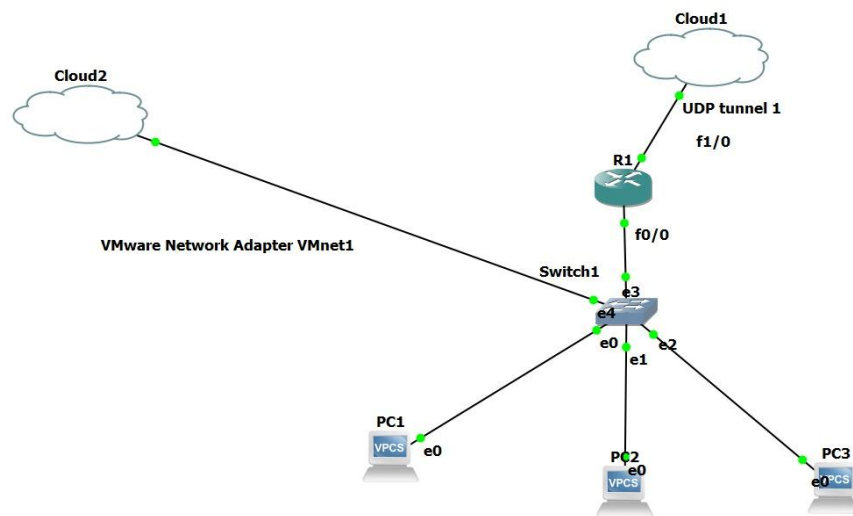


# Département Monitoring

- Topologie :

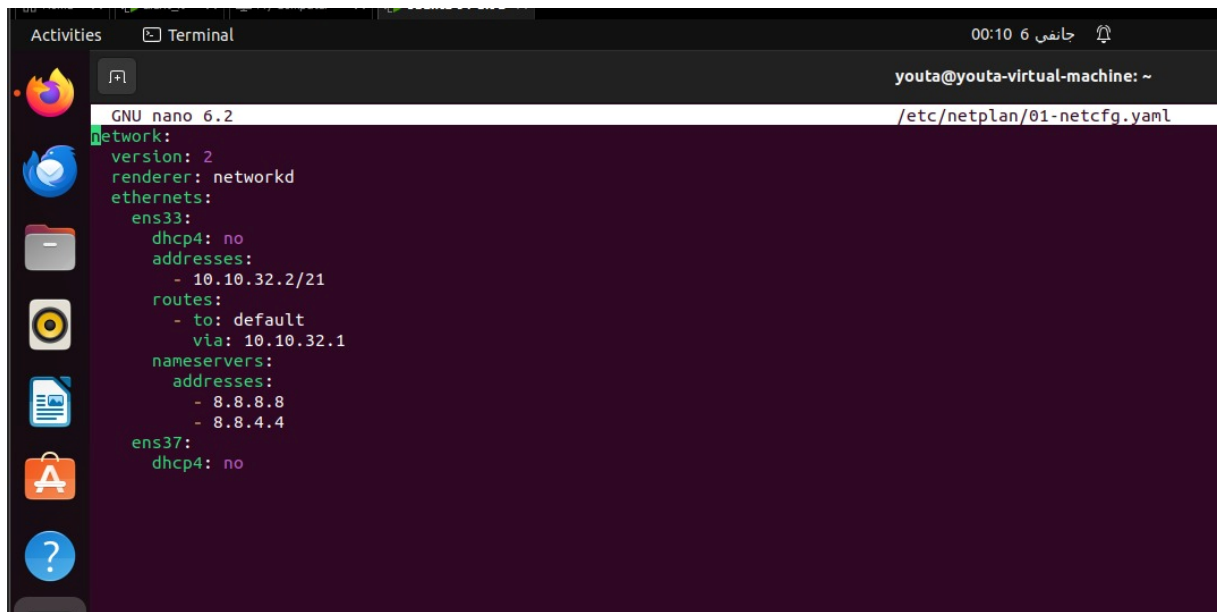


- Configuration Routeur RZ-2 - DHCP, IPsec et Services de Base :

```
Overview R1
crypto ipsec transform-set TS-GRE esp-aes esp-sha-hmac
mode transport
!
!
!
interface Tunnel0
no ip address
!
interface Tunnel1
ip address 172.16.2.2 255.255.255.252
tunnel source FastEthernet0/1
tunnel destination 200.0.0.13
!
interface FastEthernet0/0
ip address 10.10.32.1 255.255.248.0
ip helper-address 10.10.32.1
ip nat inside
ip virtual-reassembly
duplex auto
speed auto
!
interface FastEthernet0/1
ip address 200.0.0.14 255.255.255.252
ip nat outside
ip virtual-reassembly
duplex auto
speed auto
!
router ospf 1
router-id 22.22.22.22
log-adjacency-changes
network 10.10.32.0 0.0.7.255 area 0
network 172.16.2.0 0.0.0.3 area 0
network 192.168.100.0 0.0.0.255 area 0
network 200.200.200.0 0.0.0.3 area 0
!
ip forward-protocol nd
ip route 201.201.201.0 255.255.255.0 Tunnel1
!
!
no ip http server
no ip http secure-server
ip nat inside source list 10 interface FastEthernet0/1 overload
ip nat inside source list 100 interface FastEthernet0/1 overload
!
ip access-list extended ACL-VPN-BACKBONE
permit gre host 200.0.0.14 host 200.0.0.13
!
access-list 10 permit 10.10.32.0 0.0.7.255
access-list 100 permit ip 10.10.32.0 0.0.7.255 any
access-list 100 deny ip any 172.16.0.0 0.0.0.3
access-list 100 deny ip any 201.201.201.0 0.0.0.3
no cdp log mismatch duplex
!
!
!
control-plane
!
!
!
!
```



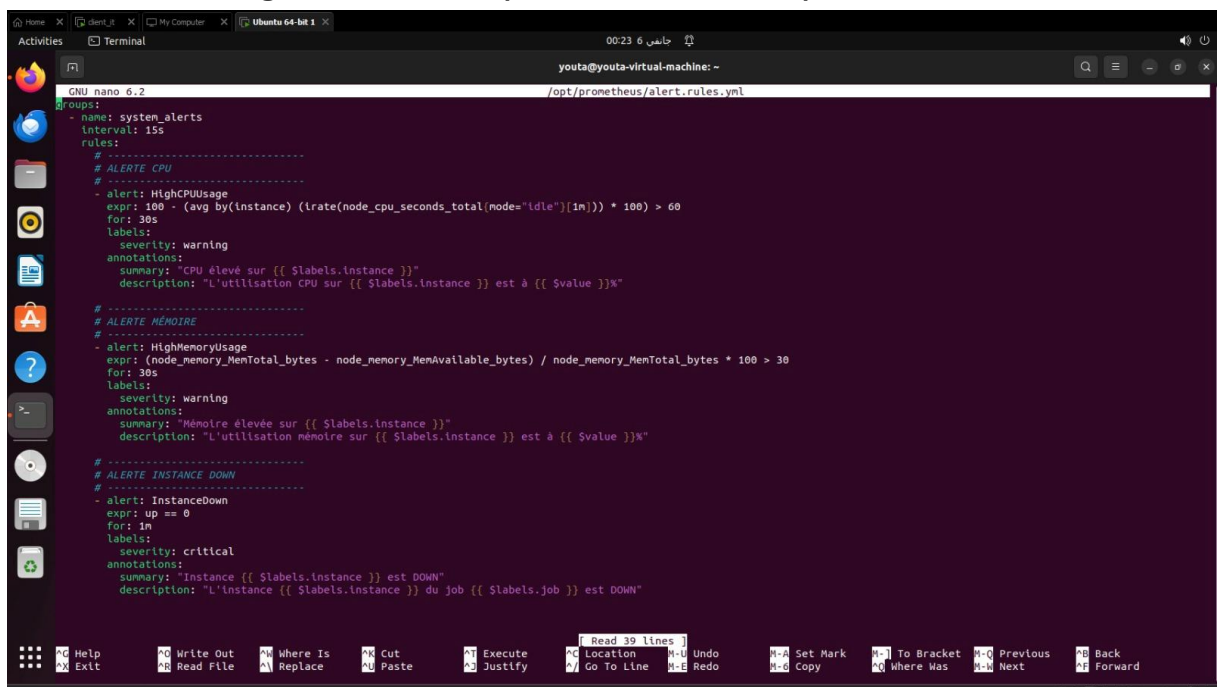
- Configuration Réseau Netplan



The screenshot shows a terminal window with the GNU nano 6.2 editor open. The file being edited is `/etc/netplan/01-netcfg.yaml`. The configuration is for a network interface `ens33` with the following settings:

```
network:
  version: 2
  renderer: networkd
  ethernets:
    ens33:
      dhcp4: no
      addresses:
        - 10.10.32.2/21
      routes:
        - to: default
          via: 10.10.32.1
      nameservers:
        addresses:
          - 8.8.8.8
          - 8.8.4.4
    ens37:
      dhcp4: no
```

- Vérification Configuration Réseau (Interfaces et Routes)



The screenshot shows a terminal window with the GNU nano 6.2 editor open. The file being edited is `/opt/prometheus/alert.rules.yml`. The configuration is for a group of alerts named `system_alerts` with an interval of 15s. The rules are:

```
groups:
- name: system_alerts
  interval: 15s
  rules:
    # -----
    # ALERTE CPU
    # -----
    - alert: HighCPUUsage
      expr: 100 - (avg by(instance) (rate(node_cpu_seconds_total{node="idle"}[1m])) * 100) > 60
      for: 30s
      labels:
        severity: warning
      annotations:
        summary: "CPU élevé sur {{ $labels.instance }}"
        description: "L'utilisation CPU sur {{ $labels.instance }} est à {{ $value }}%"
    # -----
    # ALERTE MEMOIRE
    # -----
    - alert: HighMemoryUsage
      expr: (node_memory_MemTotal_bytes - node_memory_MemAvailable_bytes) / node_memory_MemTotal_bytes * 100 > 30
      for: 30s
      labels:
        severity: warning
      annotations:
        summary: "Mémoire élevée sur {{ $labels.instance }}"
        description: "L'utilisation mémoire sur {{ $labels.instance }} est à {{ $value }}%"
    # -----
    # ALERTE INSTANCE DOWN
    # -----
    - alert: InstanceDown
      expr: up == 0
      for: 1m
      labels:
        severity: critical
      annotations:
        summary: "Instance {{ $labels.instance }} est DOWN"
        description: "L'instance {{ $labels.instance }} du job {{ $labels.job }} est DOWN"
```

- Accès Base de Données MySQL



```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| entreprise_bd |
| information_schema |
| performance_schema |
+-----+
3 rows in set (0.67 sec)

mysql> USE entreprise_bd;

Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql>
mysql> SELECT * FROM employees;
+-----+-----+-----+-----+-----+-----+
| id | nom | prenom | poste | departement | salaire |
+-----+-----+-----+-----+-----+-----+
| 3 | sfer | Mohamed | Analyste Données | Base de Données | 48000.00 |
| 4 | Mansouri | Leila | Data Scientist | Base de Données | 62000.00 |
| 6 | jasser | balti | BACKBOUNA | devops | 50000.00 |
| 7 | Mohamed | Maram | NULL | NULL | 2500.00 |
| 8 | maram | samar | NULL | NULL | 65000.00 |
| 9 | dhrioua | Youta | RH | IT | 70000.00 |
| 10 | Ner | BentKaroui | hamas_lhoma | tetalfa7 | 10000.20 |
| 11 | mlh | ysn | consierge | NFS | 2500.00 |
+-----+-----+-----+-----+-----+-----+
8 rows in set (1.10 sec)

mysql>
```

- Configuration des Alertes de Supervision

The screenshot shows a terminal window with the nano text editor open, editing the file `/opt/prometheus/alert.rules.yml`. The configuration defines a group of alerts named `system_alerts` with an interval of 15s. It includes three alerts: `HighCPUUsage`, `HighMemoryUsage`, and `InstanceDown`. Each alert has a specific expression, duration, severity, and annotations.

```

groups:
- name: system_alerts
  interval: 15s
  rules:
    # -----
    # ALERTE CPU
    # -----
    - alert: HighCPUUsage
      expr: 100 - (avg by(instance) (irate(node_cpu_seconds_total{node="idle"}[1m])) * 100) > 60
      for: 30s
      labels:
        severity: warning
      annotations:
        summary: "CPU élevé sur {{ $labels.instance }}"
        description: "L'utilisation CPU sur {{ $labels.instance }} est à {{ $value }}%"

    # -----
    # ALERTE MEMOIRE
    # -----
    - alert: HighMemoryUsage
      expr: (node_memory_MemTotal_bytes - node_memory_MemAvailable_bytes) / node_memory_MemTotal_bytes * 100 > 30
      for: 30s
      labels:
        severity: warning
      annotations:
        summary: "Mémoire élevée sur {{ $labels.instance }}"
        description: "L'utilisation mémoire sur {{ $labels.instance }} est à {{ $value }}%"

    # -----
    # ALERTE INSTANCE DOWN
    # -----
    - alert: InstanceDown
      expr: up == 0
      for: 1m
      labels:
        severity: critical
      annotations:
        summary: "Instance {{ $labels.instance }} est DOWN"
        description: "L'instance {{ $labels.instance }} du job {{ $labels.job }} est DOWN"
  
```

The screenshot shows a code editor with three tabs: `commands.txt`, `grafana-server.service`, and `node_expo`. The `grafana-server.service` tab is active, displaying the systemd service configuration for Prometheus.

```

1 [Unit]
2 Description=Prometheus
3 Wants=network-online.target
4 After=network-online.target
5
6 [Service]
7 User=nobody
8 Group=nogroup
9 Type=simple
10 ExecStart=/opt/prometheus/prometheus \
11   --config.file=/opt/prometheus/prometheus.yml \
12   --storage.tsdb.path=/opt/prometheus/data
13
14 [Install]
15 WantedBy=multi-user.target
  
```



```
1 #!/bin/bash
2 # =====
3 # NFS CLIENT SETUP SCRIPT
4 # =====
5
6 SERVER_IP="10.10.42.20"
7 NFS_EXPORT="/nfs_share"
8 MOUNT_POINT="/mnt/nfs"
9
10 GROUP="nfsgrp"
11 USER="server"
12 UID_NUM="1001"
13 IP_CLIENT="10.10.32.2"
14 INTERFACE="ens33"
15 GATEWAY="10.10.42.20"
16 NFS_SERVER="10.10.42.20"
17
18 echo "Creation groupe nfsgrp"
19 getent group $GROUP || groupadd $GROUP
20
21 echo "Creation usr"
22 id $USER &>/dev/null || useradd -u $UID_NUM -m -s /bin/bash $USER
23
24 echo "ajout usr au nfsgrp"
25 usermod -aG $GROUP $USER
26
27 echo "pt partage"
28 mkdir -p $MOUNT_POINT
29
30 echo "ajout partage permanent"
31 grep -q "$SERVER_IP:$NFS_EXPORT" /etc/fstab || \
32 echo "$SERVER_IP:$NFS_EXPORT $MOUNT_POINT nfs defaults 0 0" >> /etc/fstab
33
34 echo "montage du partage nfs"
35 mount -a
36
37 echo "verifier montage"
38 mount | grep nfs
39
40 echo "Configuration CL NFS terminée avec succès"
```

---

```

1 # my global config
2 global:
3   scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1
   minute.
4   evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.
5   # scrape_timeout is set to the global default (10s).
6
7 # Alertmanager configuration
8 alerting:
9   alertmanagers:
10    - static_configs:
11      - targets:
12        # - alertmanager:9093
13
14 # Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
15 rule_files:
16   # - "first_rules.yml"
17   # - "second_rules.yml"
18
19 # A scrape configuration containing exactly one endpoint to scrape:
20 # Here it's Prometheus itself.
21 scrape_configs:
22   # The job name is added as a label `job=<job_name>` to any timeseries scraped from this
   config.
23   - job_name: "prometheus"
24
25     # metrics_path defaults to '/metrics'
26     # scheme defaults to 'http'.
27
28     static_configs:
29       - targets: ["localhost:9090"]
30       - labels:
31         app: "prometheus"

```

```

yml ×    nfs.sh ×    network.txt ×    install_monitoring.sh ×    commands.txt
1 [Unit]
2 Description=Grafana Server
3 After=network.target
4
5 [Service]
6 Type=simple
7 User=grafana
8 Group=grafana
9 ExecStart=/usr/sbin/grafana-server --homepath=/usr/share/grafana
10
11 [Install]
12 WantedBy=multi-user.target

```

```
< install_monitoring.sh × commands.txt × grafana-server.service × node_exporter.service × >
1 [Unit]
2 Description=Node Exporter
3 Wants=network-online.target
4 After=network-online.target
5
6 [Service]
7 User=prometheus
8 Group=prometheus
9 Type=simple
10 ExecStart=/usr/local/bin/node_exporter
11
12 [Install]
13 WantedBy=multi-user.target
```