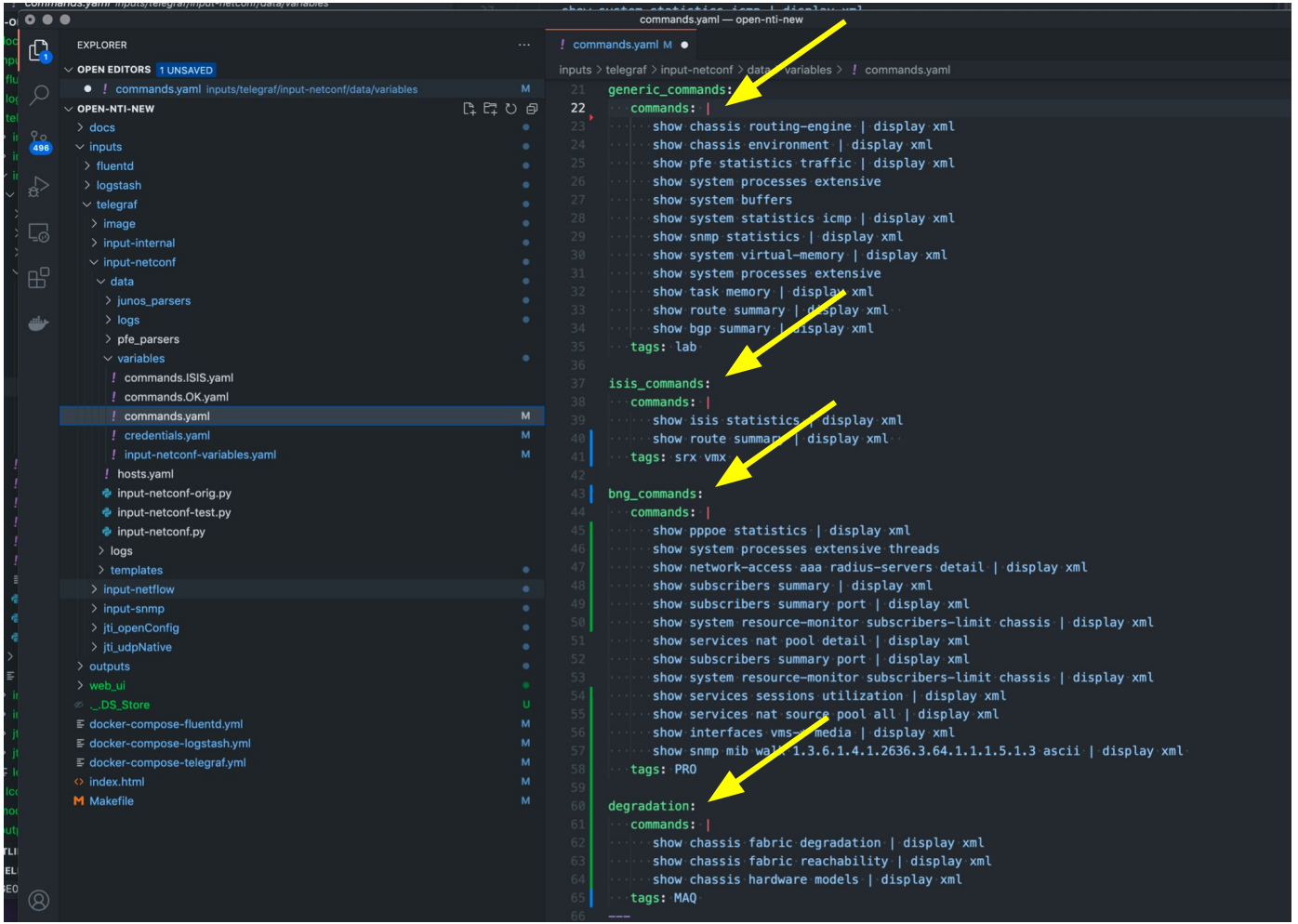
COMANDOS



ADAPTACIONES

HOSTS

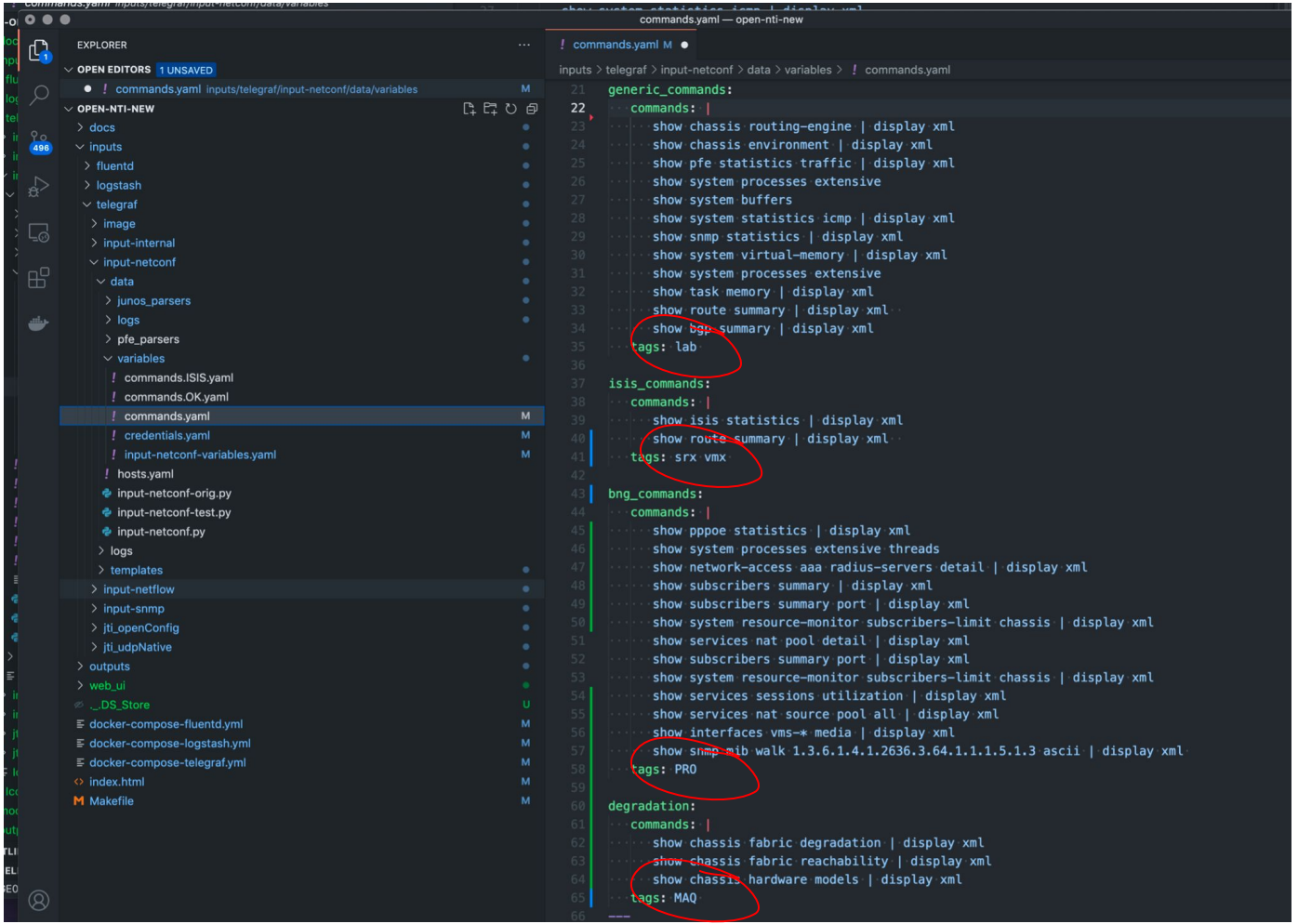
COMANDOS



ADAPTACIONES

HOSTS

COMANDOS

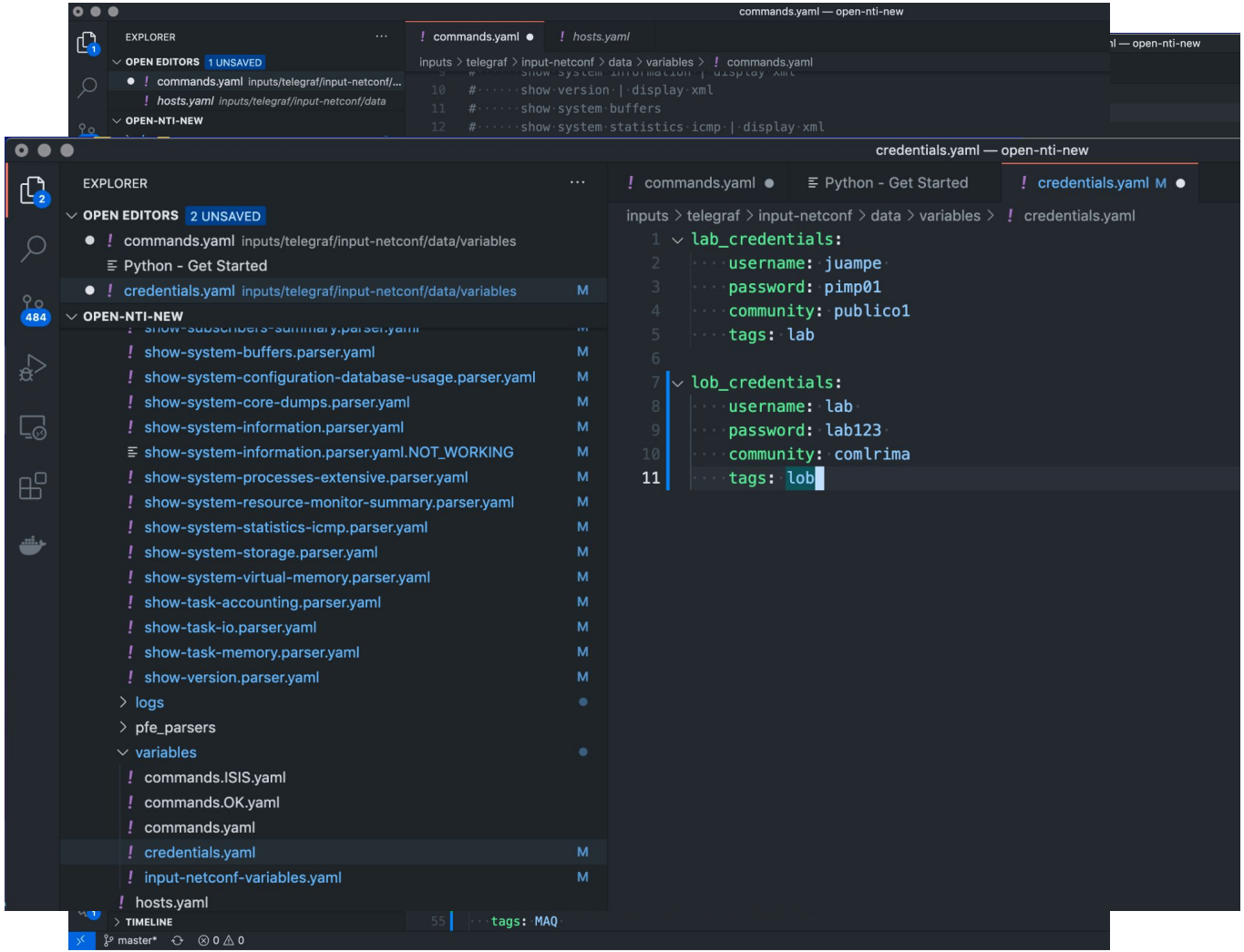


ADAPTACIONES

HOSTS

COMANDOS

AUTENTIFICACIÓN

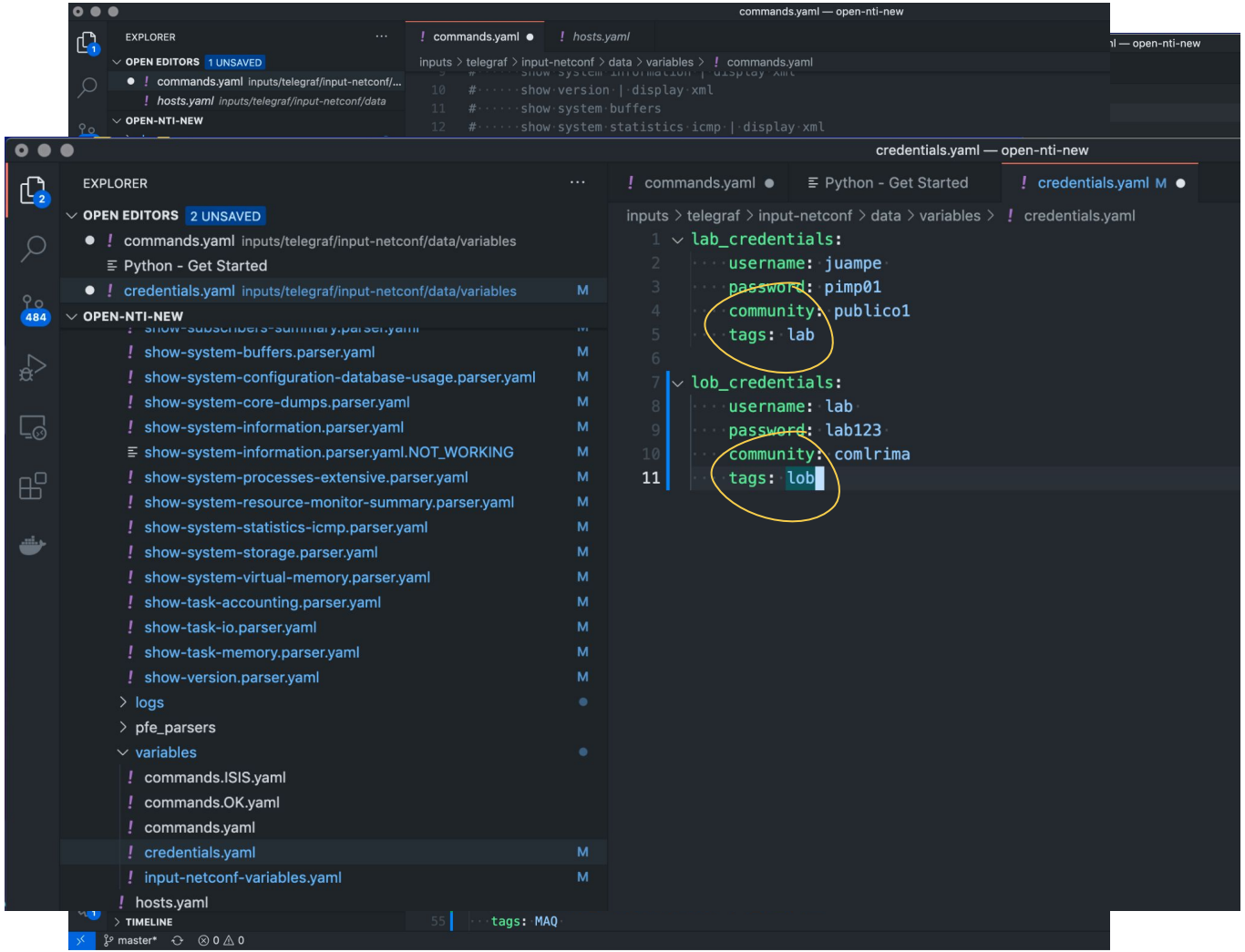


ADAPTACIONES

HOSTS

COMANDOS

AUTENTIFICACIÓN



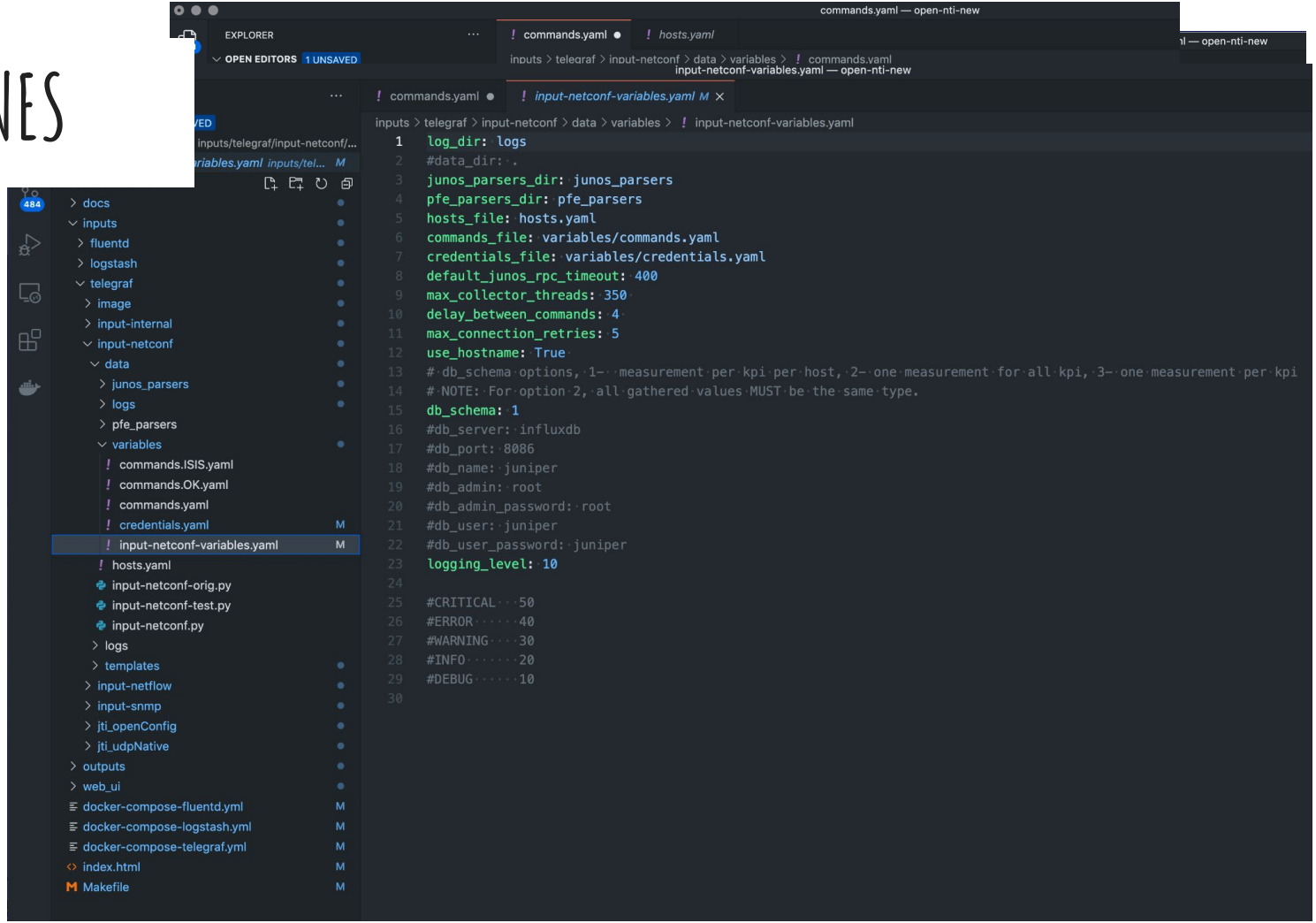
ADAPTACIONES

HOSTS

COMANDOS

AUTENTIFICACIÓN

EJECUCION



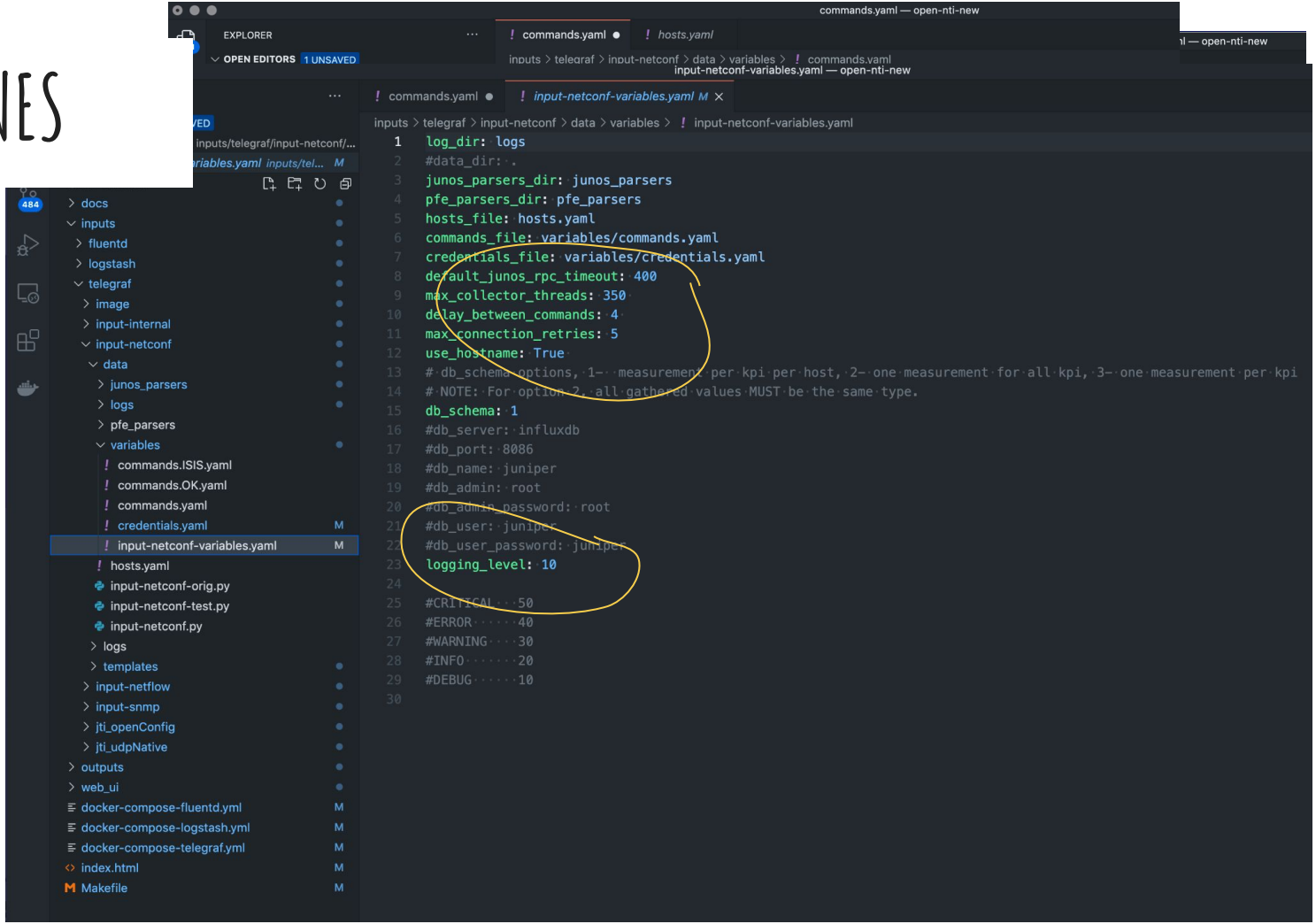
ADAPTACIONES

HOSTS

COMANDOS

AUTENTIFICACIÓN

EJECUCION

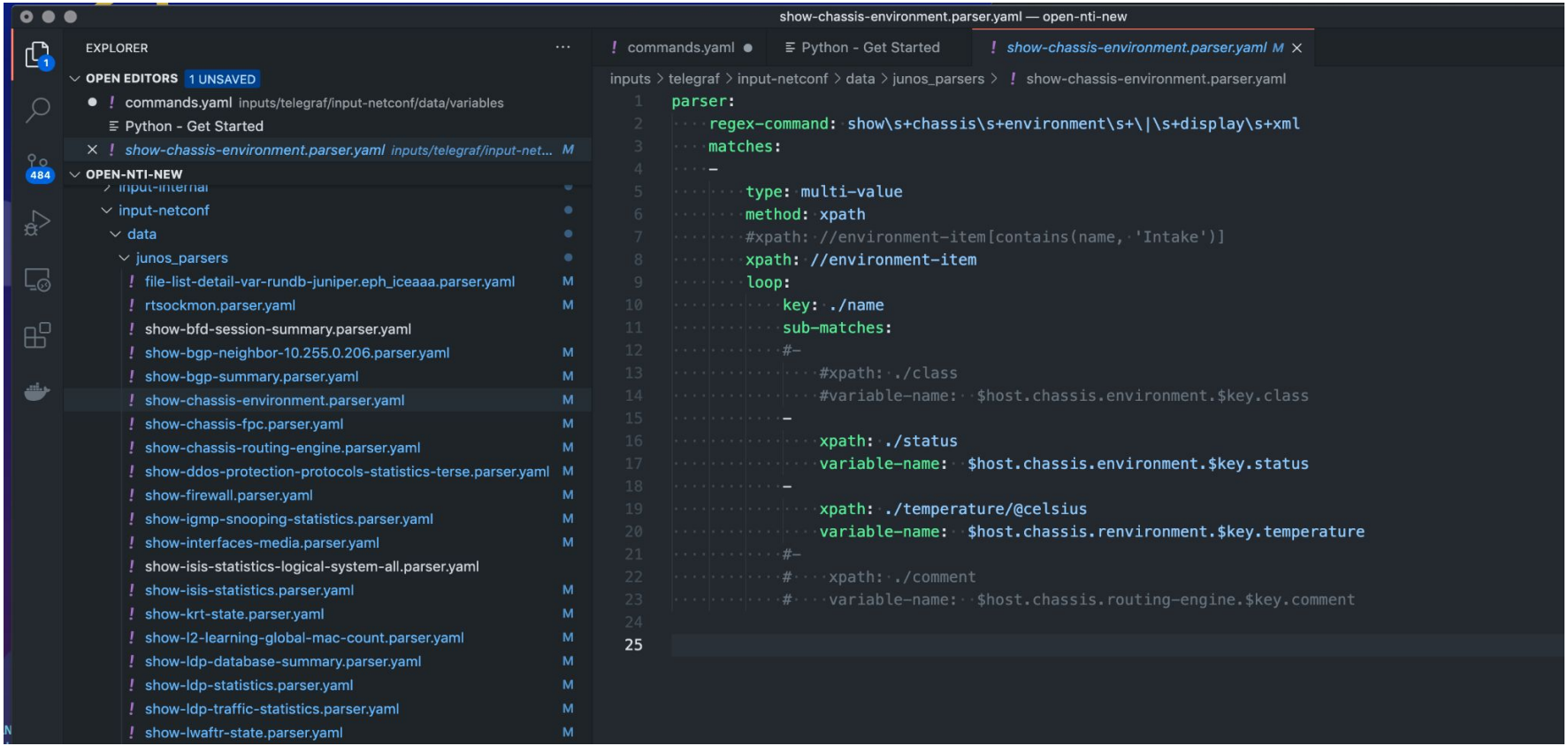


ADAPTACIONES

Personalización de los KPIs a través de **PARSERS**

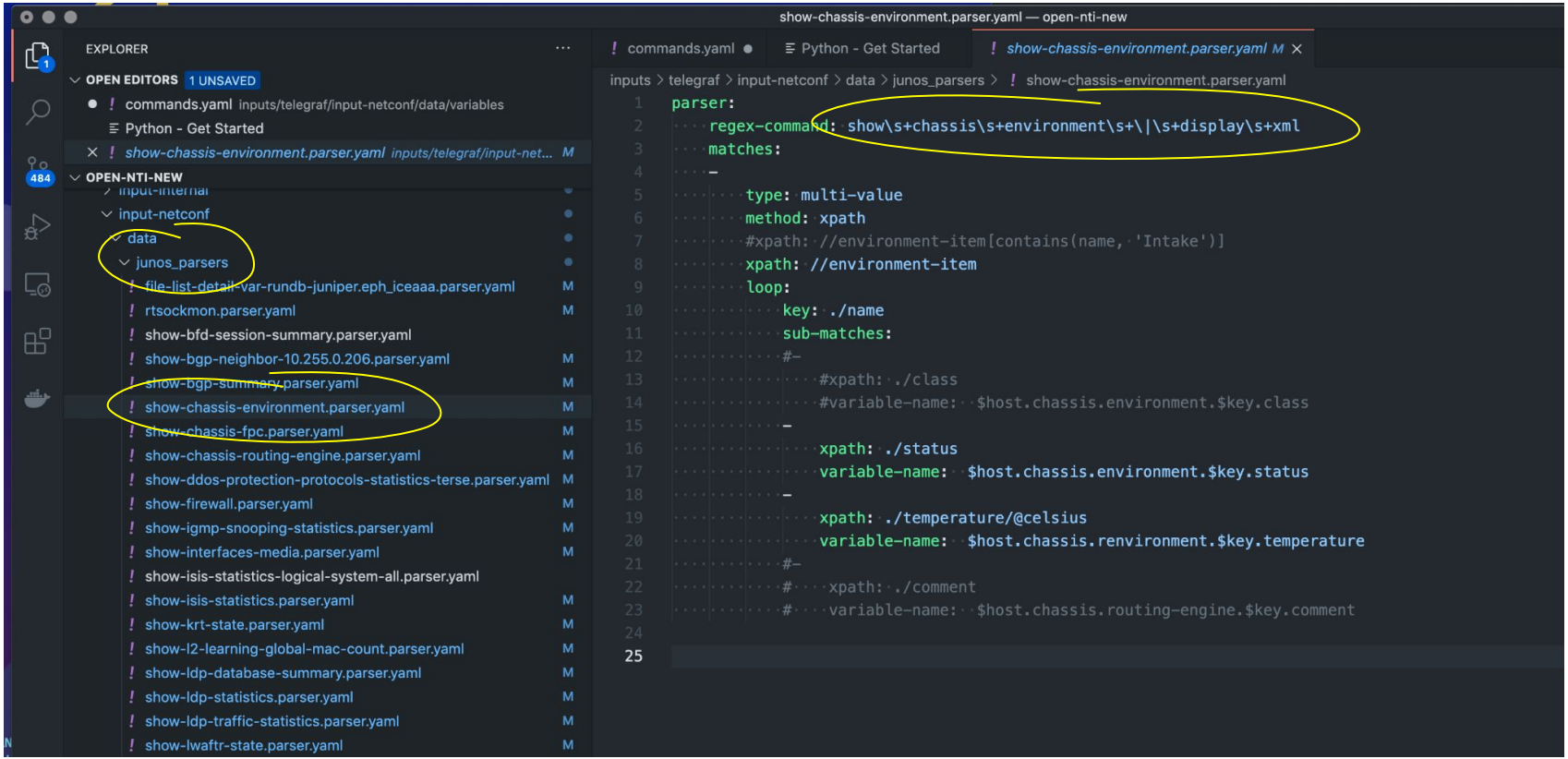
ADAPTACIONES

Personalización de los KPIs a través de PARSERS



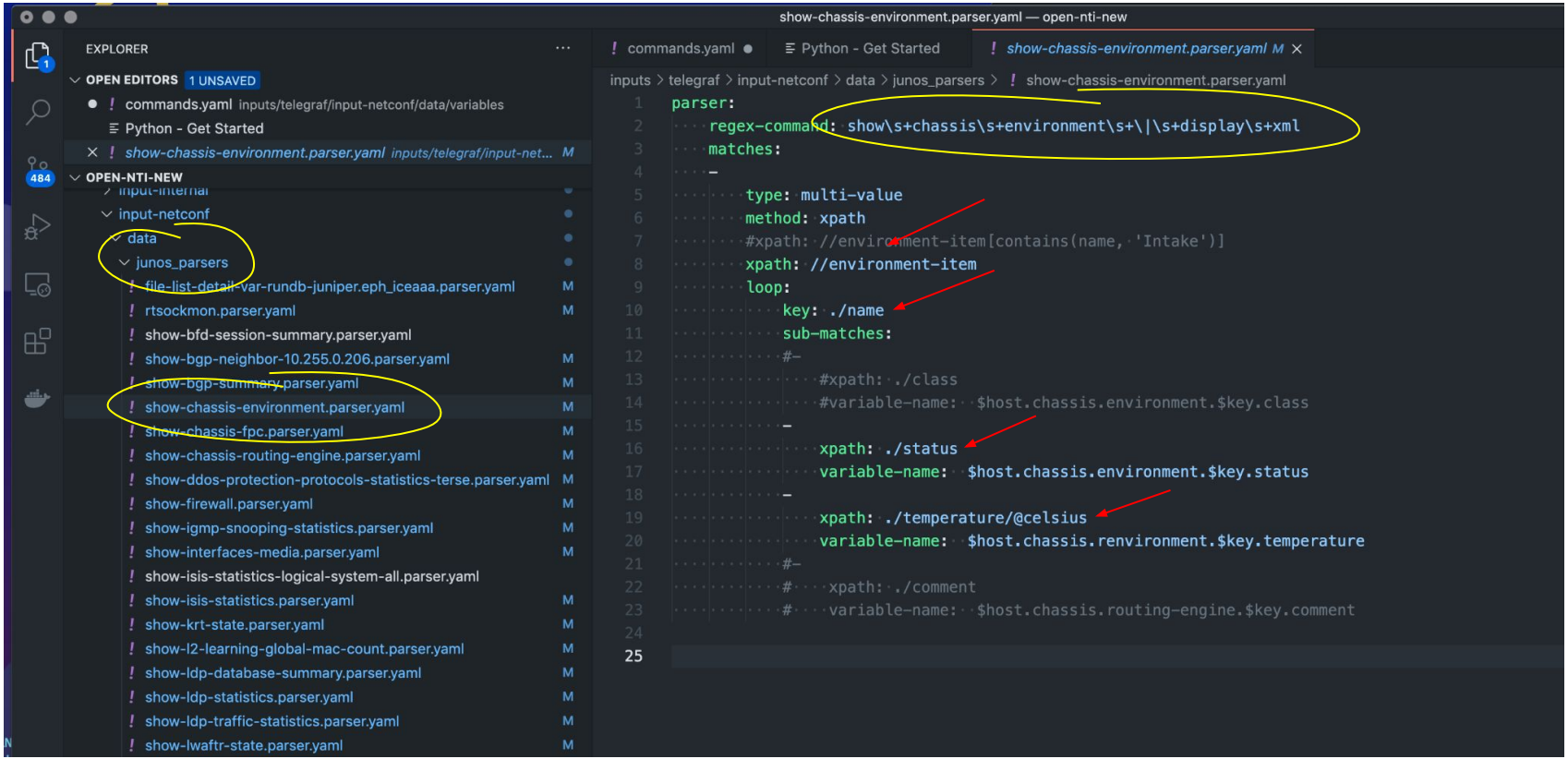
ADAPTACIONES

Personalización de los KPIs a través de PARSERS



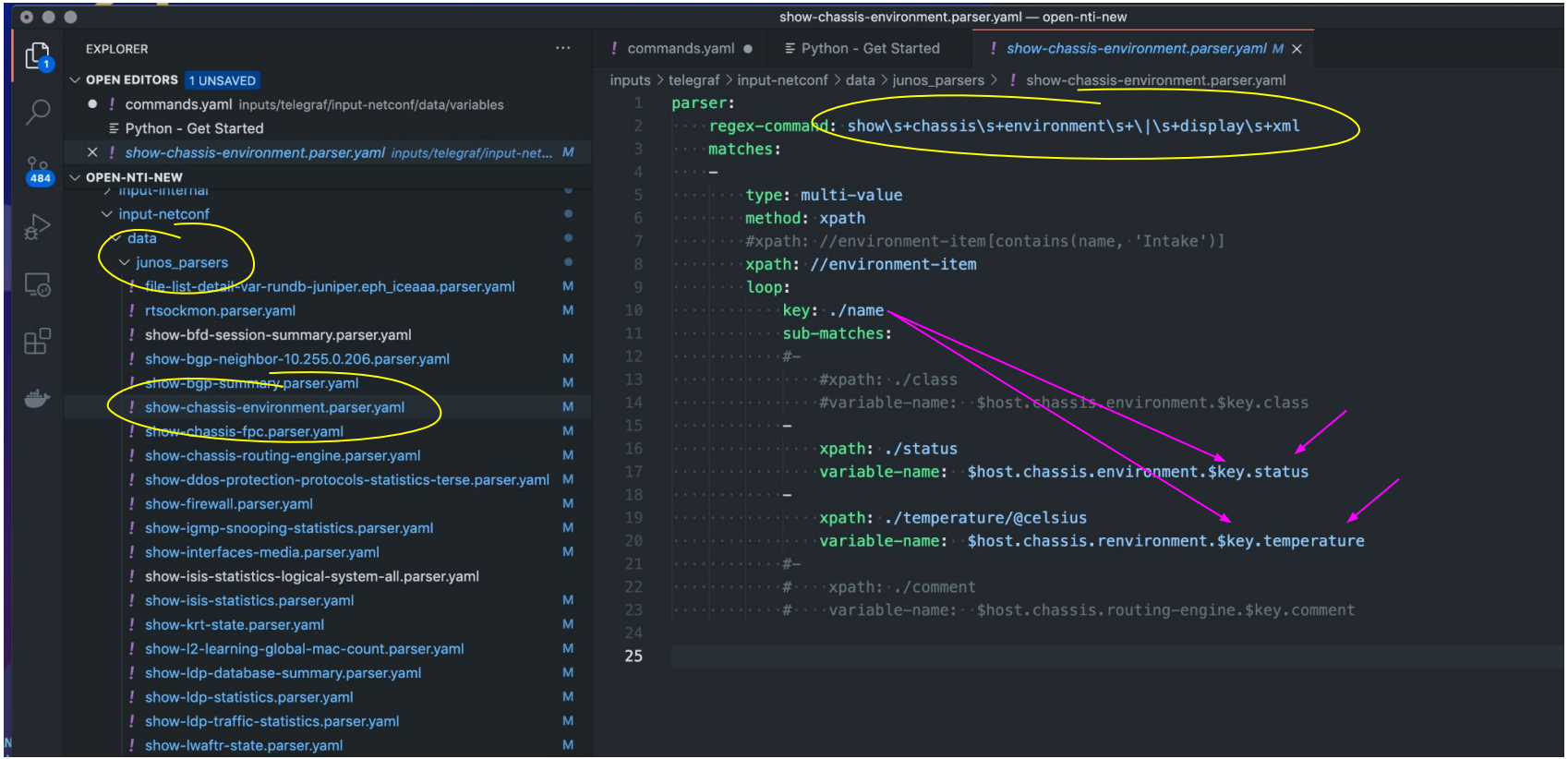
ADAPTACIONES

Personalización de los KPIs a través de PARSERS



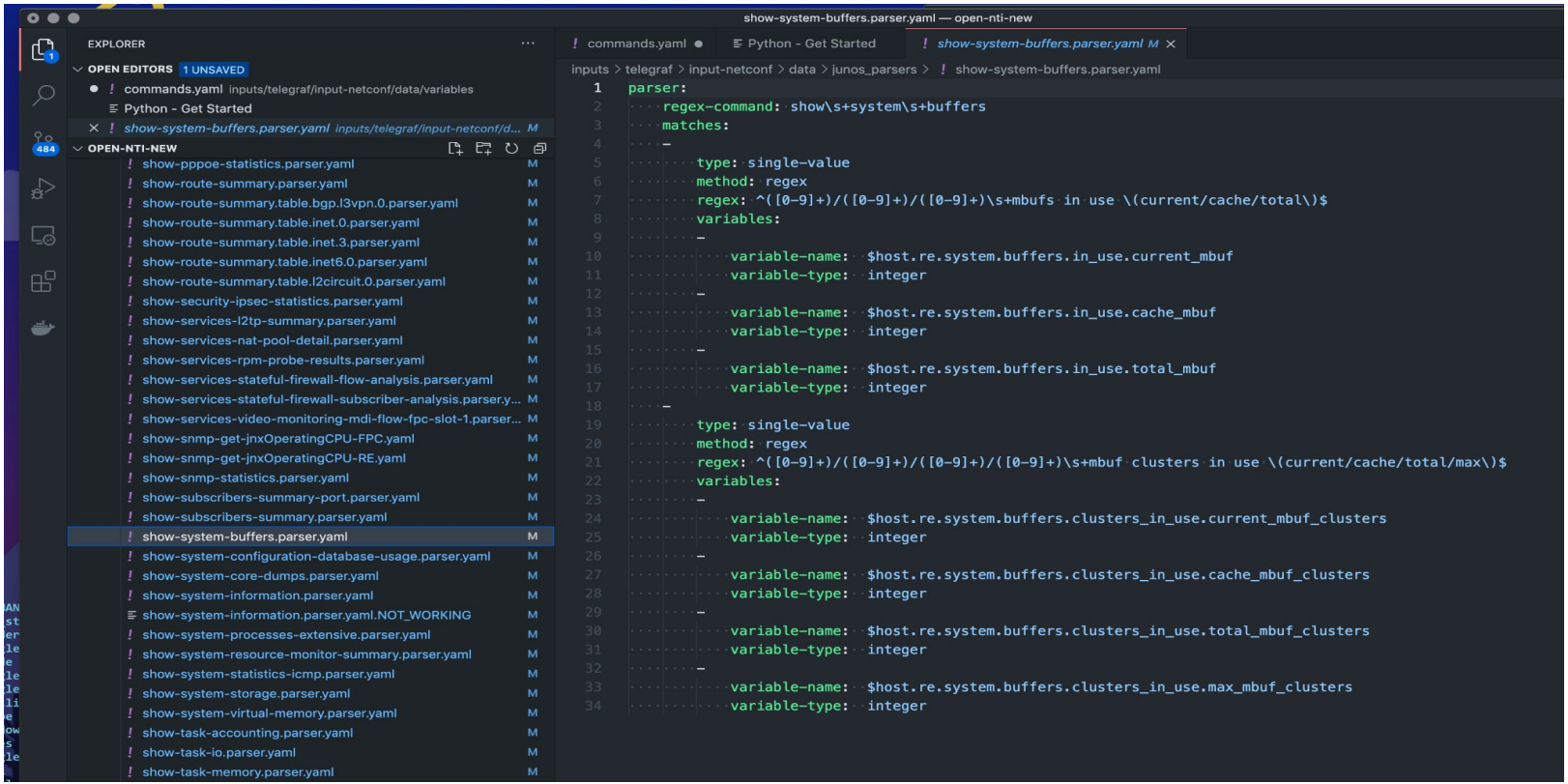
ADAPTACIONES

Personalización de los KPIs a través de PARSERS



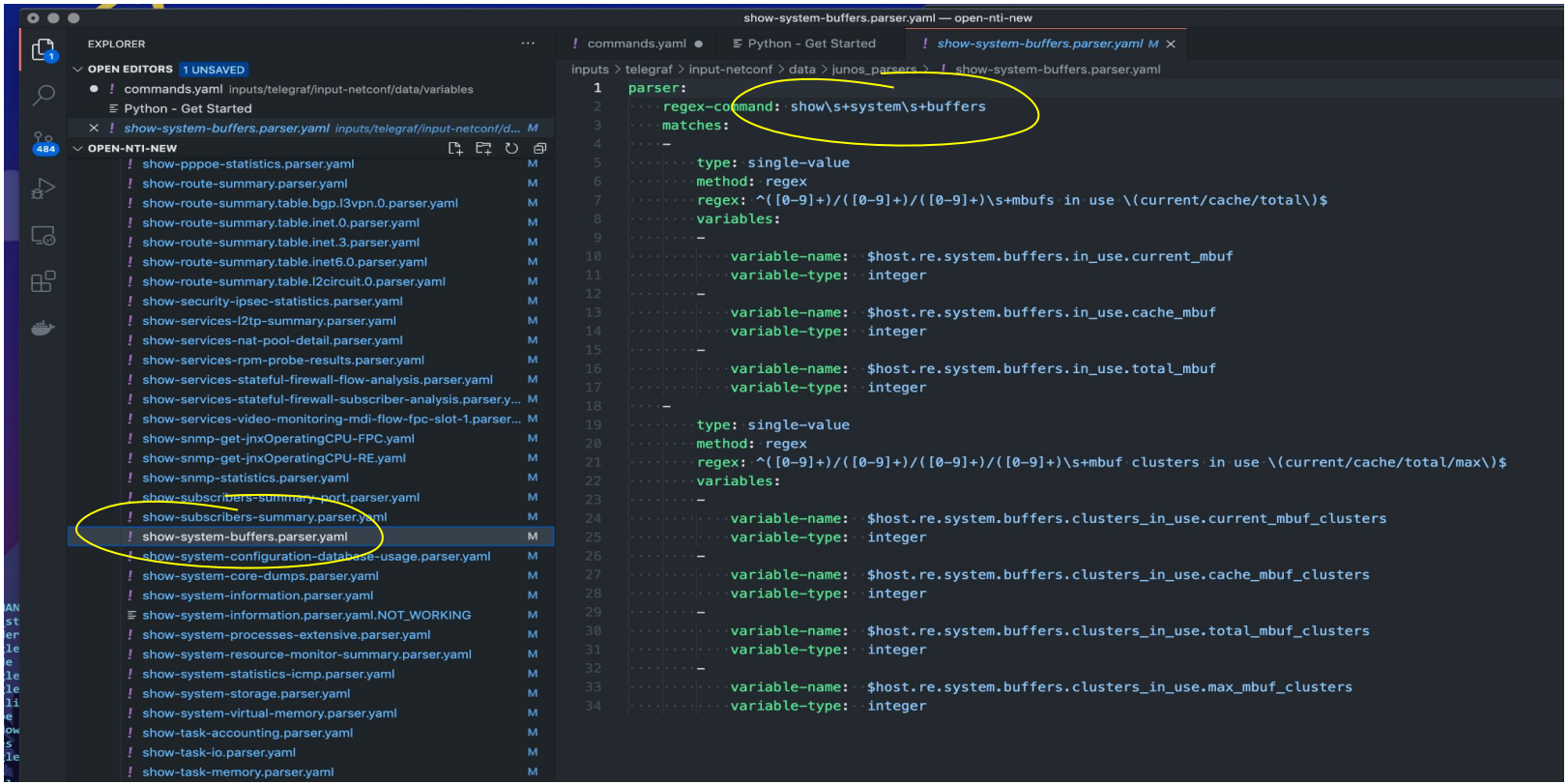
ADAPTACIONES

Personalización de los KPIs a través de PARSERS



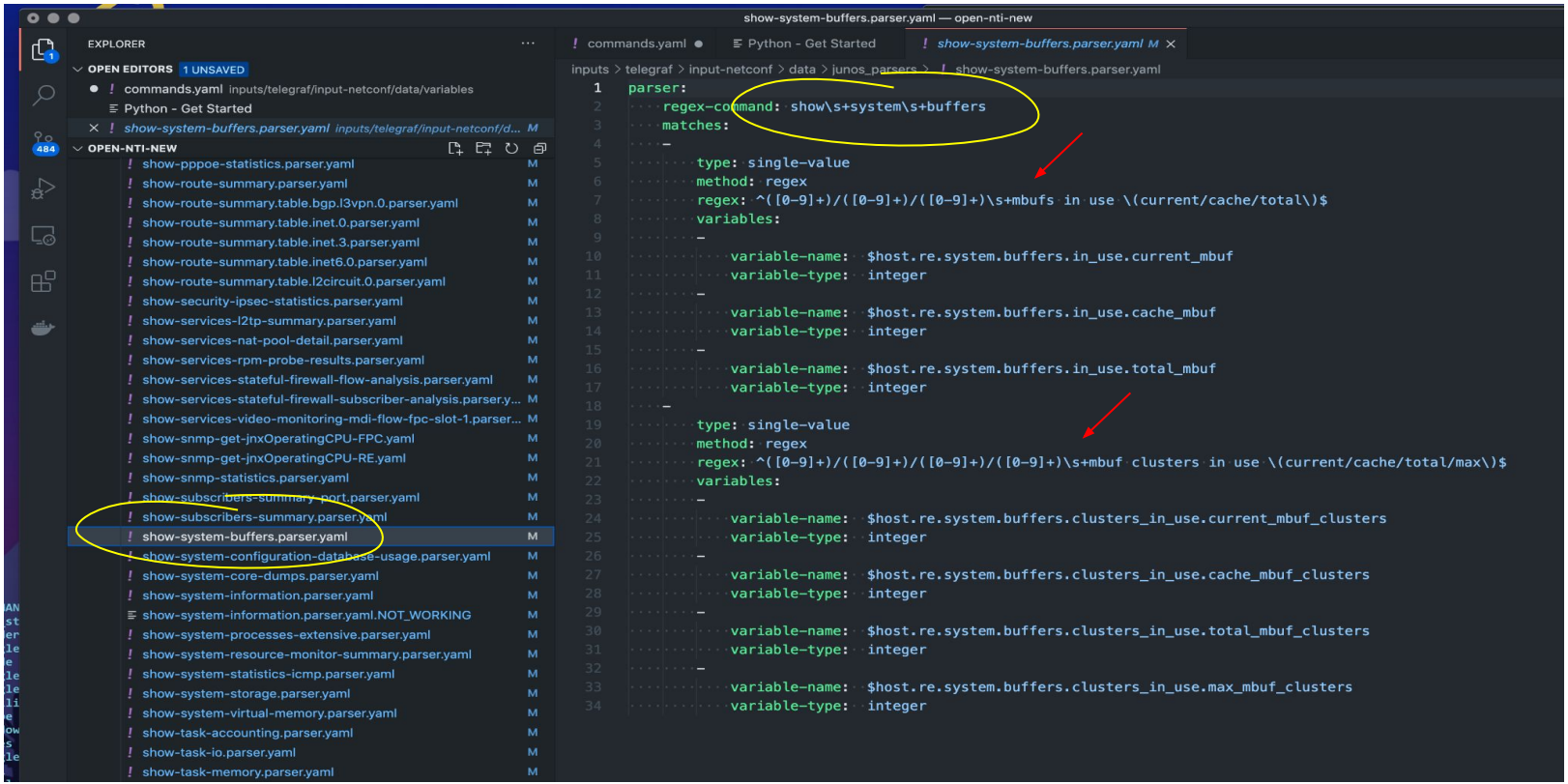
ADAPTACIONES

Personalización de los KPIs a través de PARSERS



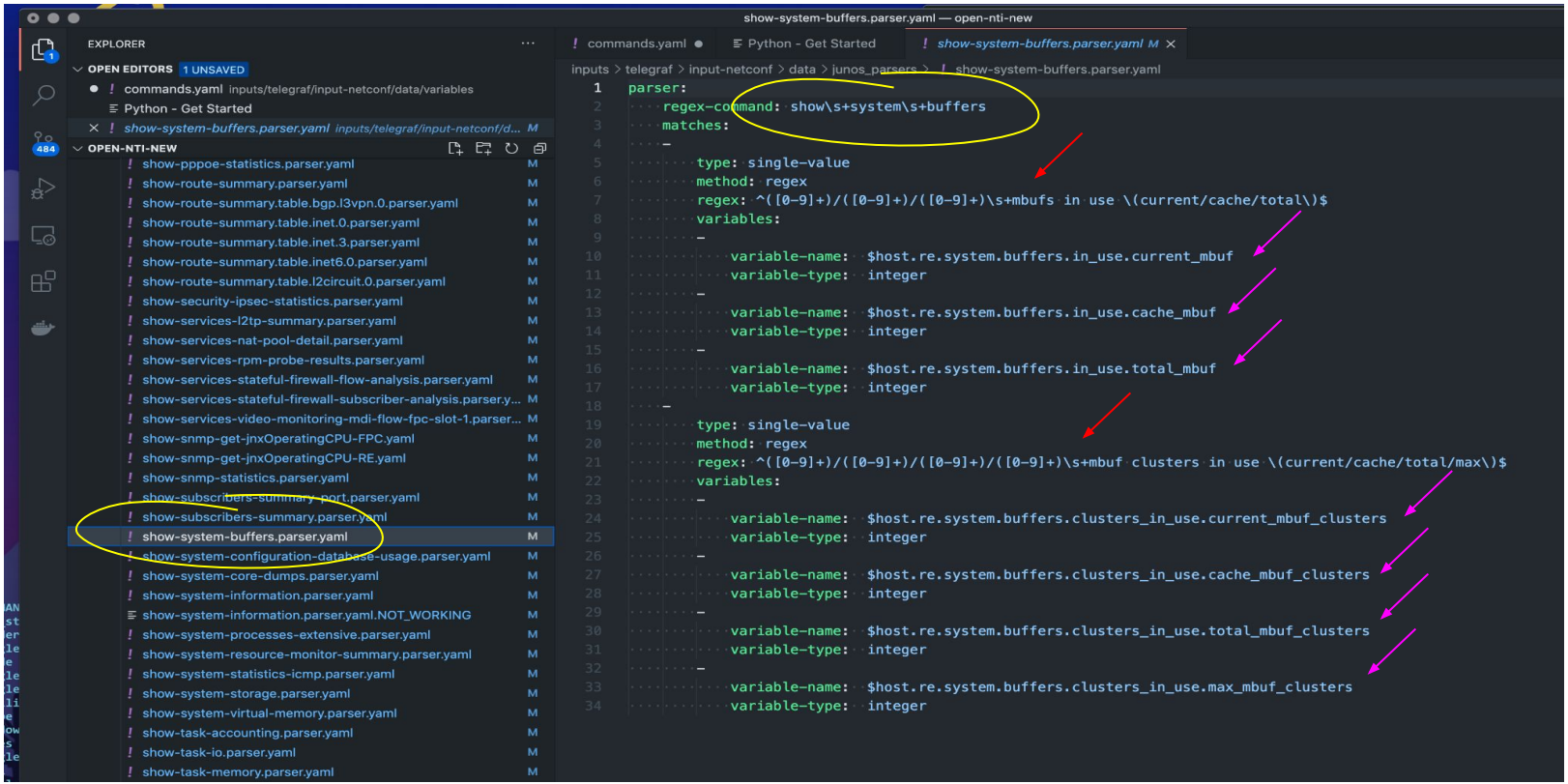
ADAPTACIONES

Personalización de los KPIs a través de PARSERS



ADAPTACIONES

Personalización de los KPIs a través de PARSERS



ADAPTACIONES

Personalización del setup Docker

ADAPTACIONES

Entorno docker-compose por cada configuración, 3 preconfigurados:

docker-compose-fluentd.yml
docker-compose-logstash.yml
docker-compose-telegraf.yml

Especificar:

1. Colectores a utilizar (*telegraf*)
2. DB y herramientas (*influxdb, chronograph, karabiner*)
3. GUI (*grafana*)
4. Otros (*grafana-reporting*)

(ver editor en vivo)

Arranque

Operación vía docker/docker-compose

- Primer arranque: descarga de imagenes, creacion de containers

Arranque

Operación vía docker/docker-compose

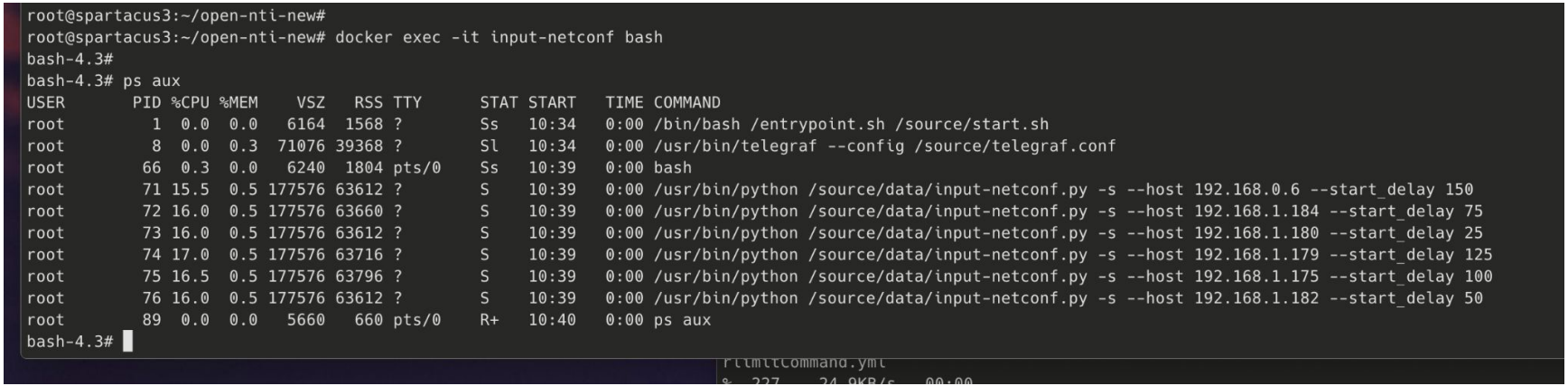
- Primer arranque: descarga de imagenes, creacion de containers
- Arranque parada controlada (destrucción de containers sin almacenamiento persistente)

```
root@spartacus3:~/open-nti-new# docker-compose -f docker-compose-telegraf.yml start
Starting influxdb                ... done
Starting input-internal          ... done
Starting input-netconf           ... done
Starting input-snmp              ... done
Starting input-netflow-telegraf  ... done
Starting jti_openconfig_telegraf ... done
Starting jti_udp_telegraf        ... done
Starting kapacitor               ... done
Starting chronograf              ... done
Starting grafana                 ... done
Starting renderer                ... done
root@spartacus3:~/open-nti-new# docker ps
CONTAINER ID   IMAGE                                        COMMAND                  CREATED      STATUS         PORTS                                                    NAMES
58509170c573   quay.io/influxdb/chronograf:1.5.0.1          "/usr/bin/chronograf…"   4 days ago   Up 28 seconds  0.0.0.0:8888->8888/tcp                                   chronograf
bb8f74f4ad1d   open-nti-new_input-internal                  "/entrypoint.sh /sou…"   4 days ago   Up 29 seconds  8092/udp, 8125/udp, 8094/tcp                             input-internal
e9fd6c83323a   open-nti-new_input-netflow-telegraf          "/entrypoint.sh /sou…"   4 days ago   Up 36 seconds  8092/udp, 0.0.0.0:2055->2055/udp, 8125/udp, 8094/tcp     input-netflow-telegraf
0b6a4b2d2875   open-nti-new_input-netconf                   "/entrypoint.sh /sou…"   4 days ago   Up 32 seconds  8092/udp, 8125/udp, 8094/tcp                             input-netconf
b9aa9120b43c   open-nti-new_jti_udp_telegraf                "/entrypoint.sh /sou…"   4 days ago   Up 30 seconds  8092/udp, 8125/udp, 8094/tcp, 0.0.0.0:50000->50000/udp   jti_udpNative_telegraf
8cd17bb48434   open-nti-new_jti_openconfig_telegraf         "/entrypoint.sh /sou…"   4 days ago   Up 34 seconds  8092/udp, 8125/udp, 8094/tcp, 0.0.0.0:50051->50051/tcp   jti_openConfig_telegraf
bc58091b19b7   open-nti-new_input-snmp                      "/entrypoint.sh /sou…"   4 days ago   Up 33 seconds  8092/udp, 0.0.0.0:162->162/udp, 8125/udp, 8094/tcp       input-snmp
e7d3ec011bc2   kapacitor:1.5.0                              "/entrypoint.sh kapa…"   4 days ago   Up 35 seconds  0.0.0.0:9092->9092/tcp                                   kapacitor
81ec61fdcbd4   grafana/grafana                              "/run.sh"                4 days ago   Up 46 seconds  0.0.0.0:3000->3000/tcp                                   grafana
2f476d89291f   grafana/grafana-image-renderer:2.0.0-beta1   "dumb-init -- node b…"   4 days ago   Up 46 seconds  0.0.0.0:8081->8081/tcp                                   open-nti-new_renderer_1
a67a526e0b0e   influxdb:1.6.4                               "/entrypoint.sh infl…"   4 days ago   Up 45 seconds  0.0.0.0:8086->8086/tcp, 0.0.0.0:8090->8090/tcp           influxdb
root@spartacus3:~/open-nti-new#
```

Arranque

Operación vía docker/docker-compose

- Primer arranque: descarga de imagenes, creacion de containers
- Arranque parada controlada (destrucción de containers sin almacenamiento persistente)
- Acceso a logs, entrada en containers para resolución de problemas, etc.



Arranque

Operación vía docker/docker-compose

- Primer arranque: descarga de imagenes, creacion de containers
- Arranque parada controlada (destrucción de containers sin almacenamiento persistente)
- Acceso a logs, entrada en containers para resolución de problemas, etc.

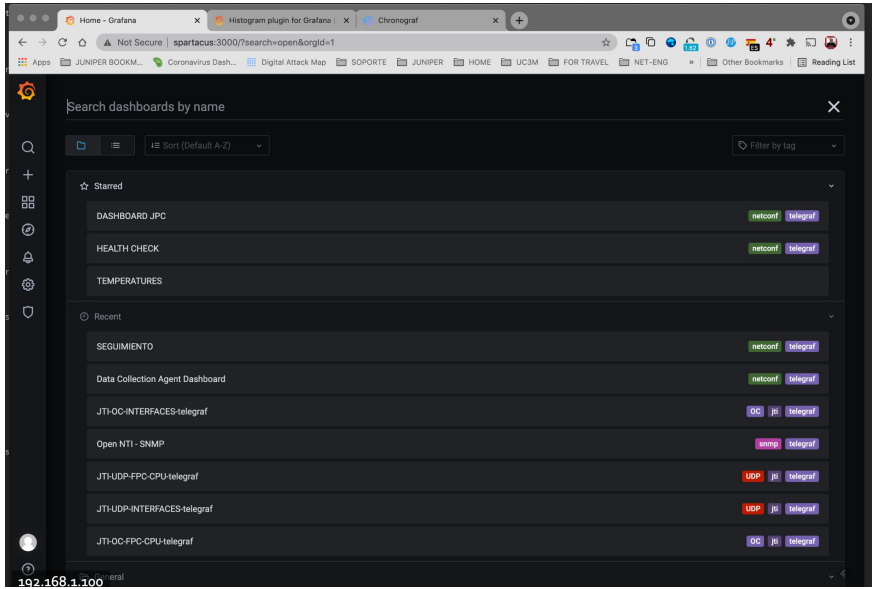
```
Starting renderer                    ... done
root@spartacus3:~/open-nti-new# docker logs -n 100 -f input-netconf
2021-05-25T10:44:11Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:44:30Z D! [outputs.influxdb] wrote batch of 1 metrics in 30.906874ms
2021-05-25T10:44:30Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:44:41Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:45:00Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:45:11Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:45:30Z D! [outputs.influxdb] wrote batch of 137 metrics in 637.410407ms
2021-05-25T10:45:30Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:45:55Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:46:00Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:46:25Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:46:30Z D! [outputs.influxdb] wrote batch of 2 metrics in 46.029292ms
2021-05-25T10:46:30Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:46:55Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:47:00Z D! [outputs.influxdb] wrote batch of 1 metrics in 468.048767ms
2021-05-25T10:47:00Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:47:25Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:47:30Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:47:55Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:48:00Z D! [outputs.influxdb] wrote batch of 1 metrics in 76.524938ms
2021-05-25T10:48:00Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:48:30Z D! [outputs.influxdb] wrote batch of 136 metrics in 237.398402ms
2021-05-25T10:48:30Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:48:39Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:49:00Z D! [outputs.influxdb] wrote batch of 1 metrics in 41.693417ms
2021-05-25T10:49:00Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:49:09Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:49:30Z D! [outputs.influxdb] wrote batch of 1 metrics in 31.074736ms
2021-05-25T10:49:30Z D! [outputs.influxdb] buffer fullness: 0 / 100000 metrics.
2021-05-25T10:49:39Z W! [agent] input "inputs.exec" did not complete within its interval
2021-05-25T10:50:00Z D! [outputs.influxdb] wrote batch of 1 metrics in 232.880989ms
```

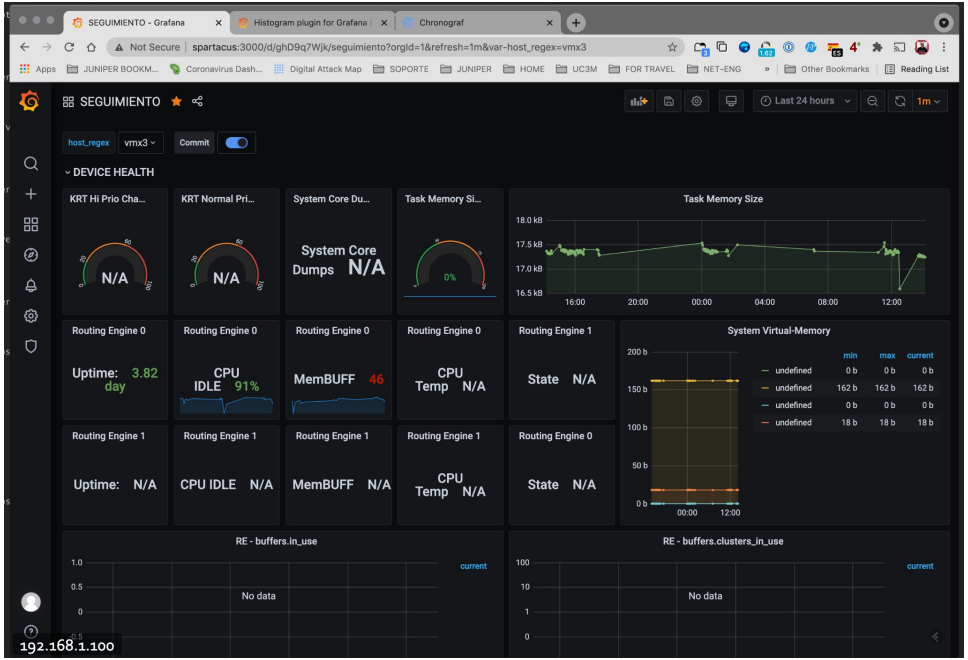
```
h /source/start.sh
fig /source/telegraf.conf

/data/input-netconf.py -s --host 192.168.0.6 --start_delay 150
/data/input-netconf.py -s --host 192.168.1.184 --start_delay 75
/data/input-netconf.py -s --host 192.168.1.180 --start_delay 25
/data/input-netconf.py -s --host 192.168.1.179 --start_delay 125
/data/input-netconf.py -s --host 192.168.1.175 --start_delay 100
/data/input-netconf.py -s --host 192.168.1.182 --start_delay 50
```

ARRANQUE

Interfaz usuario

GRAFANA

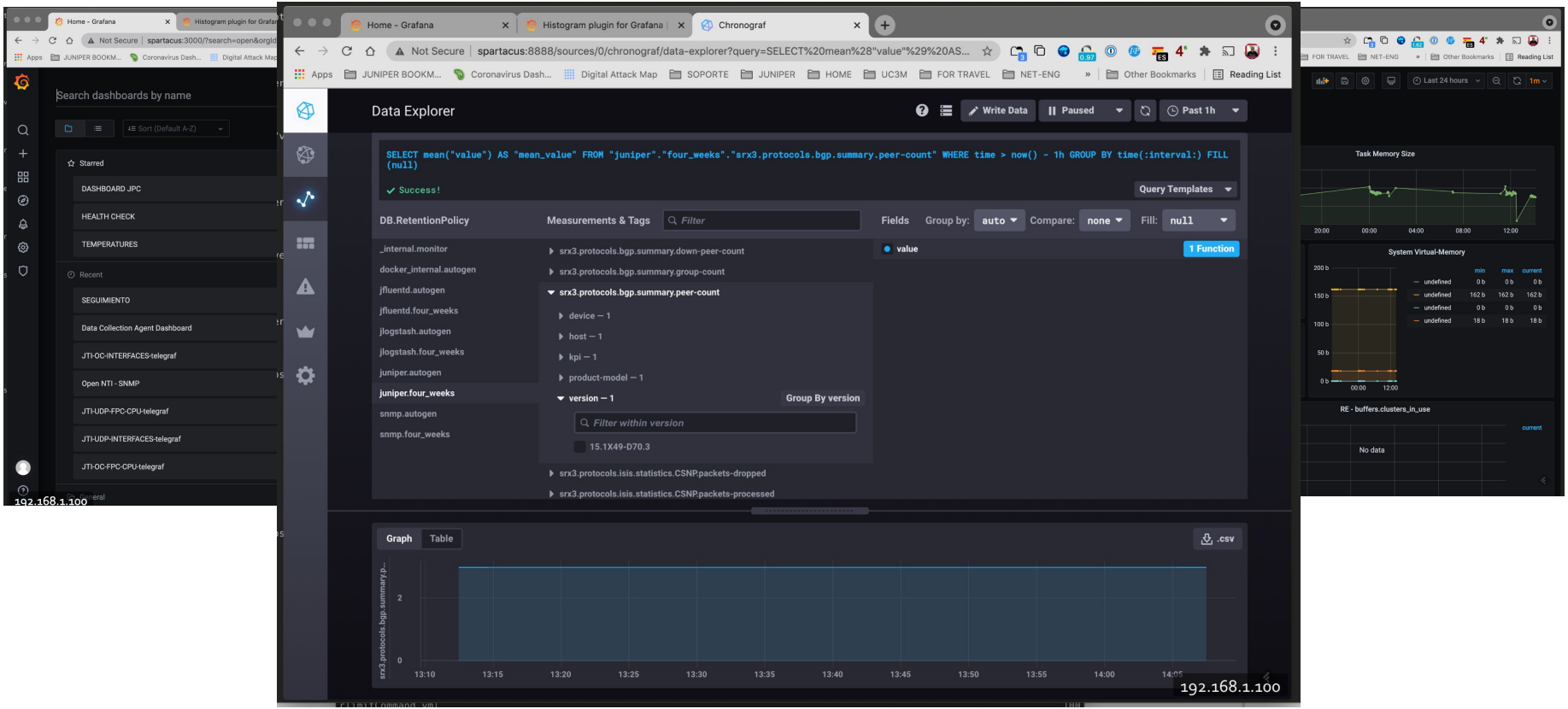


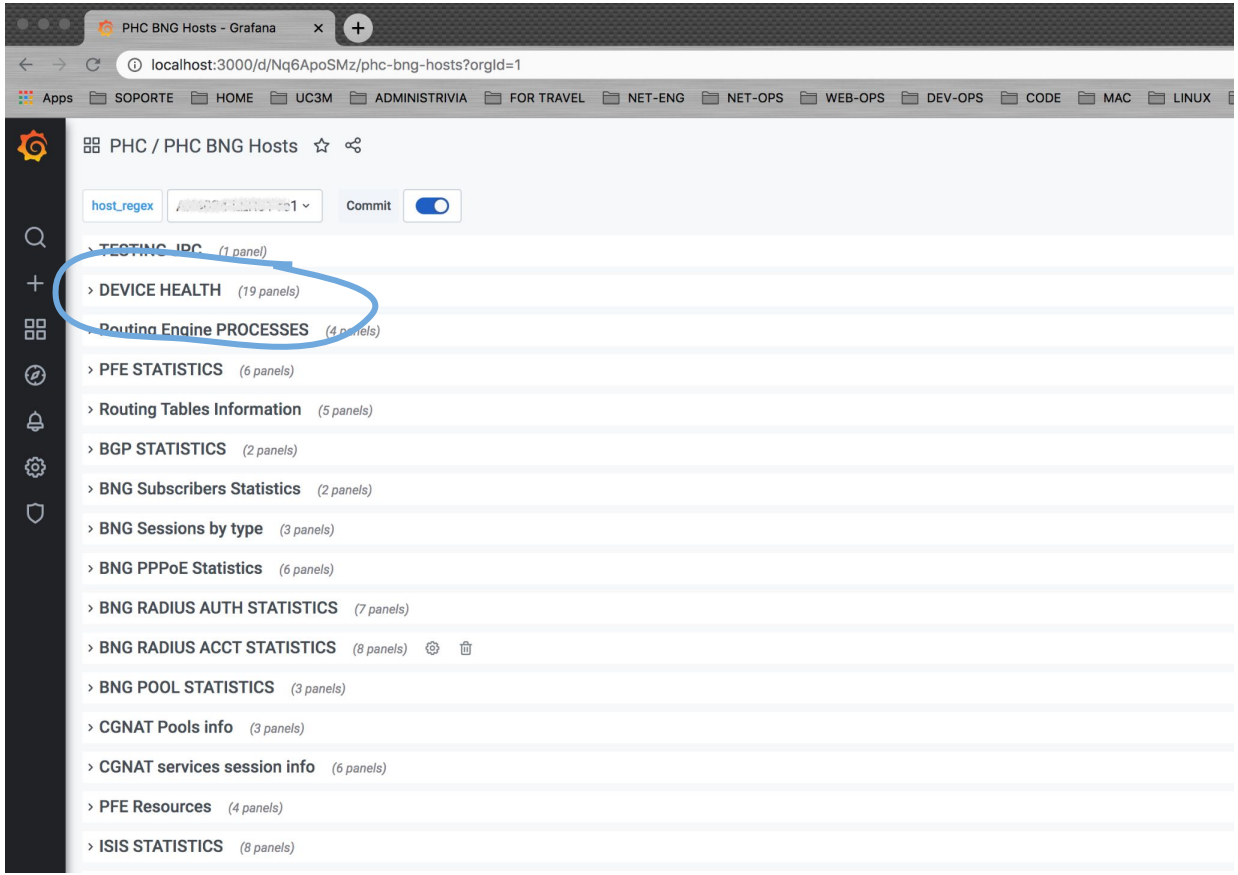


ARRANQUE

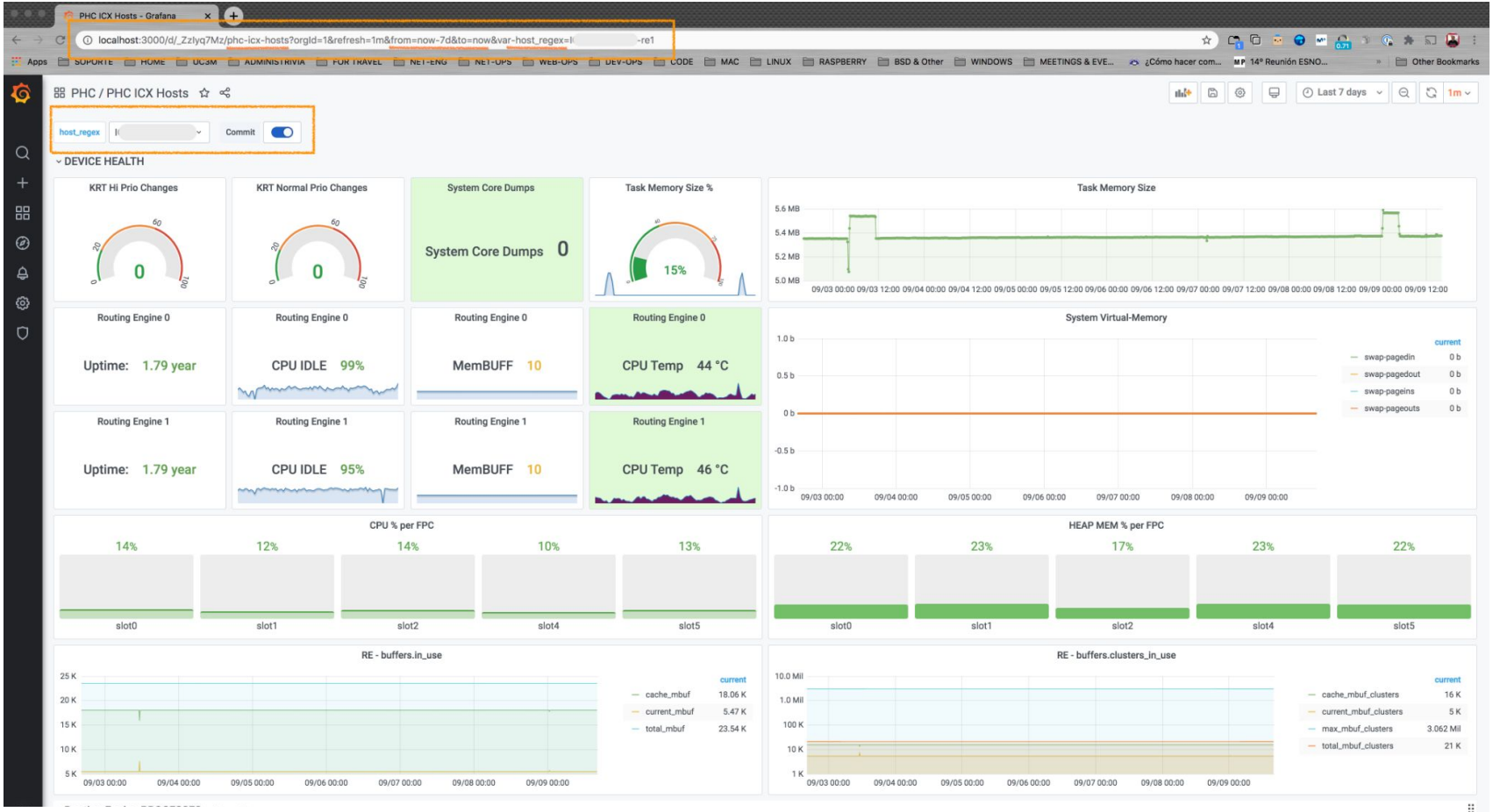
Interfaz usuario

GRAFANA y Chronograf (:8888)

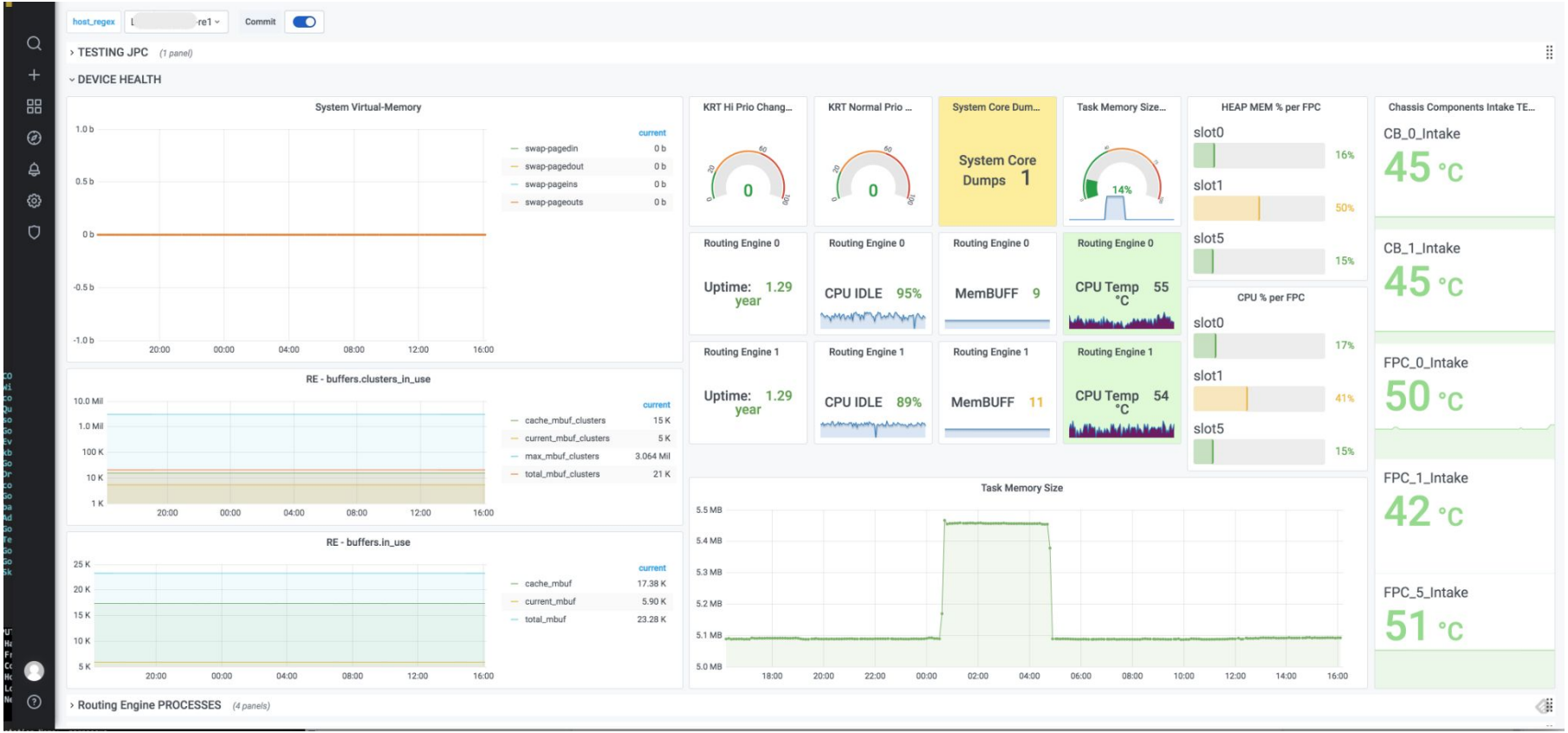




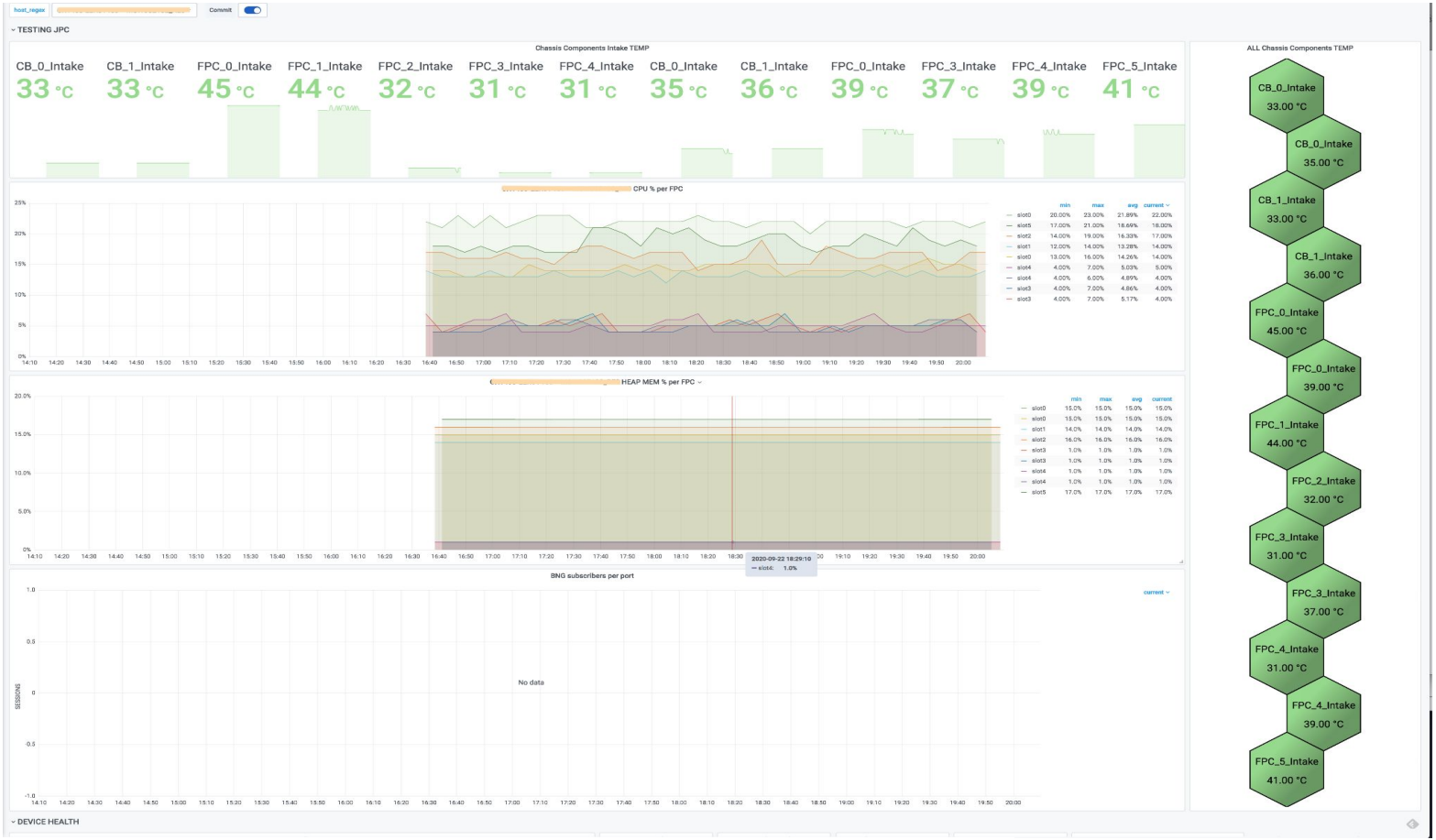
Experimento I - HealthCheck



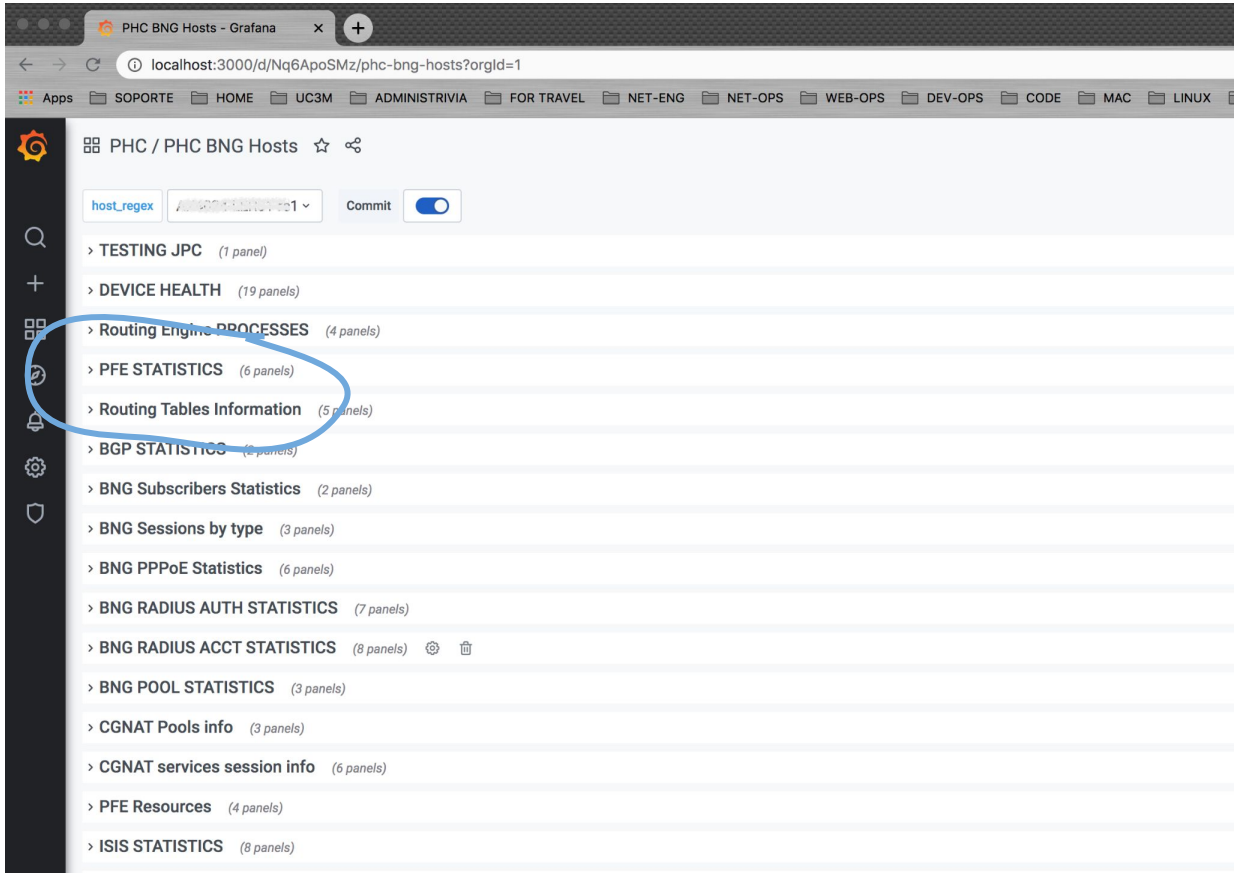
Experimento I - HealthCheck



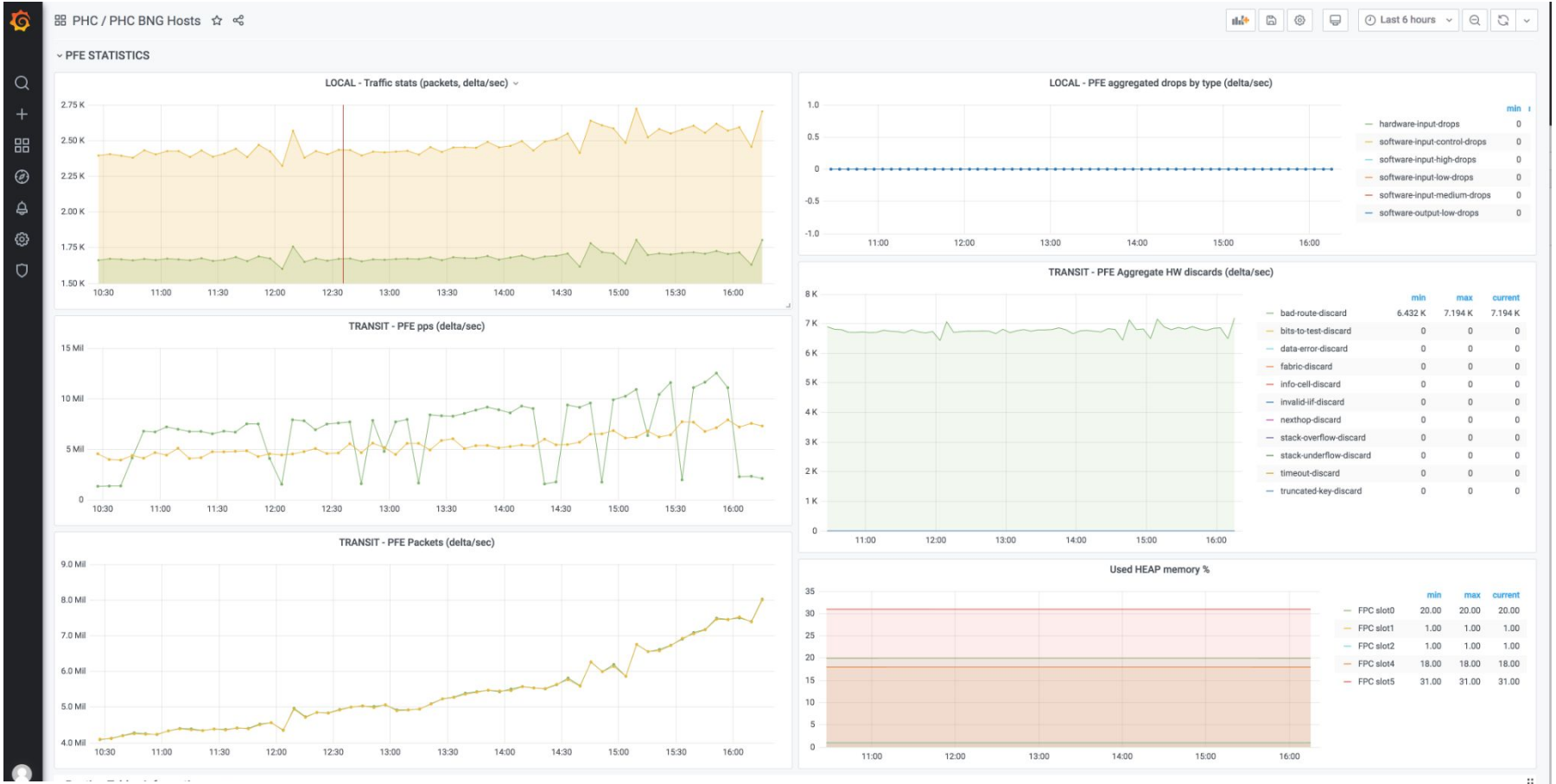
Experimento 1 - HealthCheck



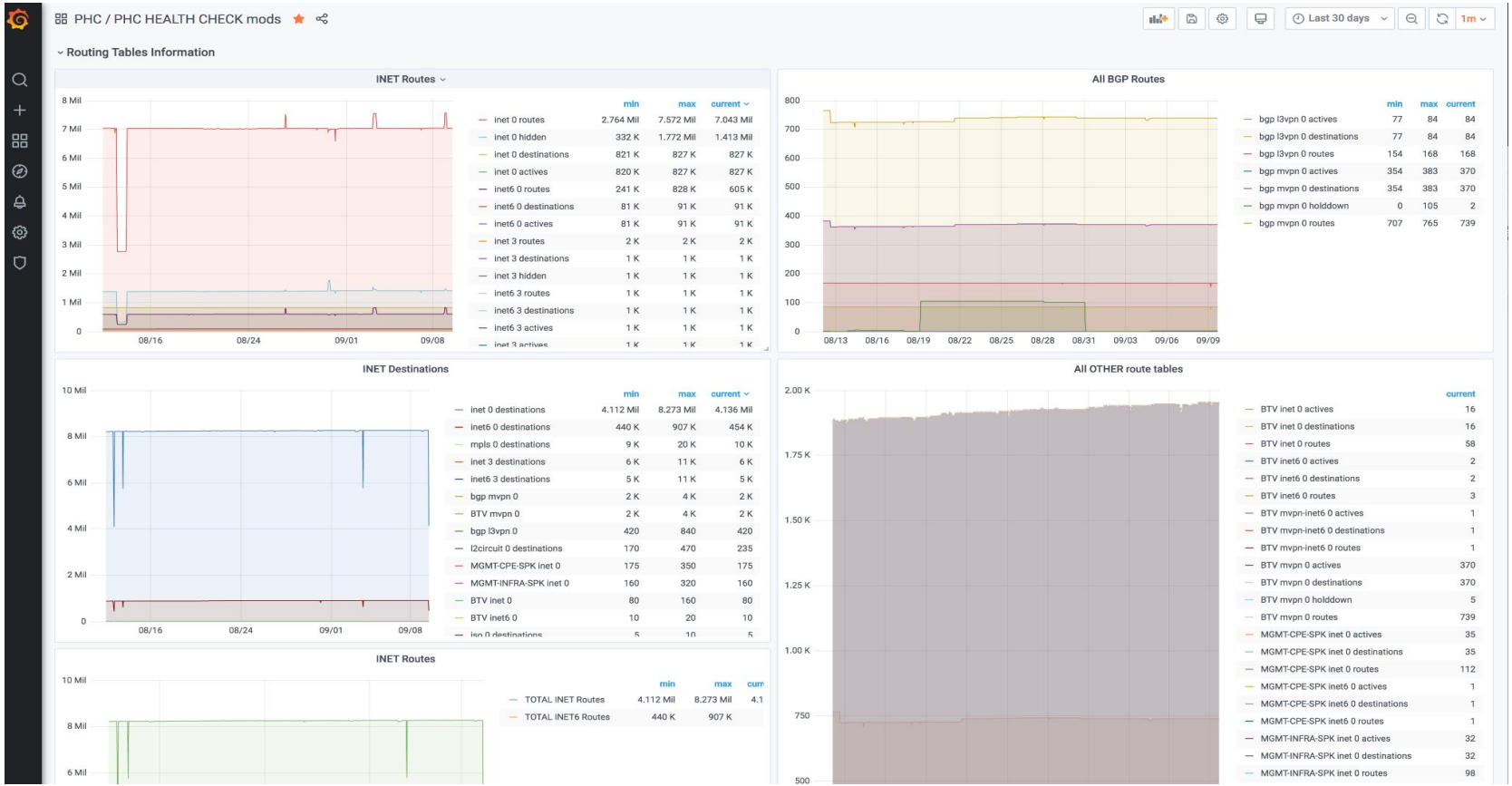
Experimento I - HealthCheck



Experimento II - DATA PLANE & Routing

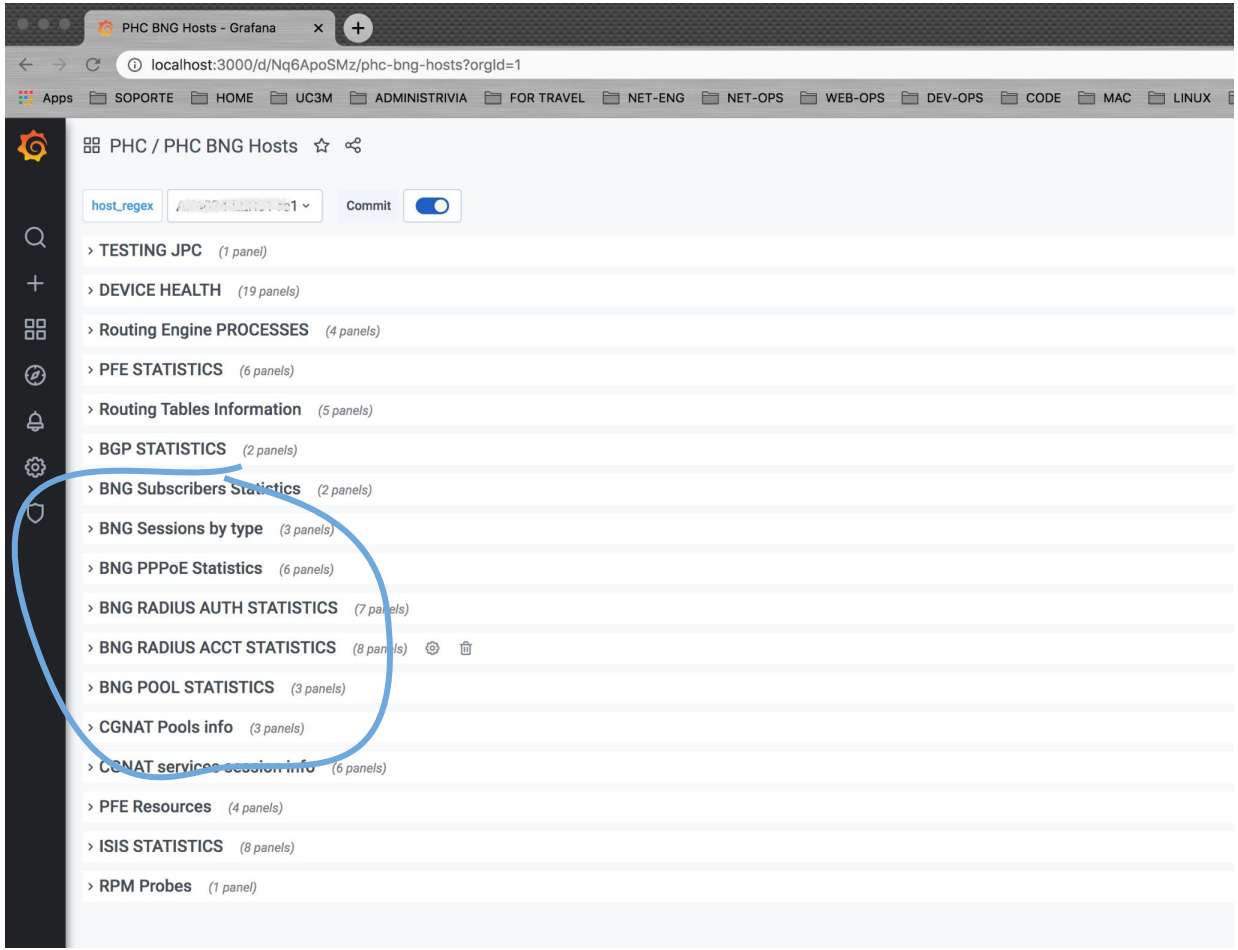


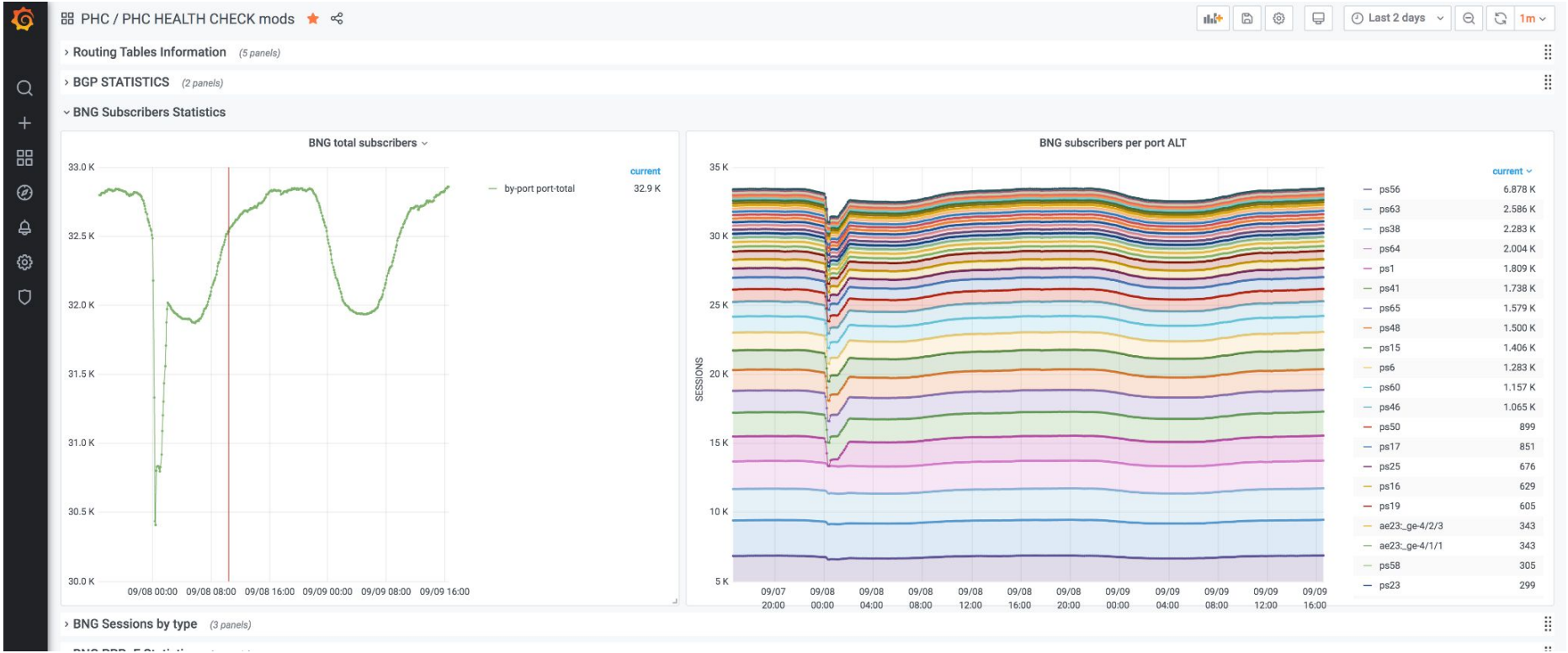
Experimento II - DATA PLANE



Experimento II - Routing

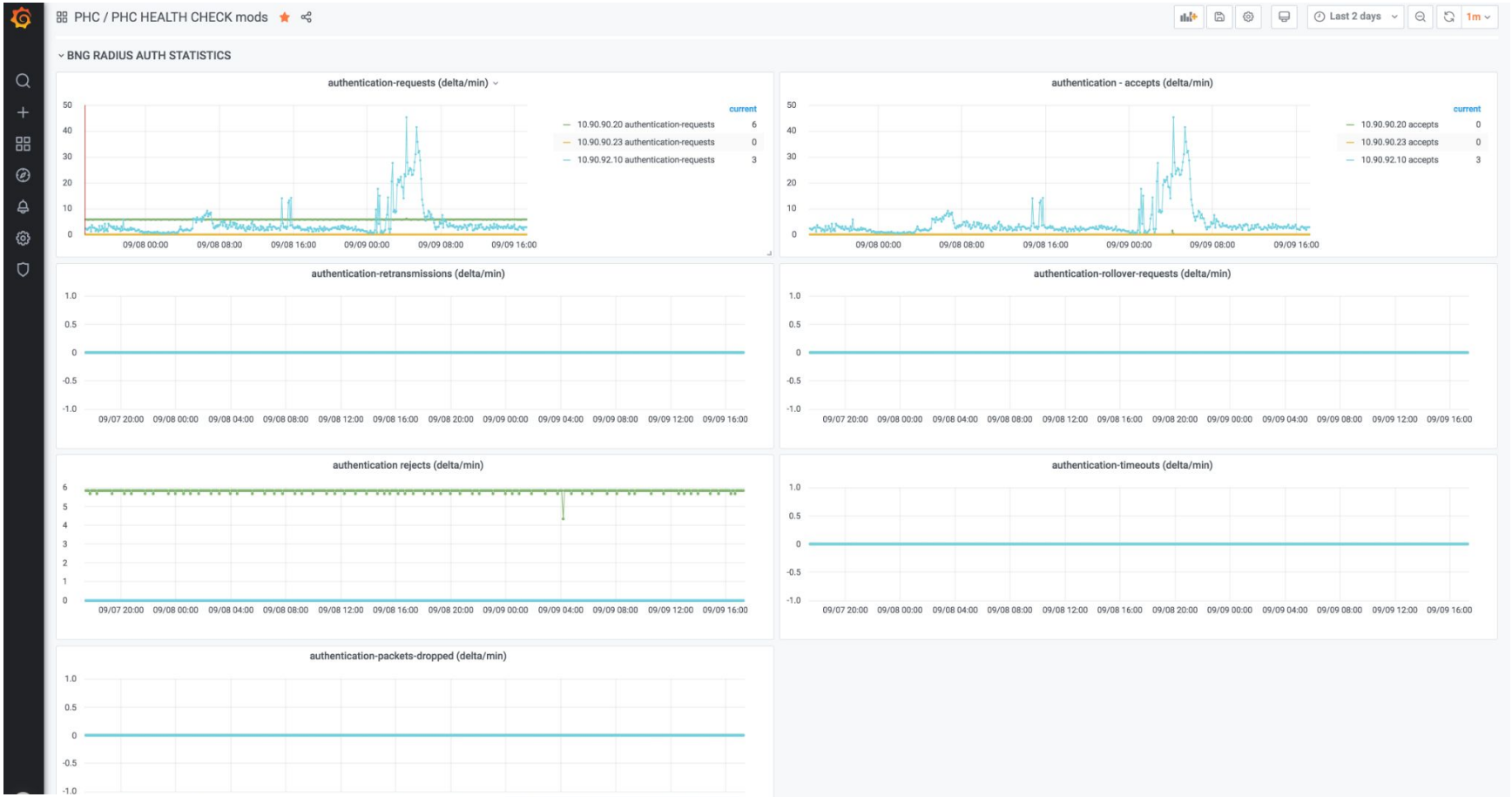
Experimento III - BNG & CGNat





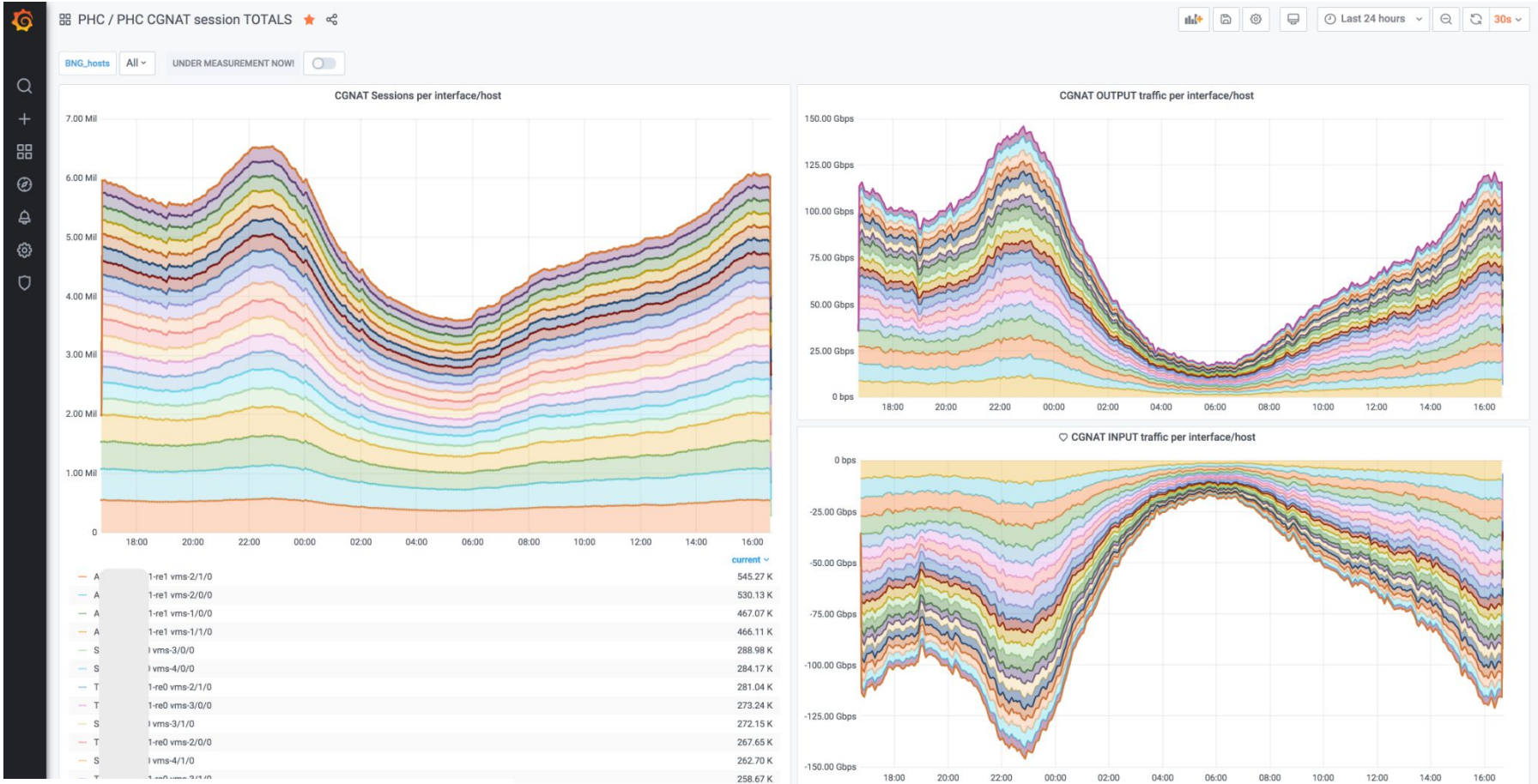
Experimento III - BNG & CGNat

Experimento III - BNG & CGNat



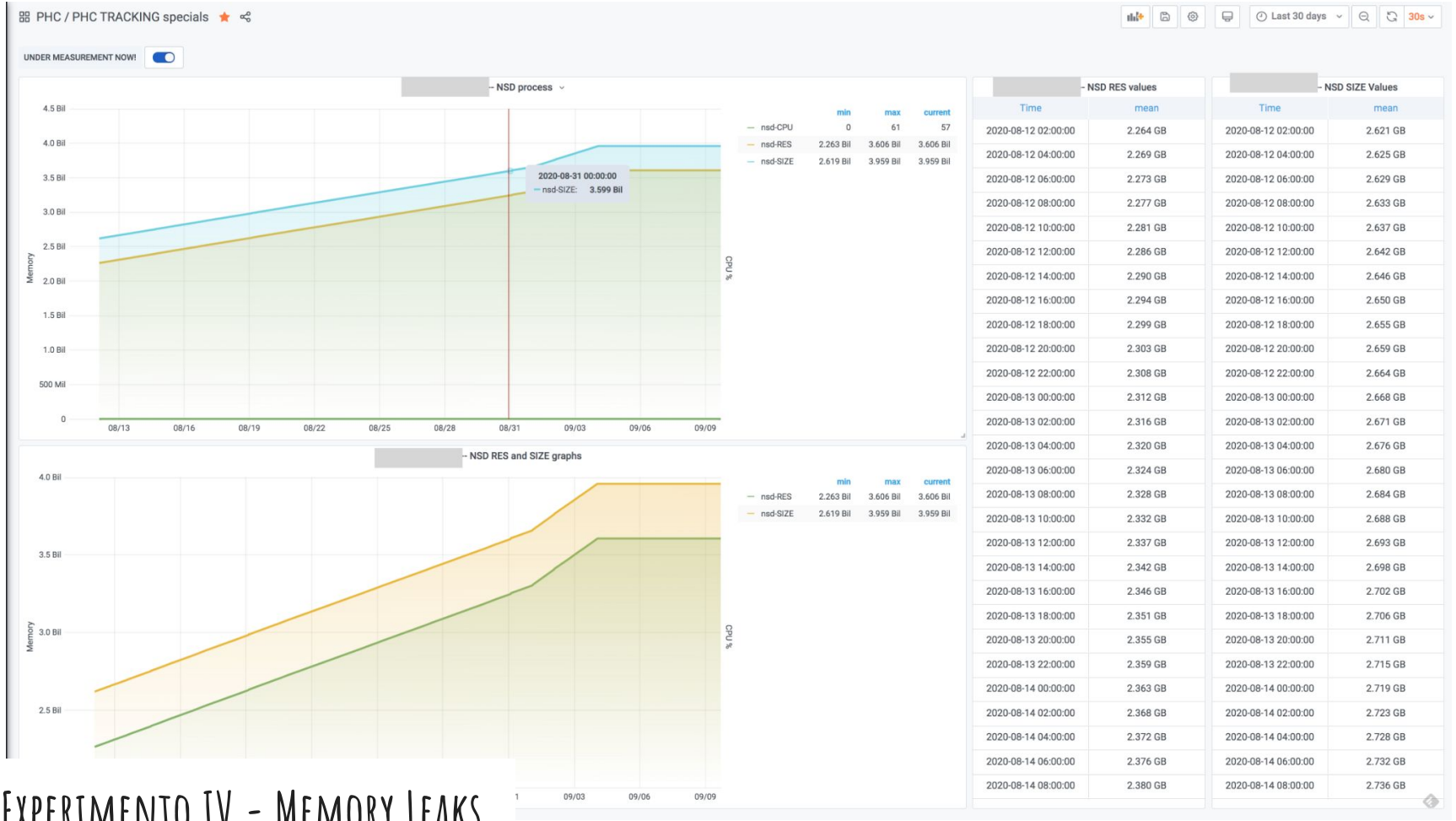
Experimento III - BNG & CGNat

Experimento III - BNG & CGNat

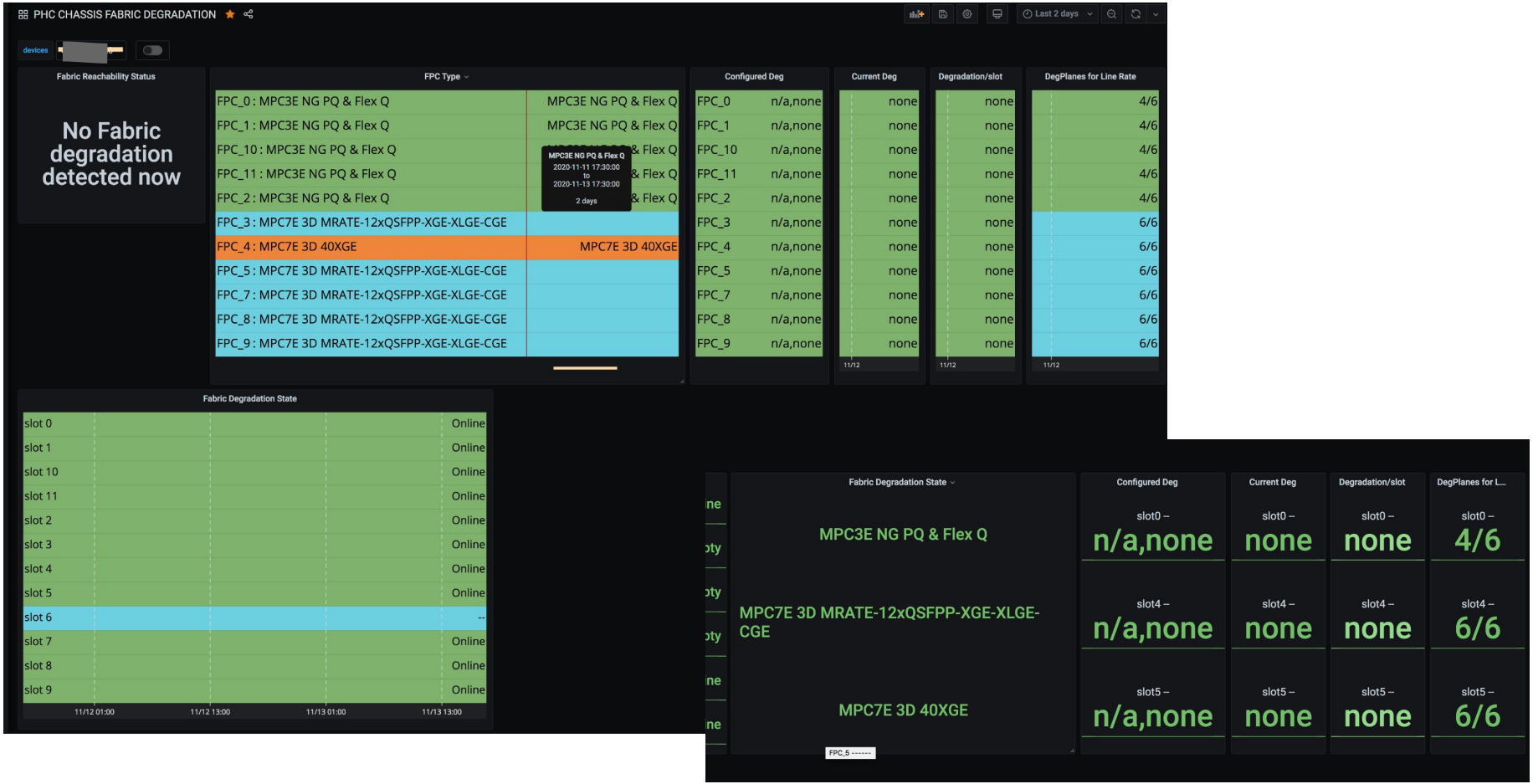


Experimento III - BNG & CGNat

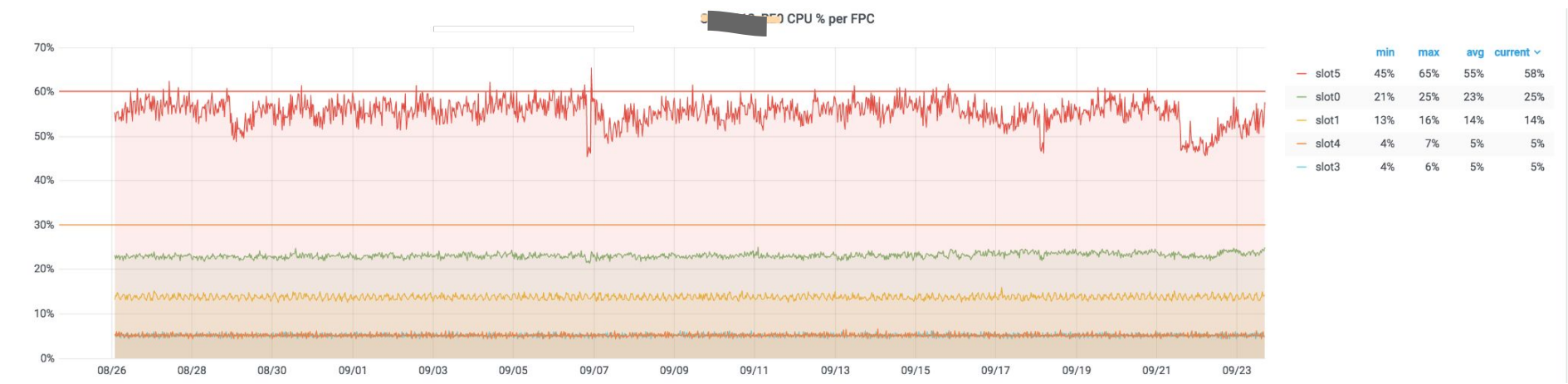
Experimento IV - Memory Leaks

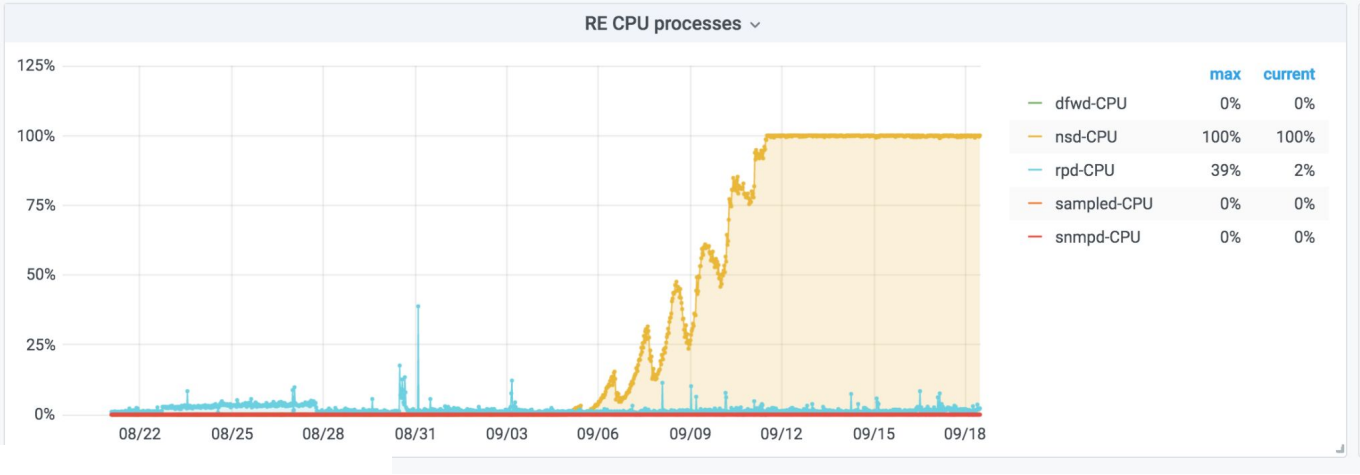


Experimento IV - Memory Leaks



Experimento V - MONITORIZACION EXTREMA





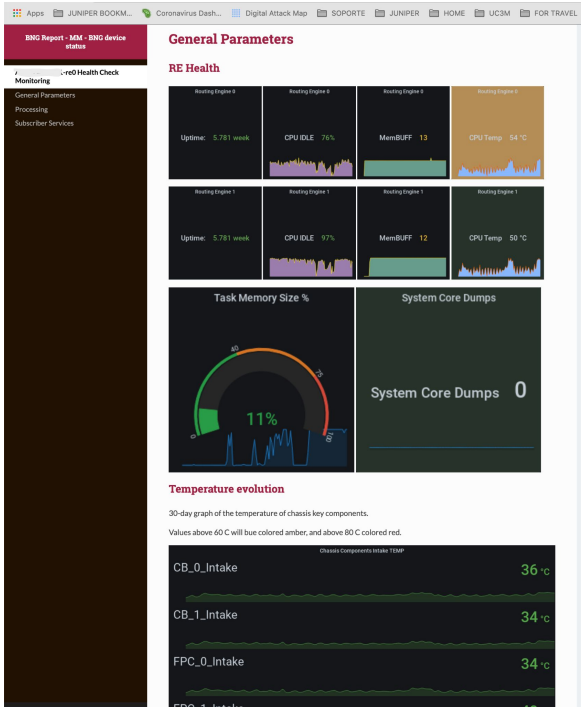
Experimento V - MONITORIZACION EXTREMA

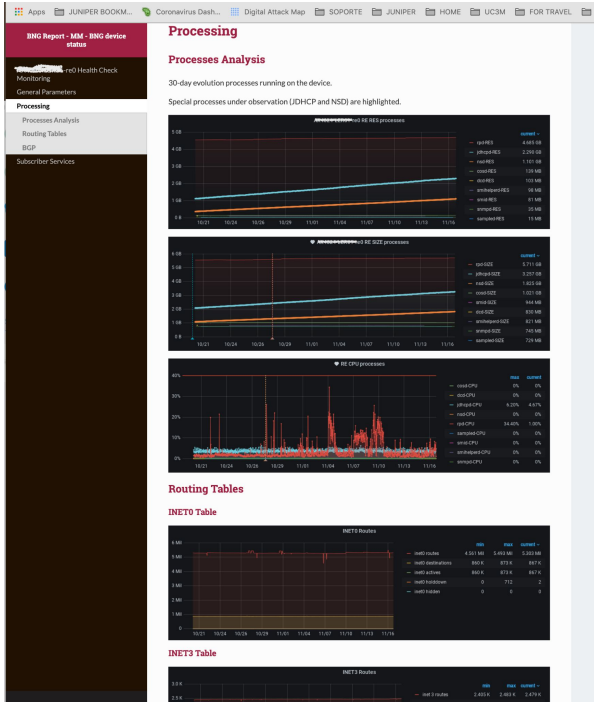
Para una segunda parte (en preparación, ahora mismo)

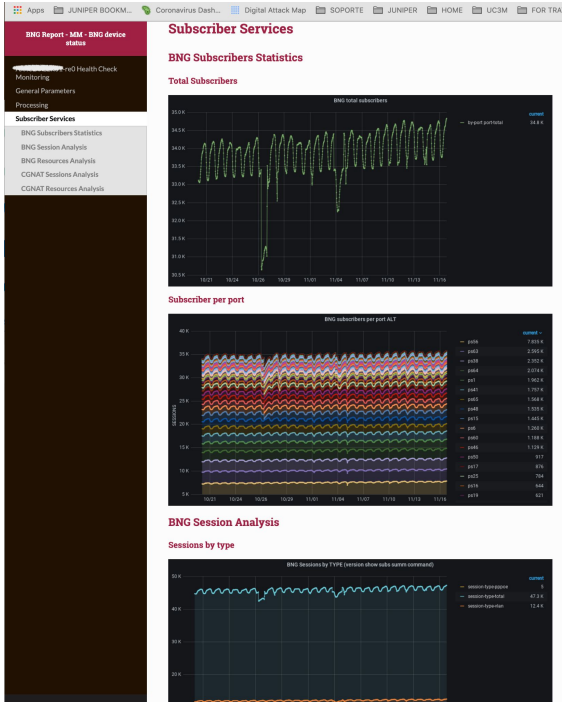
Integra gráficas de grafana, R y consultas a la db de Influxdb.

Experimento VI - Reporting

Ejemplo







Experimento VI - Reporting

Conclusión

open-nti-new puede ser una herramienta útil para:

- comprobaciones y chequeos de equipos específicos, o subnets/funcionalidades
- automatización de tareas de Health-check en redes
- ensayos de visualización de mediciones de red

necesita dedicación para ser operativa

necesita añadidos (para hacer reporting, por ejemplo)

GRACIAS!!