



**RAPPORT R5.02**

**Université de la réunion / IUT**

**Département Réseaux, Télécommunication en  
Cybersécurité – 3ème années**

**R5.02  
SUPERVISION DES RÉSEAUX**

**EMMANUEL GRONDIN**

**TP1**

## Variables du dashboard :

### Variable \$host :

La variable `$host` permet de sélectionner dynamiquement la machine (hôte Docker) sur laquelle les métriques sont collectées.

- Elle filtre les métriques par instance, ce qui permet d'observer l'utilisation CPU, RAM, réseau et disque **spécifiquement pour un host choisi**.
- Grâce à ce mécanisme, il est possible de comparer plusieurs hôtes ou de cibler uniquement l'infrastructure TAAF.

The image shows two panels from the Grafana configuration interface. The left panel is titled 'host' and shows the 'General' section with the variable name 'host' and the label 'instance'. The right panel is titled 'Query options' and shows the 'Data source' set to 'prometheus', 'Query type' set to 'Label values', 'Label \*' set to 'instance', and 'Metric' set to 'container\_last\_seen'. The 'Label filters' section shows 'Select label' and 'Select value' options. The 'Regex' section is also visible.

### Variable \$conteneur :

La variable `$conteneur` sert à filtrer les métriques en fonction d'un conteneur particulier.

- Elle est utilisée dans les requêtes PromQL avec le label `name=~"$conteneur"`.
- Elle permet de sélectionner rapidement un conteneur parmi la liste (postgres-af, prometheus-taaf, grafana-taaf, etc.) et d'afficher uniquement ses ressources ou son trafic.
- Cela rend le dashboard **flexible et interactif**, car on peut naviguer d'un conteneur à l'autre sans changer les requêtes.

### conteneur

Variable type

Query

#### General

Name

The name of the template variable. (Max. 50 characters)

conteneur

Label

Optional display name

conteneur

Sort

How to sort the values of this variable

Disabled

Refresh

When to update the values of this variable

On dashboard load On time range change

#### Selection options

☒ Multi-value

Enables multiple values to be selected at the same time

☒ Allow custom values

Enables users to add custom values to the list

☒ Include All option

Enables an option to include all values

### Query options

Data source

prometheus

Query type

Label values

Label \*

name

Metric

container\_last\_seen

Label filters

Instance

Regex

Optional, if you want to extract part of a series name or metric node segment. Named capture groups can be used to separate the display text and value (see examples of).

## Réponses attendues :

### Question 1 – Variables de base

Combien d'hosts différents sont disponibles dans \$host ?

### Preview of values

All

cadvisor:8080

Il y'a en tout 1 hosts dans la variable \$host.

### Question 2 – Métriques simples

Combien de conteneurs sont actuellement en cours d'exécution au total ?

PromQL :

```
count(container_last_seen{name!=""})*
```

Dashboard final :



Réponse : 14 Conteneur actif

---

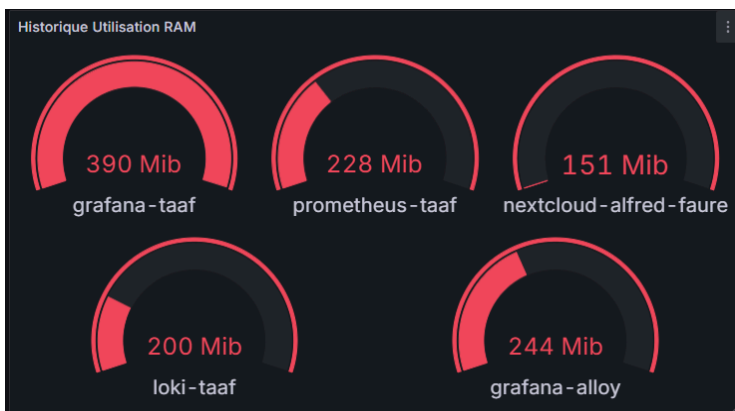
### Question 3 – Filtrage par conteneur

Top 5 conteneurs mémoire.

PromQL :

```
topk(5, container_memory_usage_bytes{name!="", instance=~"$host", name=~"$conteneur"})
```

Dashboard final :



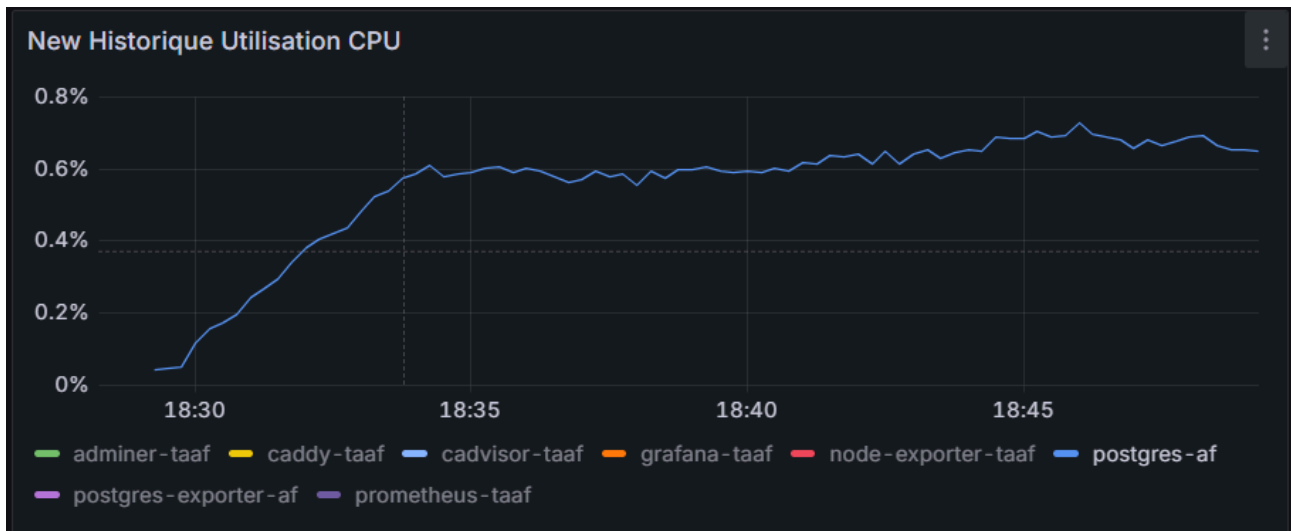
---

### Question 4 – Pourcentage CPU du conteneur **postgres-af**

PromQL :

```
avg(rate(container_cpu_usage_seconds_total{name="postgres-af"}[5m])) * 100
```

Dashboard final :

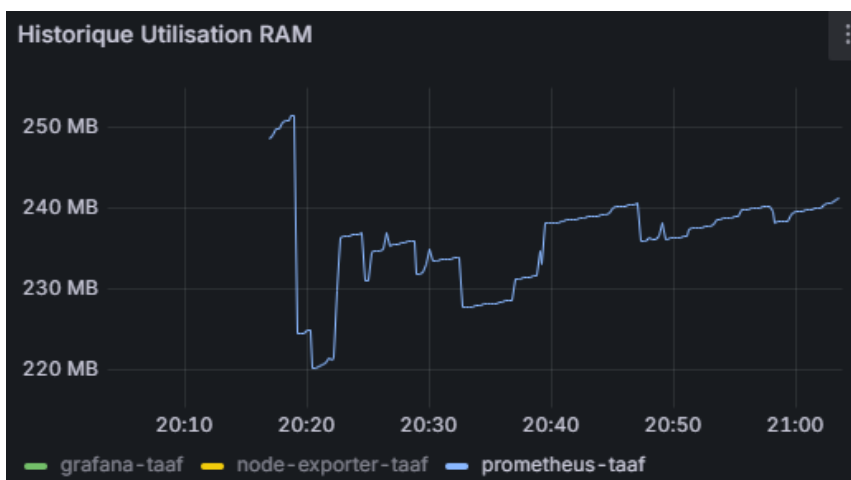


Consommation en moyenne : **0,68 %** de CPU

## Question 5 – Analyse temporelle

Pic mémoire du conteneur **prometheus-taaf** sur la dernière heure.

Dashboard final :



**Réponse :** Au cours de la dernière heure, le conteneur *prometheus* a atteint un pic de consommation mémoire de **251 MB** vers **20:18**, probablement lors d'une opération de collecte ou d'évaluation de requêtes lourdes.

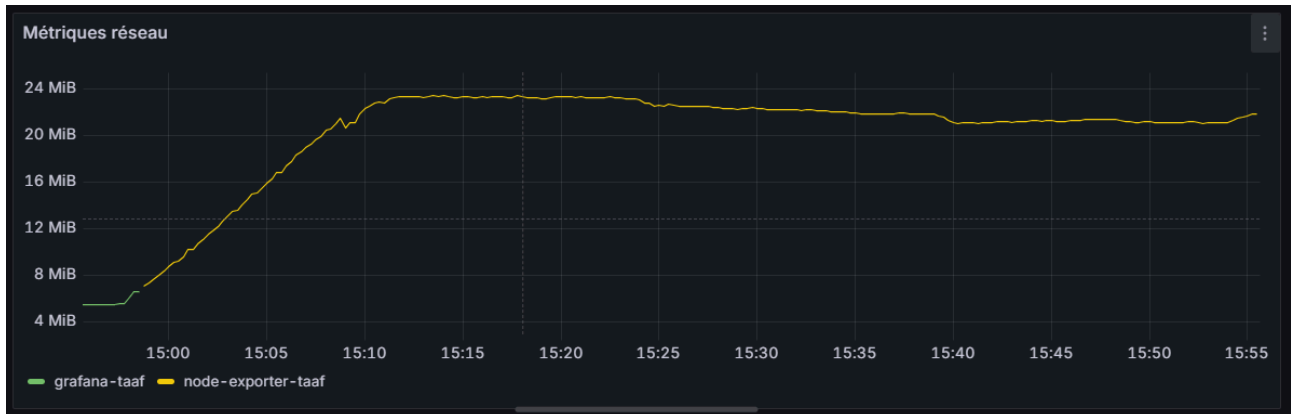
## Question 6 – Métriques réseau

Conteneur avec le plus de trafic sortant (TX bytes).

PromQL :

```
topk(1, sum(rate(container_network_transmit_bytes_total{name!="",
instance=~"$host", name=~"$conteneur"}[1h])) by (name))
```

Dashboard final :



**Réponse** : Durant la dernière heure, le conteneur postgres-af a généré le plus de trafic réseau sortant avec environ **22 MiB**, lié à l'envoi continu de métriques système vers Prometheus.

## Question 7 – Corrélation métrique

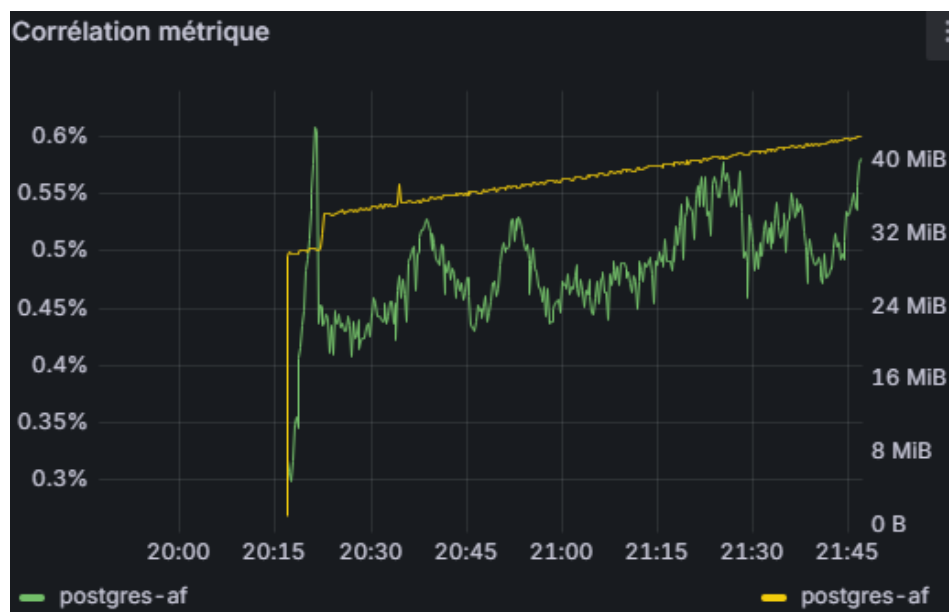
CPU vs Mémoire sur 2h.

- CPU :

```
rate(container_cpu_usage_seconds_total{name="postgres-af",
instance=~"$host"}[5m]) * 100
```

- RAM :

```
container_memory_usage_bytes{name="postgres-af", instance=~"$host"}
```



En vert : CPU

En jaune : RAM

**Réponse :** On observe une corrélation positive entre l'utilisation CPU et la mémoire du conteneur postgres-af.

Les pics CPU correspondent à des phases de traitement ou d'accès intensif aux données, suivies d'une augmentation progressive de la mémoire utilisée.

Ce comportement est cohérent avec le fonctionnement de PostgreSQL, où la charge CPU ponctuelle alimente une montée en mémoire due au caching et à la gestion des buffers internes.

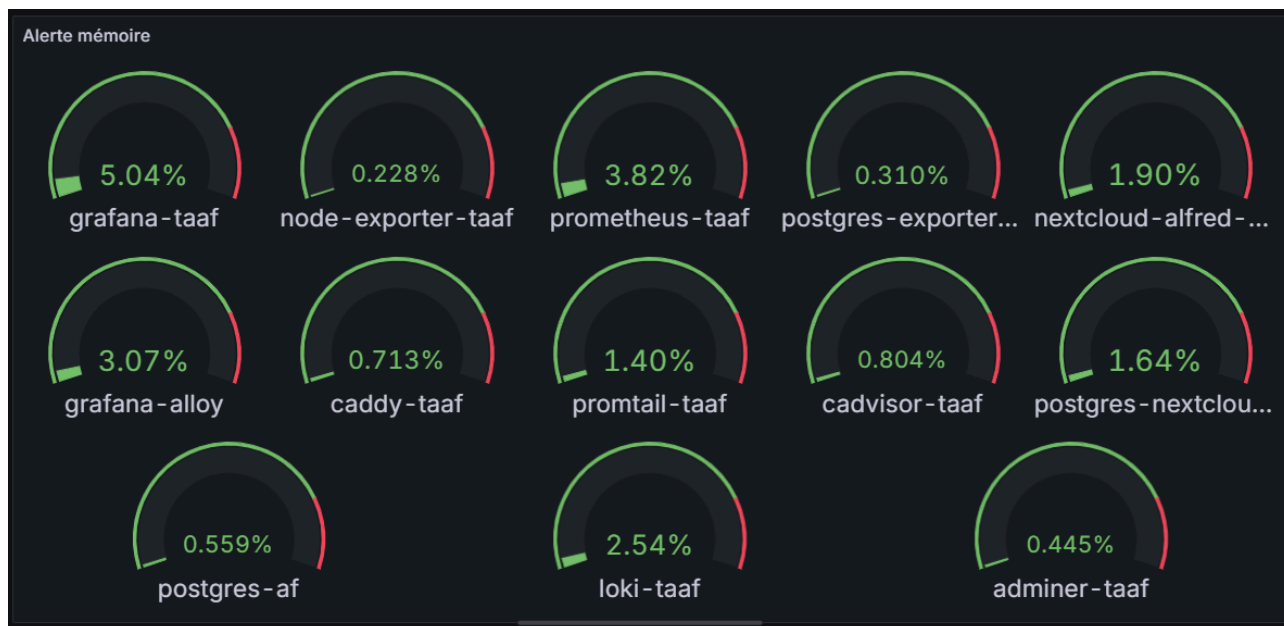
## Question 8 – Alertes mémoire

Conteneurs avec >80% mémoire.

On prend

PromQL (si les limites mémoire existent via `container_spec_memory_limit_bytes`) :

```
(container_memory_usage_bytes{name!="", job="cadvisor"}  
  / (7.762 * 1024 * 1024 * 1024)  
  ) * 100 > 0
```



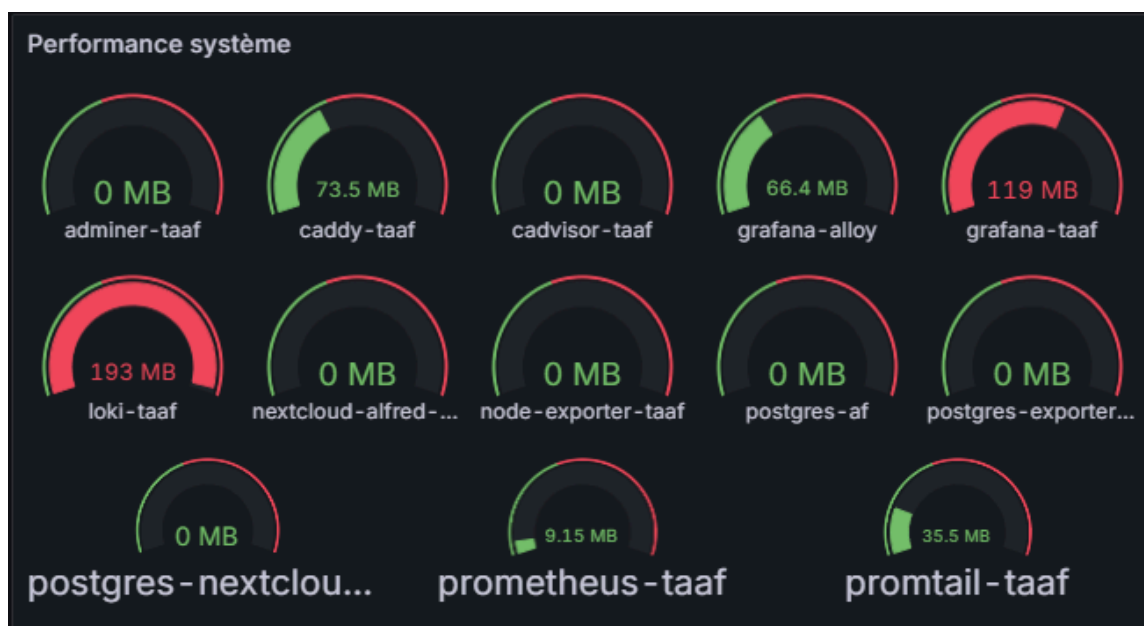
**Réponse :** Après avoir défini des limites mémoire sur les conteneurs Docker, Prometheus rapporte désormais des valeurs correctes. On voit les conteneurs *qui peuvent* dépasser 80 % de leur limite , ce qui déclenche l'alerte mémoire. Les autres conteneurs restent en dessous du seuil.

## Question 9 – Impact I/O disque

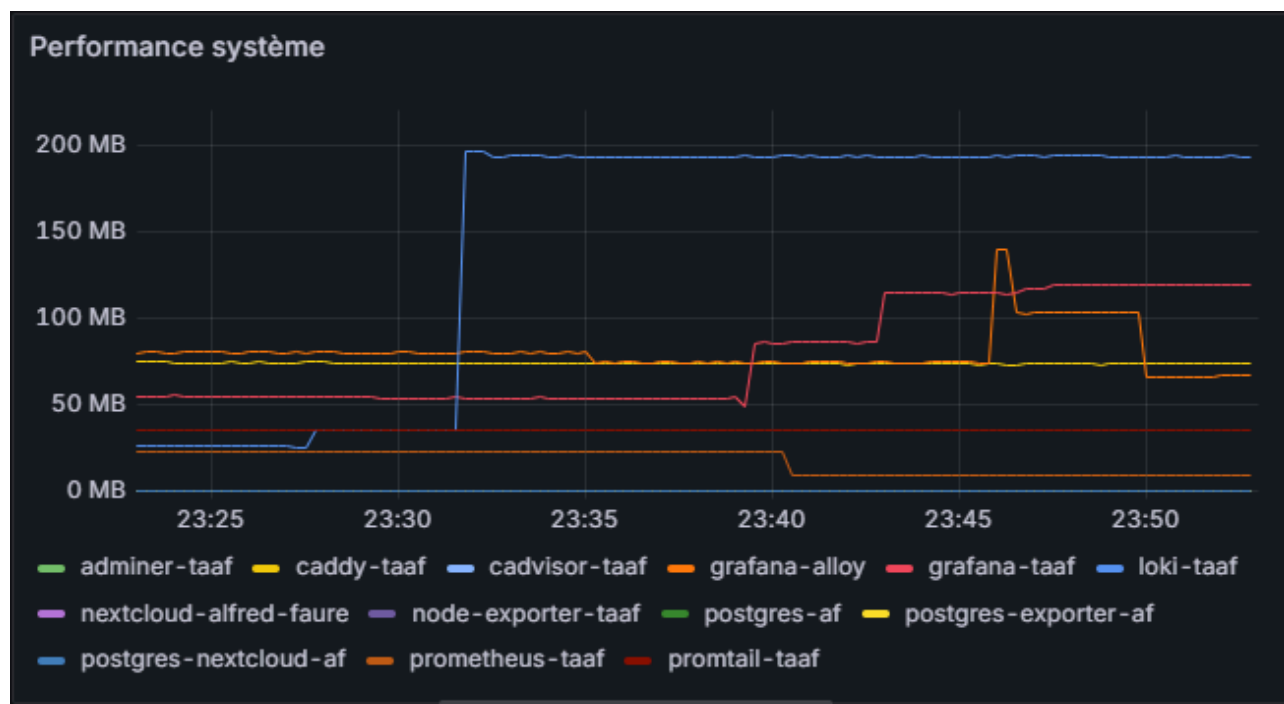
Quel conteneur fait le plus d'I/O disque ?

PromQL :

```
sum(rate(container_fs_reads_bytes_total{name!="", instance=~"$host",
name=~"$conteneur"}[1h])) by (name)
```







**Réponse :** Durant la dernière heure, le conteneur *loki-taaf* a généré le plus d'I/O disque (~193 MB), principalement pour écrire et indexer les logs entrants. Cela peut affecter la performance globale du host si le disque n'est pas suffisamment rapide, tandis que les autres conteneurs ont une activité disque moindre.

### Impact opérationnel

- Le conteneur *loki-taaf* peut saturer le disque si le flux de logs est élevé.
- Risque : ralentissement du host pour les autres conteneurs, délais d'accès aux fichiers, latence pour Prometheus et Grafana.
- Solution : stockage rapide (SSD), quotas I/O, ou isolation sur un disque dédié.

## Question 10 – Optimisation infrastructure

Propositions possibles :

### 1- Optimisation Postgres

- **Action :** Mettre en place une alerte si le CPU du conteneur Postgres dépasse 70 % sur 10 min.
- **Justification technique :** Permet d'identifier les pics de charge dus aux requêtes lourdes ou aux index manquants. Optimiser les requêtes ou les indices réduit le risque de saturation CPU.

- **Impact opérationnel** : Maintien des performances de la base de données, garantie de disponibilité des services bancaires simulés, réduction des ralentissements pour les utilisateurs.

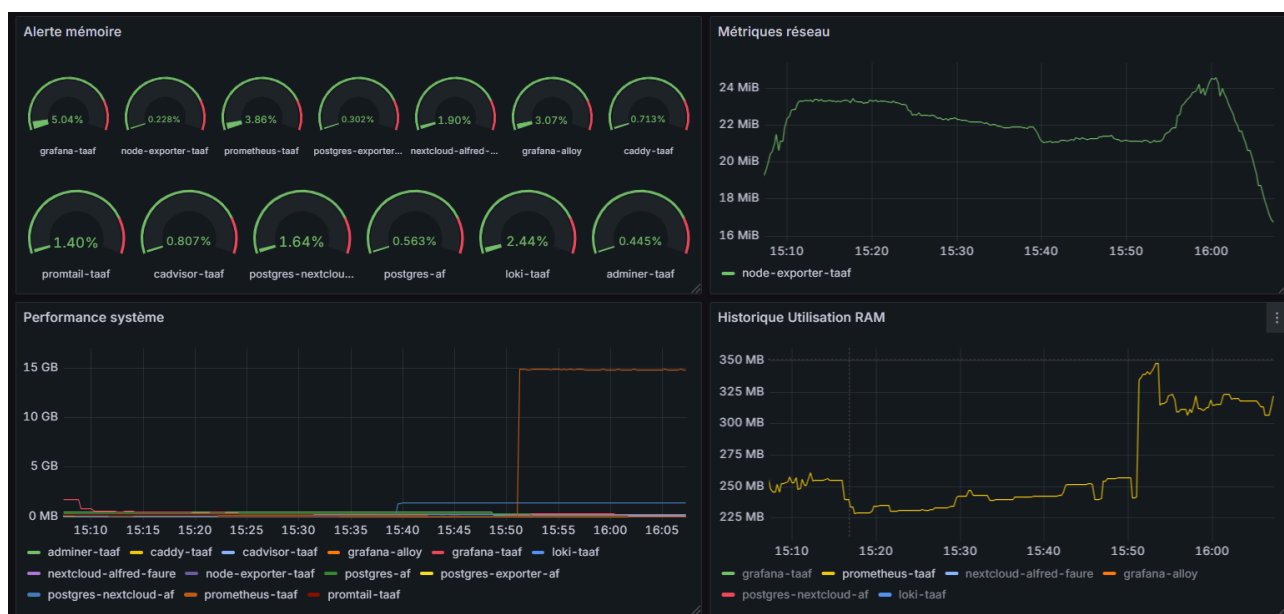
## 2- Limitation mémoire pour Grafana et Prometheus

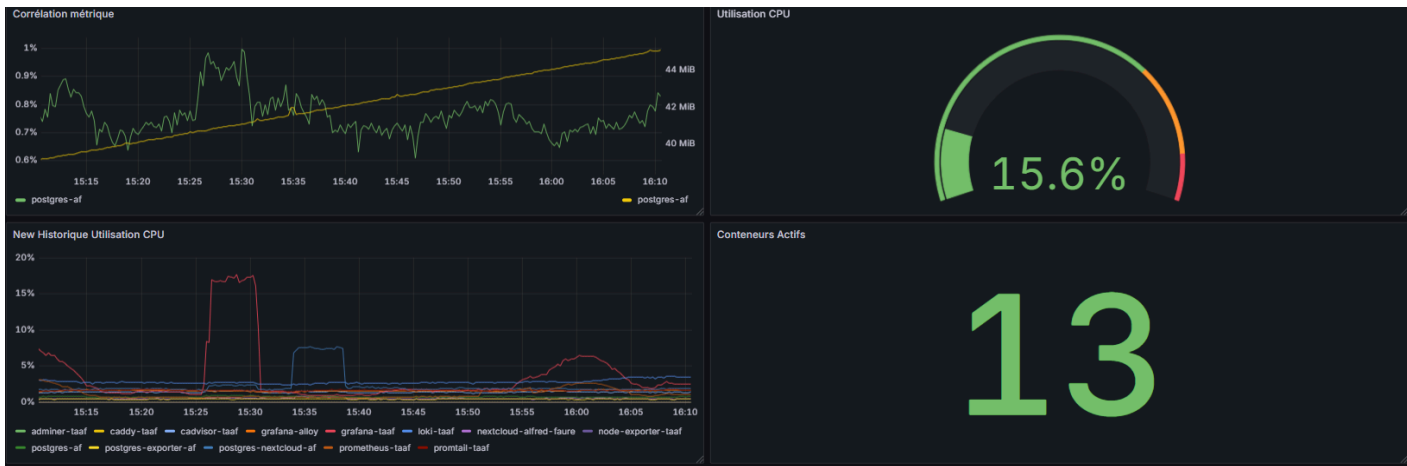
- **Action** : Définir des limites mémoire sur les conteneurs et créer une alerte si utilisation > 80 %.
- **Justification technique** : Ces conteneurs consomment beaucoup de RAM pour les dashboards et la collecte de métriques. Limiter la mémoire évite le swapping et les crashes.
- **Impact opérationnel** : Assure la continuité du monitoring et la fiabilité des métriques affichées dans Grafana.

## 3- Surveillance du trafic réseau

- **Action** : Créer une alerte si un conteneur dépasse un débit sortant moyen de 50 MB/s.
- **Justification technique** : Permet de détecter les conteneurs générant un trafic réseau excessif pouvant saturer le lien ou impacter la communication entre conteneurs.
- **Impact opérationnel** : Prévention des congestions réseau, maintien de la collecte de métriques et de la communication inter-conteneurs, garantissant la stabilité de l'infrastructure :

Dashboard final :





## ANNEXE JSON :

```
{
  "id": 16,
  "type": "timeseries",
  "title": " Performance système",
  "gridPos": {
    "x": 0,
    "y": 0,
    "h": 9,
    "w": 13
  },
  "fieldConfig": {
    "defaults": {
      "custom": {
        "drawStyle": "line",
        "lineInterpolation": "linear",
        "barAlignment": 0,
        "barWidthFactor": 0.6,
        "lineWidth": 1,
        "fillOpacity": 0,
        "gradientMode": "none",
        "spanNulls": false,
        "insertNulls": false,
        "showPoints": "auto",
        "pointSize": 5,

```

```
"stacking": {
  "mode": "none",
  "group": "A"
},
"axisPlacement": "auto",
"axisLabel": "",
"axisColorMode": "text",
"axisBorderShow": false,
"scaleDistribution": {
  "type": "linear"
},
"axisCenteredZero": false,
"hideFrom": {
  "tooltip": false,
  "viz": false,
  "legend": false
},
"thresholdsStyle": {
  "mode": "off"
}
},
"color": {
  "mode": "palette-classic"
},
"mappings": [],
"thresholds": {
  "mode": "absolute",
  "steps": [
    {
      "color": "green",
      "value": null
    },
    {
      "color": "red",
      "value": 80
    }
  ]
},
```

```
    "unit": "decbytes"
  },
  "overrides": []
},
"pluginVersion": "12.1.1",
"targets": [
  {
    "editorMode": "code",
    "expr": "sum(rate(container_fs_reads_bytes_total{name!=\"\", instance=~\"$host\", name=~\"$conteneur\"}[1h])) by (name)\r\n",
    "legendFormat": "{{name}}",
    "range": true,
    "refId": "A"
  }
],
"datasource": {
  "type": "prometheus",
  "uid": "dezjs3gl35r7kc"
},
"options": {
  "tooltip": {
    "mode": "single",
    "sort": "none",
    "hideZeros": false
  },
  "legend": {
    "showLegend": true,
    "displayMode": "list",
    "placement": "bottom",
    "calcs": []
  }
}
}
```

```
{
  "id": 15,
  "type": "gauge",
  "title": "Alerte mémoire",
  "gridPos": {
    "x": 13,
    "y": 0,
    "h": 9,
    "w": 11
  },
  "fieldConfig": {
    "defaults": {
      "mappings": [],
      "thresholds": {
        "mode": "absolute",
        "steps": [
          {
            "color": "green",
            "value": null
          },
          {
            "color": "red",
            "value": 80
          }
        ]
      },
      "color": {
        "mode": "thresholds"
      },
      "unit": "decbytes"
    },
    "overrides": []
  },
  "pluginVersion": "12.1.1",
  "targets": [
    {
```

```

        "editorMode": "code",
        "expr": "(container_memory_usage_bytes{name!=\"\"},
instance=~\"$host\") / 1024 / 1024) > 160\\r\\n",
        "legendFormat": "{{name}}",
        "range": true,
        "refId": "A"
    }
],
"datasource": {
    "type": "prometheus",
    "uid": "dezjs3gl35r7kc"
},
"options": {
    "reduceOptions": {
        "values": false,
        "calcs": [
            "lastNotNull"
        ],
        "fields": ""
    },
    "orientation": "auto",
    "showThresholdLabels": false,
    "showThresholdMarkers": true,
    "sizing": "auto",
    "minVizWidth": 75,
    "minVizHeight": 75
}
}

```

```

{
    "id": 8,
    "type": "timeseries",
    "title": "Corrélation métrique",
    "gridPos": {
        "x": 0,
        "y": 9,
        "h": 8,

```

```
"w": 13
},
"fieldConfig": {
  "defaults": {
    "custom": {
      "drawStyle": "line",
      "lineInterpolation": "linear",
      "barAlignment": 0,
      "barWidthFactor": 0.6,
      "lineWidth": 1,
      "fillOpacity": 0,
      "gradientMode": "none",
      "spanNulls": false,
      "insertNulls": false,
      "showPoints": "auto",
      "pointSize": 5,
      "stacking": {
        "mode": "none",
        "group": "A"
      },
      "axisPlacement": "auto",
      "axisLabel": "",
      "axisColorMode": "text",
      "axisBorderShow": false,
      "scaleDistribution": {
        "type": "linear"
      },
      "axisCenteredZero": false,
      "hideFrom": {
        "tooltip": false,
        "viz": false,
        "legend": false
      },
      "thresholdsStyle": {
        "mode": "off"
      }
    },
  },
  "color": {
```



```
    "mode": "palette-classic"
  },
  "mappings": [],
  "thresholds": {
    "mode": "absolute",
    "steps": [
      {
        "color": "green",
        "value": null
      },
      {
        "color": "red",
        "value": 80
      }
    ]
  },
  "unit": "percent"
},
"overrides": [
  {
    "matcher": {
      "id": "byFrameRefID",
      "options": "B"
    },
    "properties": [
      {
        "id": "custom.axisPlacement",
        "value": "right"
      },
      {
        "id": "unit",
        "value": "bytes"
      }
    ]
  }
]
},
"pluginVersion": "12.1.1",
```

```
"targets": [
  {
    "editorMode": "code",
    "expr":
"sum(rate(container_cpu_usage_seconds_total{name=~\"postgres-af\",
instance=~\"$host\"}[5m])) by (name) * 100",
    "legendFormat": "__auto",
    "range": true,
    "refId": "A"
  },
  {
    "datasource": {
      "type": "prometheus",
      "uid": "dezjs3gl35r7kc"
    },
    "editorMode": "code",
    "expr": "sum(container_memory_usage_bytes{name=~\"postgres-af\",
instance=~\"$host\"}) by (name)",
    "hide": false,
    "instant": false,
    "legendFormat": "__auto",
    "range": true,
    "refId": "B"
  }
],
"datasource": {
  "type": "prometheus",
  "uid": "dezjs3gl35r7kc"
},
"options": {
  "tooltip": {
    "mode": "single",
    "sort": "none",
    "hideZeros": false
  },
  "legend": {
    "showLegend": true,
    "displayMode": "list",
```

```
    "placement": "bottom",
    "calcs": []
  }
}
}
```

```
{
  "id": 6,
  "type": "timeseries",
  "title": "Métriques réseau",
  "gridPos": {
    "x": 13,
    "y": 9,
    "h": 8,
    "w": 11
  },
  "fieldConfig": {
    "defaults": {
      "custom": {
        "drawStyle": "line",
        "lineInterpolation": "linear",
        "barAlignment": 0,
        "barWidthFactor": 0.6,
        "lineWidth": 1,
        "fillOpacity": 0,
        "gradientMode": "none",
        "spanNulls": false,
        "insertNulls": false,
        "showPoints": "auto",
        "pointSize": 5,
        "stacking": {
          "mode": "none",
          "group": "A"
        },
        "axisPlacement": "auto",
        "axisLabel": "",
        "axisColorMode": "text",
```

```
    "axisBorderShow": false,
    "scaleDistribution": {
      "type": "linear"
    },
    "axisCenteredZero": false,
    "hideFrom": {
      "tooltip": false,
      "viz": false,
      "legend": false
    },
    "thresholdsStyle": {
      "mode": "off"
    }
  },
  "color": {
    "mode": "palette-classic"
  },
  "mappings": [],
  "thresholds": {
    "mode": "absolute",
    "steps": [
      {
        "color": "green",
        "value": null
      },
      {
        "color": "red",
        "value": 80
      }
    ]
  },
  "unit": "kbytes"
},
"overrides": []
},
"pluginVersion": "12.1.1",
"targets": [
  {
```

```

        "editorMode": "code",
        "expr": "topk(1,
sum(rate(container_network_transmit_bytes_total{name!=\"\",
instance=~\"$host\", name=~\"$conteneur\"}[1h])) by (name))",
        "legendFormat": "{{name}}",
        "range": true,
        "refId": "A"
    }
],
"datasource": {
    "type": "prometheus",
    "uid": "dezjs3gl35r7kc"
},
"options": {
    "tooltip": {
        "mode": "single",
        "sort": "none",
        "hideZeros": false
    },
    "legend": {
        "showLegend": true,
        "displayMode": "list",
        "placement": "bottom",
        "calcs": []
    }
}
}

```

```

{
    "id": 1,
    "type": "stat",
    "title": "Conteneurs Actifs",
    "gridPos": {
        "x": 0,
        "y": 32,
        "h": 8,
        "w": 12
    }
}

```

```
},
"fieldConfig": {
  "defaults": {
    "mappings": [],
    "thresholds": {
      "mode": "absolute",
      "steps": [
        {
          "color": "green",
          "value": null
        },
        {
          "color": "red",
          "value": 80
        }
      ]
    },
  },
  "color": {
    "mode": "thresholds"
  },
},
"overrides": [],
},
"pluginVersion": "12.1.1",
"targets": [
  {
    "datasource": {
      "type": "prometheus",
      "uid": "dezjs3gl35r7kc"
    },
    "editorMode": "code",
    "exemplar": false,
    "expr": "count(container_last_seen{name!=\"\", instance=~\"$host\"})",
    "format": "time_series",
    "instant": true,
    "legendFormat": "__auto",
    "range": false,
    "refId": "A"
  }
]
```

```
    }
  ],
  "datasource": {
    "type": "prometheus",
    "uid": "dezjs3gl35r7kc"
  },
  "options": {
    "reduceOptions": {
      "values": false,
      "calcs": [
        "lastNotNull"
      ],
      "fields": ""
    },
    "orientation": "auto",
    "textMode": "auto",
    "wideLayout": true,
    "colorMode": "value",
    "graphMode": "area",
    "justifyMode": "auto",
    "showPercentChange": false,
    "percentChangeColorMode": "standard"
  }
}
```

```
{
  "id": 3,
  "type": "gauge",
  "title": "Utilisation RAM",
  "gridPos": {
    "x": 0,
    "y": 26,
    "h": 6,
    "w": 12
  },
  "fieldConfig": {
    "defaults": {
```

```
"mappings": [],
"thresholds": {
  "mode": "absolute",
  "steps": [
    {
      "color": "green",
      "value": null
    },
    {
      "color": "red",
      "value": 80
    }
  ]
},
"color": {
  "mode": "thresholds"
},
"unit": "bytes"
},
"overrides": []
},
"pluginVersion": "12.1.1",
"targets": [
  {
    "editorMode": "code",
    "expr": "sum(container_memory_usage_bytes{name=~\"$conteneur\",
instance=~\"$host\"})",
    "legendFormat": "__auto",
    "range": true,
    "refId": "A"
  }
],
"datasource": {
  "type": "prometheus",
  "uid": "dezjs3gl35r7kc"
},
"options": {
  "reduceOptions": {
```



```
    "values": false,
    "calcs": [
        "lastNotNull"
    ],
    "fields": ""
},
"orientation": "auto",
"showThresholdLabels": false,
"showThresholdMarkers": true,
"sizing": "auto",
"minVizWidth": 75,
"minVizHeight": 75
}
}
```

```
{
  "id": 5,
  "type": "timeseries",
  "title": "New Historique Utilisation CPU",
  "gridPos": {
    "x": 0,
    "y": 17,
    "h": 9,
    "w": 13
  },
  "fieldConfig": {
    "defaults": {
      "custom": {
        "drawStyle": "line",
        "lineInterpolation": "linear",
        "barAlignment": 0,
        "barWidthFactor": 0.6,
        "lineWidth": 1,
        "fillOpacity": 0,
        "gradientMode": "none",
        "spanNulls": false,
        "insertNulls": false,

```

```
"showPoints": "auto",
"pointSize": 5,
"stacking": {
  "mode": "none",
  "group": "A"
},
"axisPlacement": "auto",
"axisLabel": "",
"axisColorMode": "text",
"axisBorderShow": false,
"scaleDistribution": {
  "type": "linear"
},
"axisCenteredZero": false,
"hideFrom": {
  "tooltip": false,
  "viz": false,
  "legend": false
},
"thresholdsStyle": {
  "mode": "off"
}
},
"color": {
  "mode": "palette-classic"
},
"mappings": [],
"thresholds": {
  "mode": "absolute",
  "steps": [
    {
      "color": "green",
      "value": null
    },
    {
      "color": "red",
      "value": 80
    }
  ]
}
```

```
    ],
    },
    "unit": "percent"
  },
  "overrides": []
},
"pluginVersion": "12.1.1",
"targets": [
  {
    "editorMode": "code",
    "expr": "sum(rate(container_cpu_usage_seconds_total{name!=\"\", instance=~\"$host\", name=~\"$conteneur\"}[5m])) by (name) * 100",
    "legendFormat": "{{name}}",
    "range": true,
    "refId": "A"
  }
],
"datasource": {
  "type": "prometheus",
  "uid": "dezjs3gl35r7kc"
},
"options": {
  "tooltip": {
    "mode": "single",
    "sort": "none",
    "hideZeros": false
  },
  "legend": {
    "showLegend": true,
    "displayMode": "list",
    "placement": "bottom",
    "calcs": []
  }
}
}
```

```
{
  "id": 4,
  "type": "timeseries",
  "title": "Historique Utilisation RAM",
  "gridPos": {
    "x": 13,
    "y": 17,
    "h": 9,
    "w": 11
  },
  "fieldConfig": {
    "defaults": {
      "custom": {
        "drawStyle": "line",
        "lineInterpolation": "linear",
        "barAlignment": 0,
        "barWidthFactor": 0.6,
        "lineWidth": 1,
        "fillOpacity": 0,
        "gradientMode": "none",
        "spanNulls": false,
        "insertNulls": false,
        "showPoints": "auto",
        "pointSize": 5,
        "stacking": {
          "mode": "none",
          "group": "A"
        },
        "axisPlacement": "auto",
        "axisLabel": "",
        "axisColorMode": "text",
        "axisBorderShow": false,
        "scaleDistribution": {
          "type": "linear"
        },
        "axisCenteredZero": false,
```

```
    "hideFrom": {
      "tooltip": false,
      "viz": false,
      "legend": false
    },
    "thresholdsStyle": {
      "mode": "off"
    }
  },
  "color": {
    "mode": "palette-classic"
  },
  "mappings": [],
  "thresholds": {
    "mode": "absolute",
    "steps": [
      {
        "color": "green",
        "value": null
      },
      {
        "color": "red",
        "value": 80
      }
    ]
  },
  "unit": "decbytes"
},
"overrides": [
  {
    "__systemRef": "hideSeriesFrom",
    "matcher": {
      "id": "byNames",
      "options": {
        "mode": "exclude",
        "names": [
          "prometheus-taaf"
        ]
      }
    }
  }
]
```

```
        "prefix": "All except:",
        "readOnly": true
    },
    "properties": [
        {
            "id": "custom.hideFrom",
            "value": {
                "legend": false,
                "tooltip": false,
                "viz": true
            }
        }
    ]
}
]
},
"pluginVersion": "12.1.1",
"targets": [
    {
        "editorMode": "code",
        "expr": "topk(5, container_memory_usage_bytes{name!=\"\",
instance=~\"$host\", name=~\"$conteneur\"})\r\n",
        "legendFormat": "{{name}}",
        "range": true,
        "refId": "A"
    }
],
"datasource": {
    "type": "prometheus",
    "uid": "dezjs3gl35r7kc"
},
"options": {
    "tooltip": {
        "mode": "single",
        "sort": "none",
        "hideZeros": false
    },

```

```
"legend": {
  "showLegend": true,
  "displayMode": "list",
  "placement": "bottom",
  "calcs": []
}
}
```

```
{
  "id": 2,
  "type": "gauge",
  "title": "Utilisation CPU",
  "gridPos": {
    "x": 12,
    "y": 26,
    "h": 6,
    "w": 12
  },
  "fieldConfig": {
    "defaults": {
      "mappings": [],
      "thresholds": {
        "mode": "absolute",
        "steps": [
          {
            "color": "green",
            "value": null
          },
          {
            "color": "orange",
            "value": 70
          },
          {
            "color": "red",
            "value": 90
          }
        ]
      }
    }
  }
}
```

```
    ],
    },
    "color": {
        "mode": "thresholds"
    },
    "unit": "percent"
},
"overrides": []
},
"pluginVersion": "12.1.1",
"targets": [
    {
        "editorMode": "code",
        "expr":
"sum(rate(container_cpu_usage_seconds_total{name=~\"$conteneur\",
instance=~\"$host\"}[5m])) * 100",
        "legendFormat": "__auto",
        "range": true,
        "refId": "A"
    }
],
"datasource": {
    "type": "prometheus",
    "uid": "dezjs3gl35r7kc"
},
"options": {
    "reduceOptions": {
        "values": false,
        "calcs": [
            "lastNotNull"
        ],
        "fields": ""
    },
    "orientation": "auto",
    "showThresholdLabels": false,
    "showThresholdMarkers": true,
    "sizing": "auto",
    "minVizWidth": 75,
```



```
    "minVizHeight": 75  
  }  
}
```