Documentation de la Plateforme

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Mis à jour par | Révisé par |
| 1.0 | 26/06/2018 | Philippe Mondon |  |

[Description de la Plateforme 2](#_Toc517795400)

[Normalisation des Indexes et Pipelines 3](#_Toc517795401)

[Indexes de données - Plateforme de développement 3](#_Toc517795402)

[Indexes Templates de données - Plateforme de développement 3](#_Toc517795403)

[Ingest Node Pipelines Simulate – Plateforme de développement 3](#_Toc517795404)

[Ingest Node Pipelines – Plateforme de développement 3](#_Toc517795405)

[Indexes de monitoring – Plateforme de développement 3](#_Toc517795406)

[Serveur de développement 4](#_Toc517795407)

[Création des services systemd filebeats spécifiques 4](#_Toc517795408)

[Serveur d’Intégration 4](#_Toc517795409)

[Création d’un service systemd spécifiques 4](#_Toc517795410)

[Monitoring de la plateforme 5](#_Toc517795411)

[Monitoring : Recherche d’un processus filebeat particulier 5](#_Toc517795412)

[Processus de mise en place du monitoring de processus 5](#_Toc517795413)

[Monitoring de Qualité de données 6](#_Toc517795414)

[Duplication d’enregistrements 6](#_Toc517795415)

[Contrôle du nombre d’enregistrements intégrés 6](#_Toc517795416)

[REFERENCES 8](#_Toc517795417)

[Lignes de commandes 8](#_Toc517795418)

# Description de la Plateforme

# Normalisation des Indexes et Pipelines

## Indexes de données - Plateforme de développement

*Nom des indexes stockés dans la plateforme*

Index MNC : tpgbam \_dev\_mnc \_index

Index SCS : tpgbam \_dev\_scs \_index

Index SDS : tpgbam \_dev\_sds\_index

Le positionnement du préfix « tpgbam\_dev » permettra dans Kibana de créer un index pattern «  « tpgbam\_dev\* » et de ce fait de traiter ces indexes dans les dashboards de façon unifiée.

## Indexes Templates de données - Plateforme de développement

*Nom des index templates stockés dans la plateforme*

Index Template MNC : tpgbam\_mnc\_dev\_index\_template

Index Template SCS : tpgbam\_scs\_dev\_index\_template

Index Template SDS : tpgbam\_sds\_dev\_index\_template

## Ingest Node Pipelines Simulate – Plateforme de développement

*Nom du fichier stockant le pipeline simulate.*

*ATTENTION : aucun pipeline simulate n’est stocké dans la plateforme. La définition contenue dans ces fichiers est utilisée que dans l’environnement de développement*

Pipeline Simulate MNC : tpgbam\_mnc\_dev\_pipeline\_simulate.json

Pipeline Simulate SCS : tpgbam\_scs\_dev\_pipeline\_simulate.json

Pipeline Simulate SDS : tpgbam\_sds\_dev\_pipeline\_simulate.json

## Ingest Node Pipelines – Plateforme de développement

Pipeline MNC : tpgbam\_mnc\_dev\_pipeline

Pipeline SCS : tpgbam\_scs\_dev\_pipeline

Pipeline SDS : tpgbam\_sds\_dev\_pipeline

## Indexes de monitoring – Plateforme de développement

User de monitoring :

Username : tpgbam \_monitoring\_dev\_user

Password : monitoringdev$1

Role : logstash\_system

Monitoring Notes :

For Elasticsearch 5.x clusters: Monitoring data is retained for three days by default or as specified by the xpack.monitoring.history.duration user setting.

# Serveur de développement

## Création des services systemd filebeats spécifiques

Filebeat-bench

Description : Service de test de filebeat

Configuration filebeat associée : /root/bench/filebeat.yml

Systemd Service configuration : /lib/systemd/system/filebeat-bench.service

# Serveur d’Intégration

## Création d’un service systemd spécifiques

Un service spécifique est créé pour chaque fournisseur de données :

* Exemples de nom du service : /lib/systemd/system/filebeat.mnc.integ.service
* Exemple de configuration associée : /etc/filebeat/filebeat.mnc.integ.yml

## Monitoring de la plateforme

## Monitoring : Recherche d’un processus filebeat particulier

Le monitoring des processus « filebeat » se fera grâce aux agents **metricbeat**.

Exemple de recherche de processus filebeat et classement par les dates pour ne garder que les plus récents depuis l’interface **Kibana>Dev Tools**:

get metricbeat-6.2.4-2018.06.12/\_search

{

"query": {

"bool": {

"must": [

{"match": {"system.process.name": "filebeat"}}

]

}

},

"\_source": ["@timestamp","system.process.name","system.process.cgroup.cpu.id"],

"sort": { "@timestamp": { "order": "desc"}},

"size": 100

}

### Processus de mise en place du monitoring de processus

* 1 : Installer MetricBeat sur la plateforme hébergeant le processus à monitorer
* 2 : Après mise en œuvre de MetricBeat et collecte de données, construire dans **Kibana>Management** un index-pattern de type **metricbeat\***,
* 3 : Créer une requête de recherche dans l’onglet **Kibana>Discover** et la sauvegarder
* 4 : Dans l’onglet **Kibana>Visualize**, utiliser la recherche sauvegardée pour créer un **Chart** destiné à visualiser la métrique du processus et le sauvegarder,
* 5 : Au besoin dans l’onglet **Kibana>Dashboard**, intégrer le nouveau chart à un Dashboard de monitoring complet.

**Exemple :** requête vers les indexes **metricbeat\*** sauvegardée dans l’onglet **Kibana>Discover** :

system.process.name : "filebeat" and system.process.cgroup.id:"filebeat.mnc.integ.service"

## Monitoring de Qualité de données

### Duplication d’enregistrements

La stratégie proposée repose sur la détection d’enregistrements dupliqués :

Dès qu’un enregistrement est dupliqué, il est détecté.

Cette technique permettra :

* De détecter la liste de données caractérisée par le As-Of / As-At intégrant la duplication,
* De détruire la liste de données caractérisée par le As-Of / As-At intégrant la duplication,
* De réinjecter la liste de données caractérisée par le As-Of / As-At incriminé.

ATTENTION :

Cette technique ne détecte pas tous les enregistrements dupliqués : c’est un indicateur de duplication sur une période.

POST tpgbam\_mnc\_dev\_test\_index/\_search

{

"aggs":{

"dedup" : {

"terms": {

"script": {

"lang": "painless",

"source":"doc['@timestamp'].value.toString() + doc['phone'].value"

},

"min\_doc\_count": 2

}

,

"aggs":{

"dedup\_docs":{

"top\_hits":{

}

}

}

}

}

}

### Contrôle du nombre d’enregistrements intégrés

La stratégie proposée compare le nombre d’enregistrements dans un fichier au nombre d’enregistrements intégrés dans l’index :

* On détecte le fichier à contrôler (technique à définir)
* On parse dans le nom de fichier, le As-Of et As-At correspondant aux enregistrements,
* On compte le nombre d’enregistrements dans le fichier :
  + wc -l [nom-du-fichier]
* On requête l’index pour récupérer le nombre d’enregistrements intégrés :
  + curl --user TPG\_BAM:antaes\*1 <https://416d580d881f24fb8cc3e9be04c9ce89.us-west-2.aws.found.io:9243/tpgbam_dev_mnc_index/_count?q=as_of:2018-06-14>

**Requête Elastic relative à un fichier ingéré :**

POST tpgbam\_dev\_mnc\_index/\_count

{

"query":

{

"bool": {

"should": [

{"match": { "as\_of\_time": "17:45:01"}},

{"match": { "as\_of": "2018-06-14"}}

],

"minimum\_should\_match": 2

}

}

}

**Requête http équivalente avec extraction du count en JSON :**

curl --user TPG\_BAM:antaes\*1 https://416d580d881f24fb8cc3e9be04c9ce89.us-west-2.aws.found.io:9243/tpgbam\_dev\_mnc\*/\_count -d '{ "query": { "bool": { "should": [ {"match": { "as\_of\_time": "17:45:01"}}, {"match": { "as\_of": "2018-06-14"}} ], "minimum\_should\_match": 2 } }}' -H'Content-Type: application/json' **| jq -r '.count'**

# REFERENCES

## Lignes de commandes

Filebeat :

sudo -u mnc /usr/bin/filebeat -e -c /home/mnc/filebeat.yml -path.home /home/mnc -path.config /home/mnc -path.data /home/mnc/fbregistry -path.logs /home/mnc/log/ -d "elasticsearch"