Université d'Artois

IUT de Béthune

Département Réseaux et Télécommunications

SAÉ5.CYBER03

Installation de la suite Elastic

par

Eliot HULEUX

Table des matières

1 Installation d'Elasticsearch

1.1 Prérequis

Nous installerons Elasticsearch sur une machine virtuelle Ubuntu Desktop 22.04.5. Des prérequis sont à prévoir :

— 2 Go de mémoire vive et 2 coeurs pour le CPU
— OpenJDK 11 installé

Pour installer OpenJDK 11 :

Linux 1

```
sudo apt update
sudo apt install openjdk-11-jdk -y
```

1.2 Ajout du dépôt et installation du paquet

Les composants Elasticsearch ne sont pas disponibles dans les dépôts de paquets par défaut d'Ubuntu.

Ajoutons la clé GPG publique à *apt* :

Linux 2

```
curl -fsSL https ://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add
-
```

Ajoutons la liste des sources Elastic au répertoire *sources.list.d* où *apt* cherchera de nouvelles sources :

Linux 3

```
echo "deb [signed-by=/usr/share/keyrings/elastic.gpg] https ://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list
```

Nous pouvons maintenant installer Elasticsearch :

Linux 4

```
sudo apt update
sudo apt install elasticsearch
```

2 Configuration d'Elasticsearch

Tout d'abord, Elasticsearch a ses fichiers de configuration écrit en YAML. Le respect de la syntaxe des fichiers est donc primordial.

Modifions le fichier *elasticsearch.yml* afin de n'autoriser que la machine et resteindre l'accès extérieur :

Linux 5

```
# ———————————————— Network ————————————————
#
#
network.host : localhost
```

```
  GNU nano 6.2          /etc/elasticsearch/elasticsearch.yml *
# on the system and that the owner of the process is allowed to use this
# limit.
#
# Elasticsearch performs poorly when the system is swapping the memory.
#
# ---------------------------------- Network -----------------------------------
#
# By default Elasticsearch is only accessible on localhost. Set a different
# address here to expose this node on the network:
#
network.host: localhost
#
# By default Elasticsearch listens for HTTP traffic on the first free port it
# finds starting at 9200. Set a specific HTTP port here:
#
#http.port: 9200
#
# For more information, consult the network module documentation.
#
```

Figure 1 – */etc/elasticsearch/elasticsearch.yml*

Démarrons maintenant le service et activons-le au démarrage :

Linux 6

sudo systemctl start elasticsearch

sudo systemctl enable elasticsearch

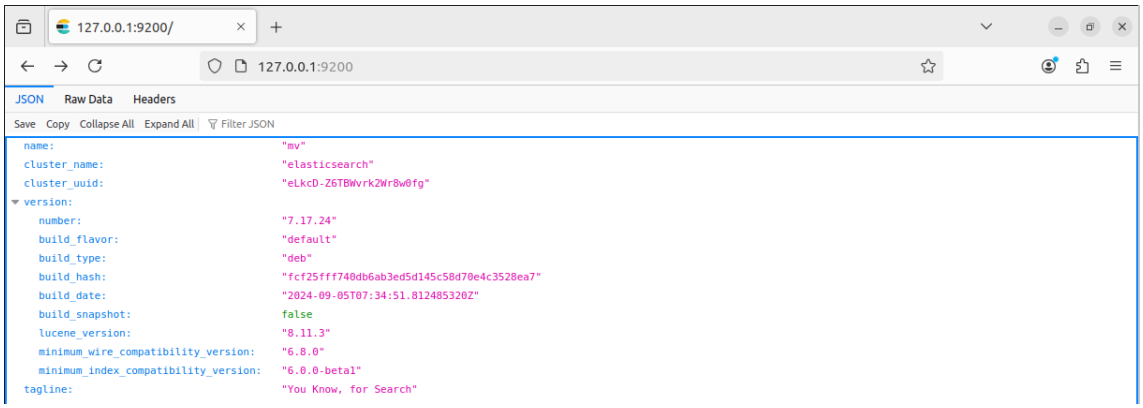

Figure 2 – *127.0.0.1 :9200.png*

Elasticsearch est maintenant bien configuré, passons à son interface visuelle, Kibana.

3 Installation de Kibana

Kibana est l'interface visuelle qui permet d'exploiter les données indexées dans Elasticsearch. Les deux outils fonctionnent ensemble pour offrir une solution complète d'analyse et de visualisation de données.

3.1 Installation du paquet et premier lancement

Étant donné que nous avons ajouté le dépôt de la suite Elastic, nous avons simplement à installer le paquet avec *apt* et le démarrer :

Linux 7

sudo apt install kibana -y

sudo systemctl start kibana

sudo systemctl enable kibana

Kibana est maintenant accessible depuis le port 5601 :

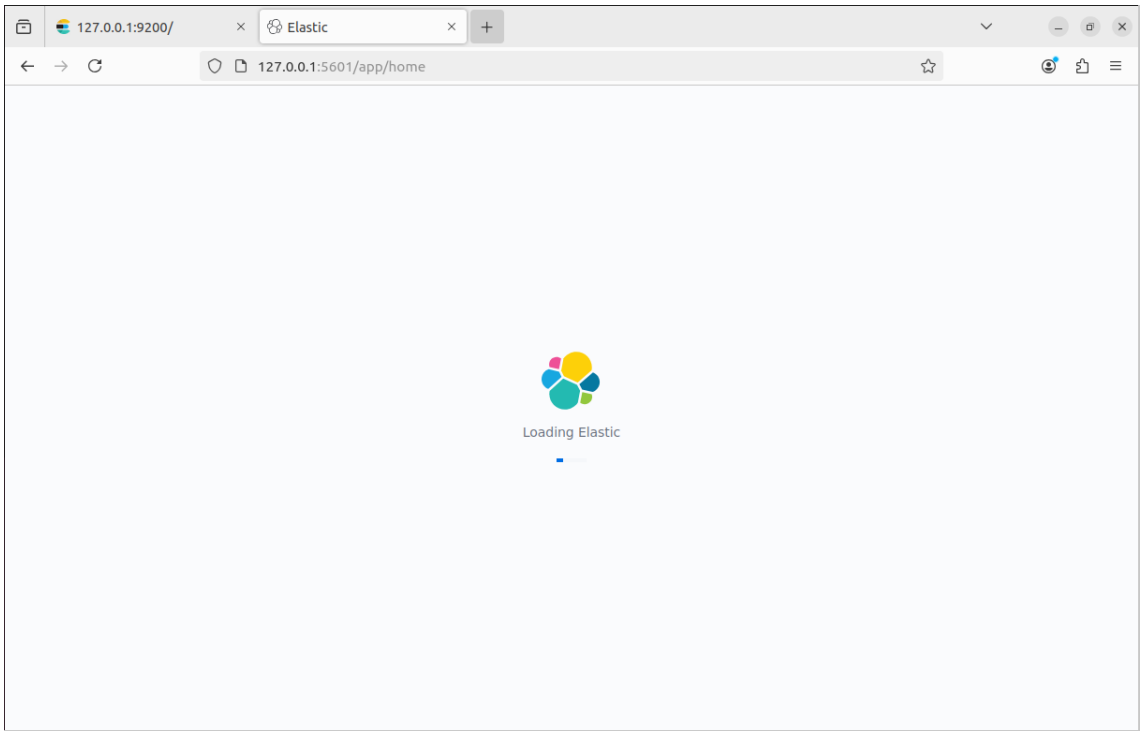

Figure 3 – *Accueil de Kibana*

4 Installation et configuration de Logstash

Linux 8

```
sudo apt install logstash -y
```

Un pipeline Logstash a deux éléments nécessaires, input et output, et un élément optionnel, filter. Les plugins d'entrée consomment des données à partir d'une source, les plugins de filtrage traitent les données et les plugins de sortie écrivent les données à une destination.

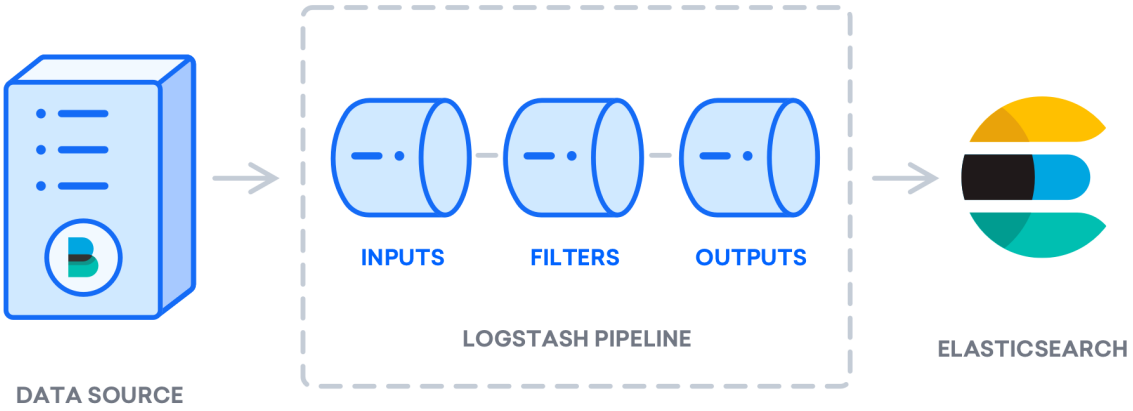

Figure 4 – *Logstash pipeline*

Créer un fichier de configuration appelé *02-beats-input.conf* pour les input :

Linux 9

```
input {
beats {
port => 5044
}
}
```

```
  GNU nano 6.2          /etc/logstash/conf.d/02-beats-input.conf *
input {
  beats {
    port => 5044
  }
}
```

Figure 5 – *02-beats-input.conf*

Faisons maintenant les sorties :

```
  GNU nano 6.2     /etc/logstash/conf.d/30-elasticsearch-output.conf
output {
  if [@metadata][pipeline] {
        elasticsearch {
        hosts => ["localhost:9200"]
        manage_template => false
        index => "%{[@metadata][beat]}-%{[@metadata][version]}-%{+YYYY.MM.dd}"
        pipeline => "%{[@metadata][pipeline]}"
        }
  } else {
        elasticsearch {
        hosts => ["localhost:9200"]
        manage_template => false
        index => "%{[@metadata][beat]}-%{[@metadata][version]}-%{+YYYY.MM.dd}"
        }
  }
}
```

Figure 6 – *30-elasticsearch-output.conf*

Nous pouvons maintenant démarrer *logstash* :

Linux 10

```
sudo systemctl start logstash
sudo systemctl enable logstash
```

5 Installation de Filebeat, Metricbeat, Packetbeat, Winlogbeat, Auditbeat & Heartbeat

5.1 Exemple avec Filebeat

Il y a simplement à installer le paquet et modifier deux fichiers de configurations :

Linux 11

```
sudo apt install filebeat -y
```

Il faut ensuite configurer Filebeat pour se connecter à Logstash :

Il faut ajouter # sur les paramètres suivants :

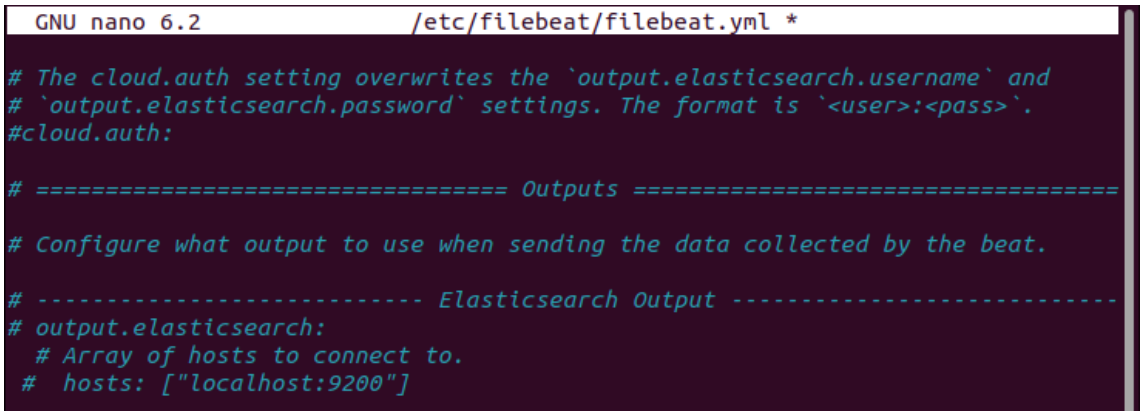

Figure 7 – *filebeat.yml*

Et décommenter ces lignes suivantes :

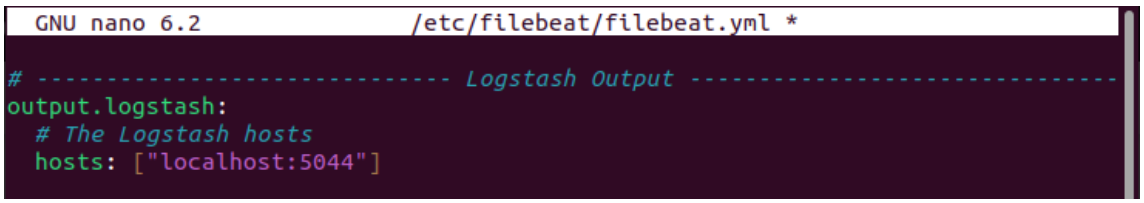

Figure 8 – *filebeat.yml*

La fonctionnalité de Filebeat peut être étendue avec les modules Filebeat.

Voici un exemple de modules que l'on peut activer :

Linux 12

sudo filebeat modules list

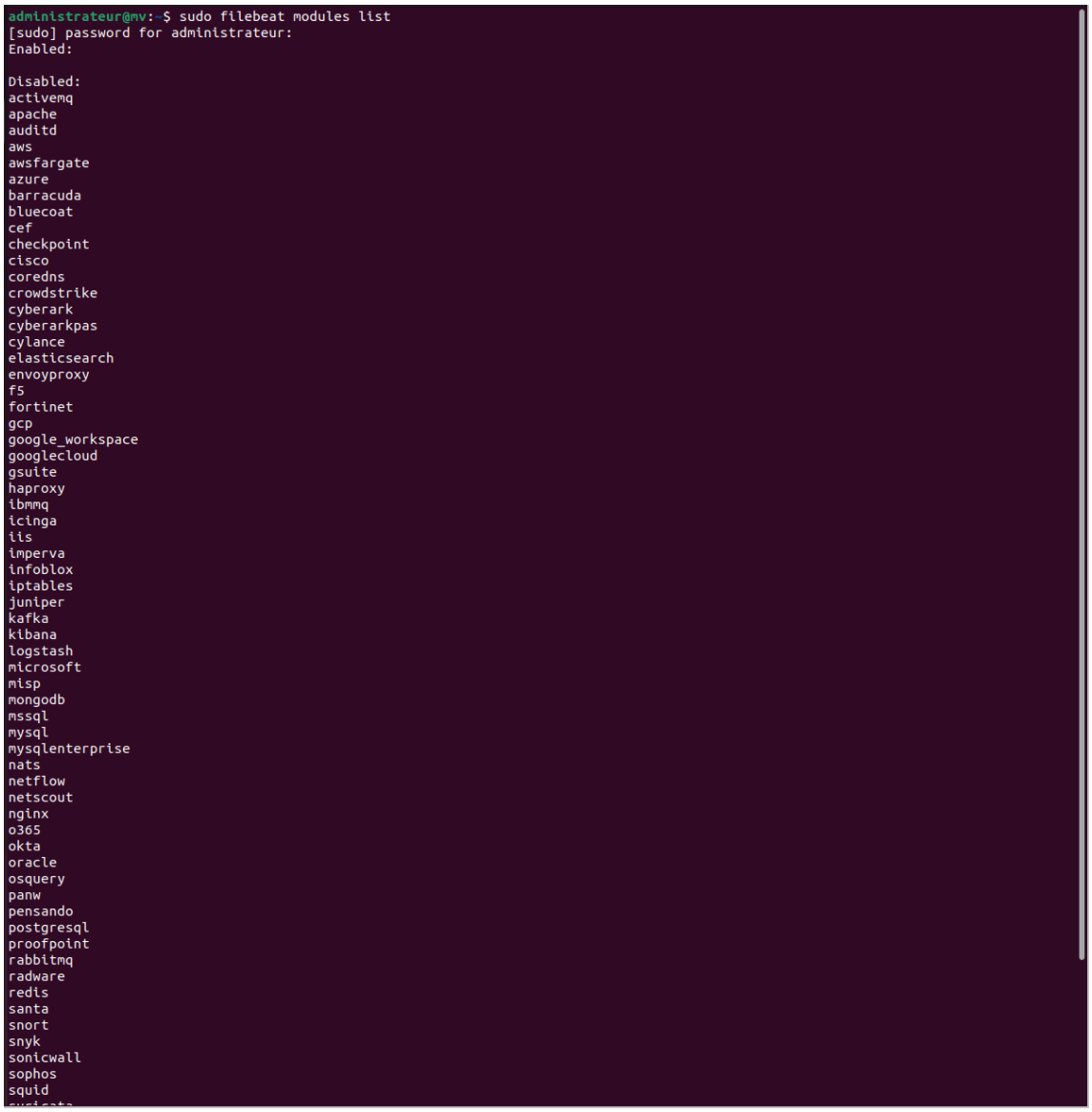

Figure 9 – *List des modules*

Si un des modules est activé, alors Filebeat collectera et analysera les logs de ce module.

Activons le module system, qui collecte et analyse les logs créés par le service de journalisation du système.

Linux 13

```
sudo filebeat modules enable system
```

Nous devons configurer les pipelines Filebeat ingest, qui analysent les données du

journal avant de les envoyer via logstash à Elasticsearch. Pour charger la canalisation d'ingest pour le module système :

Linux 14

```
sudo filebeat setup –pipelines –modules system
```

Chargons le gabarit d'index dans Elasticsearch :

Un indice de recherche Elasticsearch est un ensemble de documents qui ont des caractéristiques similaires. Les index sont identifiés par un nom, qui est utilisé pour désigner l'index lors de l'exécution de diverses opérations à l'intérieur de celui-ci. Le modèle d'index sera automatiquement appliqué lors de la création d'un nouvel index.

Linux 15

```
sudo filebeat setup –index-management -E output.logstash.enabled=false -E 'output.elasticsearch.hosts=["localhost :9200"]'
```

Filebeat est emballé avec des tableaux de bord Kibana d'échantillons qui nous permettent de visualiser les données Filebeat dans Kibana. Avant de pouvoir utiliser les tableaux de bord, nous devons créer le modèle d'index et charger les tableaux de bord en Kibana.

Lorsque les tableaux de bord se chargent, Filebeat se connecte à Elasticsearch pour vérifier les informations de version. Pour charger les tableaux de bord lorsque Logstash est activé, nous devons désactiver la sortie Logstash et activer la sortie Elasticsearch :

Linux 16

```
sudo filebeat setup -E output.logstash.enabled=false -E output.elasticsearch.hosts=['localhost :9200'] -E setup.kibana.host=localhost :5601
```

Nous pouvons maintenant démarrer Filebeat :

Linux 17

```
sudo systemctl start filebeat
sudo systemctl enable filebeat
```

5.2 Sur Kibana

Nous pouvons voir maintenant dans la section Discover que l'agent Filebeat est bien remonté.

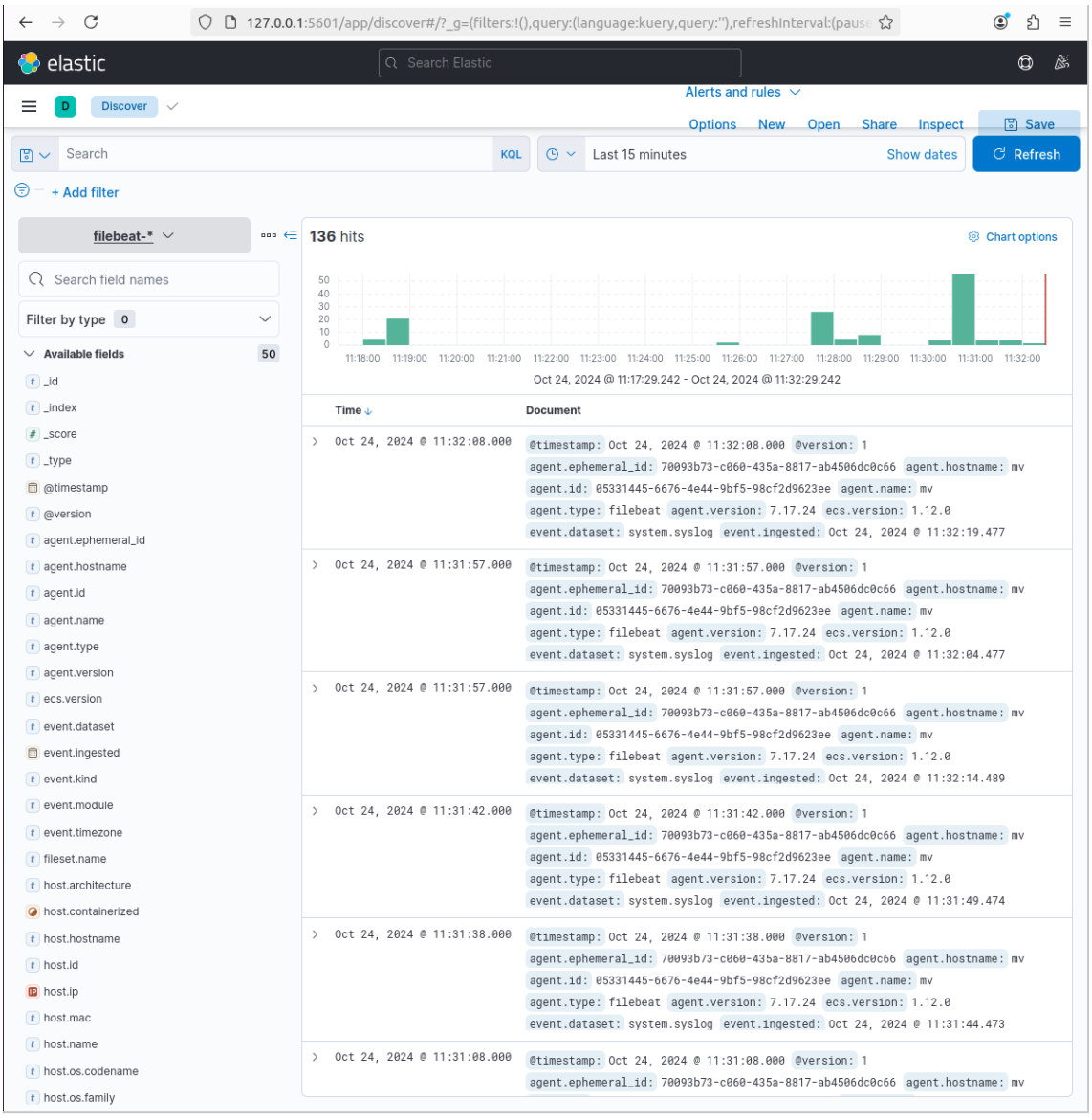

Figure 10 – *Discover*

5.3 Autres modules

Pour les autres modules (Metricbeat, Packetbeat, Winlogbeat, Auditbeat & Heartbeat), la procédure est exactement la même.

Il faut les installer sur la machine à surveiller, les configurer et les activer, tout remontra sur Elasticsearch et visualisable sur Kibana. Attention à bien renseigner l'adresse du serveur Elasticsearch car ici l'exemple a été faite sur la machine même (d'où localhost).

Université d'Artois

IUT de Béthune

Département Réseaux et Télécommunications

SAÉ5.CYBER03

Supervision d'un serveur web (Apache2) avec la suite ELK

par

Eliot HULEUX

Table des matières

1 Installation des dépendances d'Elasticsearch

Pour superviser la machine, nous aurons besoin de paquet de la suite ELK, nous devons donc ajouter le dépôt :

Linux 1

```
wget -qO - https ://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add
-
echo "deb https ://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee
/etc/apt/sources.list.d/elastic-8.x.list
```

2 Configuration d'Apache2

Pour superviseur le serveur web, nous nous servirons de son module *status* :

Linux 2

```
sudo a2enmod status
```

Configurons son accès afin de n'autoriser que la machine Elasticsearch (192.168.1.30) :

```
  GNU nano 6.2          /etc/apache2/mods-enabled/status.conf
<IfModule mod_status.c>

        <Location /server-status>
                SetHandler server-status
                Require local
                Require ip 192.168.1.30
        </Location>

        ExtendedStatus On

        <IfModule mod_proxy.c>
                ProxyStatus On
        </IfModule>

</IfModule>
```

Figure 1 – */etc/apache2/mods-enabled/status.conf*

3 Installation des modules Elasticsearch sur la machine Apache2

3.1 Ajout de Filebeat

Linux 3

sudo apt-get update

sudo apt-get install filebeat

Configurons le fichier */etc/filebeat/filebeat.yml* pour envoyer les logs à Elasticsearch :

```
  GNU nano 6.2                  /etc/filebeat/filebeat.yml *

# ---------------------------- Elasticsearch Output ----------------------------
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["http://192.168.1.30:9200"]
  preset: balanced
```

Figure 2 – */etc/filebeat/filebeat.yml*

Et maintenant, créons la liaison avec le module *status* d'Apache2 :

Linux 4

sudo filebeat modules enable apache

```
  GNU nano 6.2            /etc/filebeat/modules.d/apache.yml *
- module: apache
  access:
    enabled: true
    var.paths: ["/var/log/apache2/access.log*"]

  error:
    enabled: false
    var.paths: ["/var/log/apache2/error.log*"]
```

Figure 3 – */etc/filebeat/modules.d/apache.yml*

On peut ensuite tester la configuration pour voir si les fichiers sont correctement écrit et démarrer Filebeat :

Linux 5

sudo filebeat test config

sudo systemctl start filebeat

sudo systemctl enable filebeat

3.2 Ajout de Metricbeat

Linux 6

sudo apt-get update

sudo apt-get install metricbeat

Même principe que pour Filebeat, configurons le fichier */etc/metricbeat/metricbeat.yml* afin d'envoyer les métriques à Elasticsearch :

```
  GNU nano 6.2              /etc/metricbeat/metricbeat.yml *

# ---------------------------- Elasticsearch Output ----------------------------
output.elasticsearch:
  hosts: ["http://192.20.10.30:9200"]

  preset: balanced
```

Figure 4 – */etc/metricbeat/metricbeat.yml*

Même procédure, activons le module Apache2 pour Metricbeat :

Linux 7

sudo metricbeat modules enable apache

Maintenant, utilisons le module *status* d'Apache2 pour les rediriger vers Elasticsearch, afin de collecter les métriques du serveur web :

```
  GNU nano 6.2          /etc/metricbeat/modules.d/apache.yml *

- module: apache
  metricsets:
    - status
  period: 10s
  hosts: ["http://192.168.10.40/server-status"]
```

Figure 5 – */etc/metricbeat/mdules.d/apache.yml*

On peut tester la configuration pour voir si les fichiers sont correctement écrit et démarrer Metricbeat :

Linux 8

```
sudo metricbeat test config
sudo systemctl start metricbeat
sudo systemctl enable metricbeat
```

3.3 Ajout de Heartbeat

Linux 9

```
sudo apt-get update
sudo apt-get install heartbeat-elastic
```

Heartbeat permet de surveiller la disponibilité des services (HTTP, TCP, ICMP).

Configurons le fichier */etc/heartbeat/hearbeat.yml* afin d'envoyer les disponibilités à Elasticsearch :

```
  GNU nano 6.2              /etc/heartbeat/heartbeat.yml *
# Configure monitors inline
heartbeat.monitors:
- type: http
  enabled: true
  id: apache-status
  name: apache-status
  urls: ["http://172.20.10.2:9200"]
  schedule: '@every 10s'
  timeout: 5s

- type: tcp
  id: elasticsearch
  name: Elasticsearch TCP
  hosts: ["172.20.10.2:9200]
  schedule: '@every 10s'
```

Figure 6 – */etc/heartbeat/hearbeat.yml*

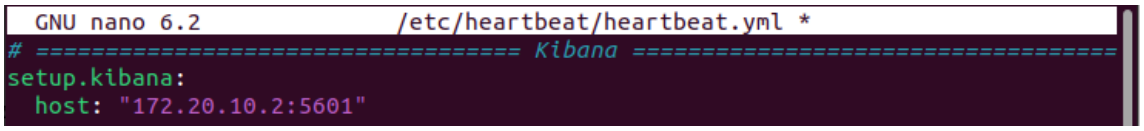

Figure 7 – */etc/heartbeat/hearbeat.yml*

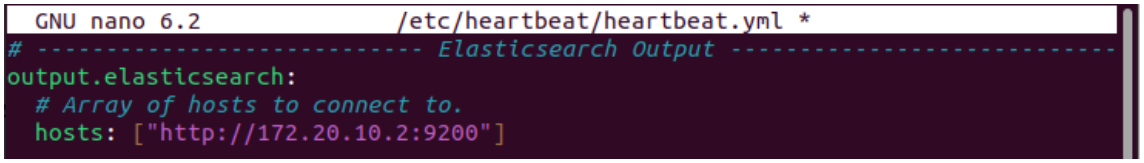

Figure 8 – */etc/heartbeat/hearbeat.yml*

On peut tester la configuration pour voir si les fichiers sont correctement écrit et démarrer Heartbeat :

Linux 10

```
sudo hearbeat-elastic test config
sudo systemctl start heartbeat-elastic
sudo systemctl enable heartbeat-elastic
```

3.4 Création des dashboards sur Kibana

Enfin, lançons Filebeat et Metricbeat pour créer les dashboards prédéfinis sur Kibana :

Linux 11

```
sudo filebeat setup
sudo metricbeat setup
sudo heartbeat setup
```

Sur Kibana, nous pouvons voir les dashboards prédéfinis de Filebeat et Filebeat avec des valeurs par défaut modifables selon les besoins du monitoring, ainsi qu'un tableau continue des ressources :

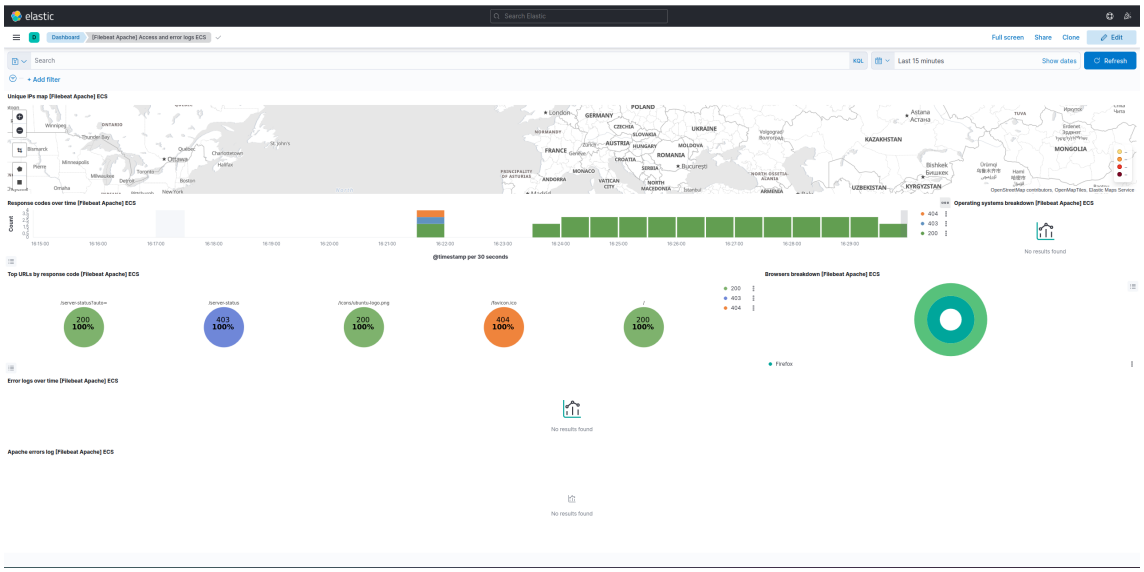

Figure 9 – *Dashboard Filebeat*

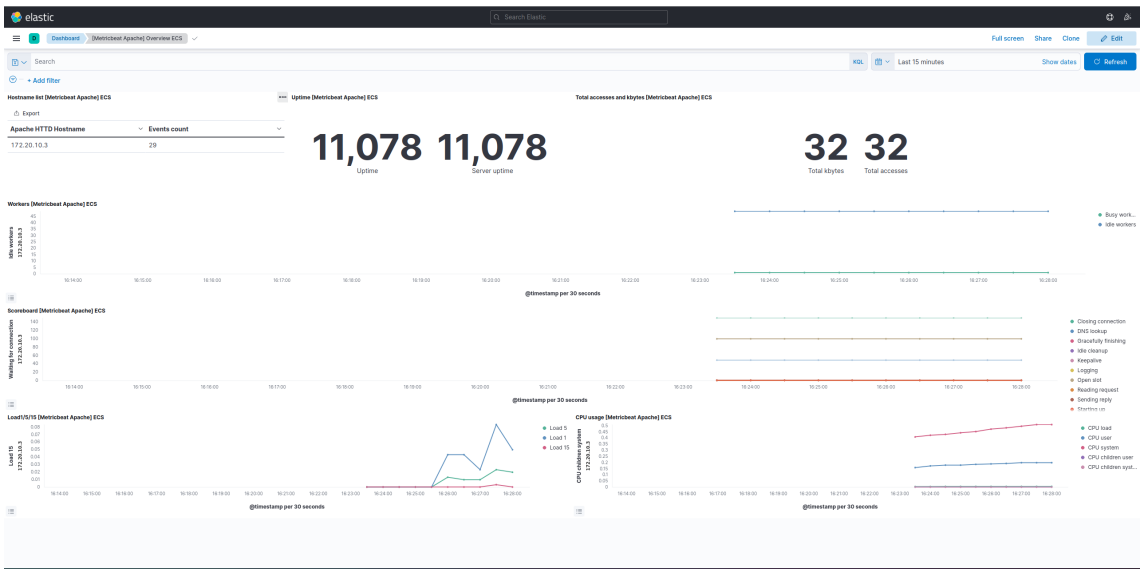

Figure 10 – *Dashboard Metricbeat*

Figure 11 – *Dashboard des ressources*

Université d'Artois

IUT de Béthune

Département Réseaux et Télécommunications

SAÉ5.CYBER03

Supervision d'une machine Ubuntu 22.04.5

par

Eliot HULEUX

Table des matières

1 Paramétrage de l'agent Elastic

1.1 Différences avec les Beats

Ici, nous utiliserons l'agent Elasticsearch, plutôt que d'installer directement les modules Beat d'Elastic.

Beats :

- Les Beats sont une collection d'agents légers conçus pour collecter et envoyer des données spécifiques vers Elasticsearch ou Logstash ;
- Chaque Beat a un objectif spécifique :
  - Filebeat : Collecte les logs
  - Metricbeat : Collecte des métriques système ou applicatives
  - Packetbeat : Analyse le trafic réseau
  - Heartbeat : Surveille la disponibilité des services
  - Auditbeat : Collecte les événements d'audit système
  - Winlogbeat : Collecte les logs d'événements Windows
- Chaque Beat est autonome et gère un type spécifique de données ;
- Les Beats sont configurés individuellement avec leurs propres fichiers YAML ;
- La gestion des Beats sur plusieurs machines peut devenir complexe dans des environnements à grande échelle ;
- Nécessite des efforts supplémentaires pour l'intégration avec d'autres composants Elastic Stack ;
- Peuvent nécessiter des outils externes pour des fonctionnalités supplémentaires (comme la sécurité ou la prévention des menaces) ;

Elastic Agent :

- Introduit pour simplifier la gestion des Beats et centraliser la collecte de données ;
- Il peut remplacer plusieurs Beats (Filebeat, Metricbeat, etc.) avec une seule installation ;
- Fonctionne avec une architecture unifiée : un seul agent peut gérer plusieurs types de données via des intégrations (comme les modules prédéfinis pour les logs, métriques, etc.) ;

— Intègre également des fonctionnalités de sécurité comme la collecte des données Endpoint et la prévention des menaces ;
— Permet une gestion centralisée via Fleet, une interface intégrée dans Kibana ;
— Les politiques de collecte de données et les intégrations sont gérées directement depuis Kibana, ce qui simplifie la configuration ;
— Idéal pour les environnements à grande échelle, où une gestion centralisée est essentielle ;
— Offre des fonctionnalités supplémentaires comme l'Endpoint Security (protection contre les malwares, détection des menaces) ;

Elastic privilégie désormais Elastic Agent pour les nouveaux projets, car il représente l'évolution et la simplification de la collecte de données.

1.2 Installation de l'agent

Pour l'installer, rendons nous dans les intégrations et rechercher l'intégration *System* :

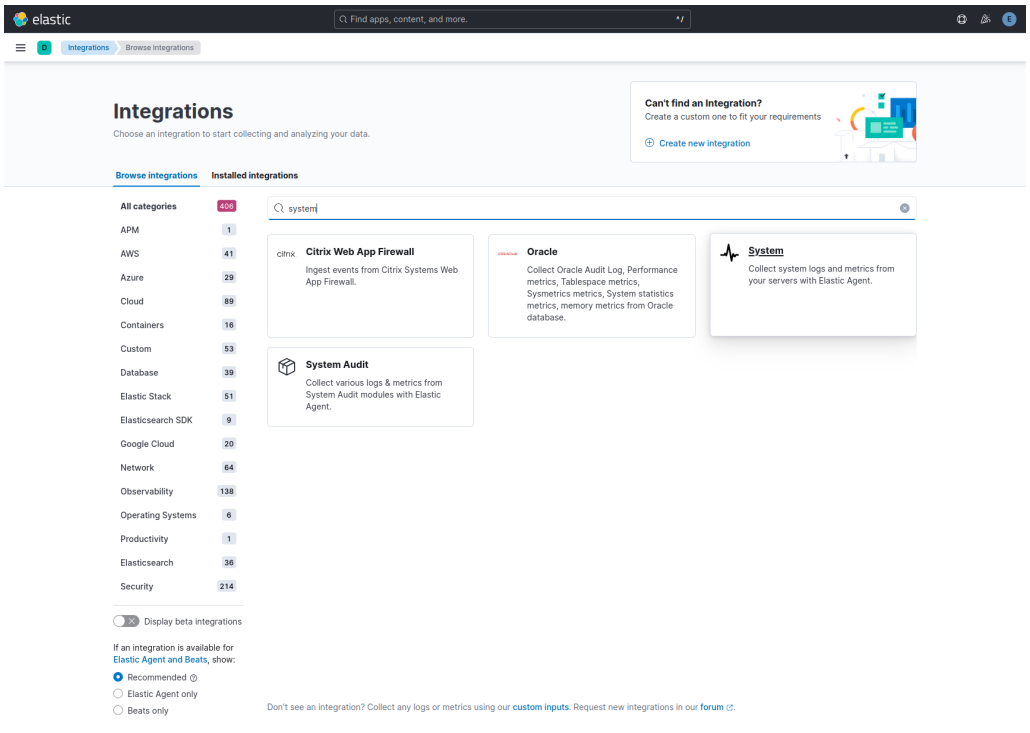

Figure 1 – *Intégration de System prt. 1*

Configurons les choix de journaux à collecter. Pour cette démo, nous sélectionnerons tous les journaux sauf Windows, étant donné que nous souhaitons superviser un système Linux.

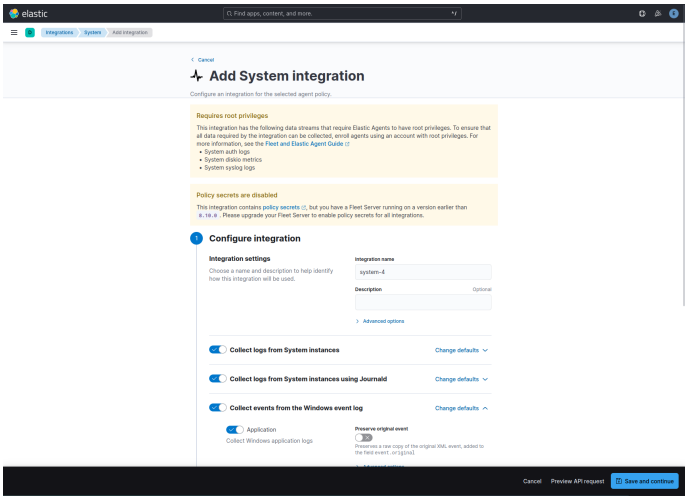

Figure 2 – *Intégration de System prt. 2*

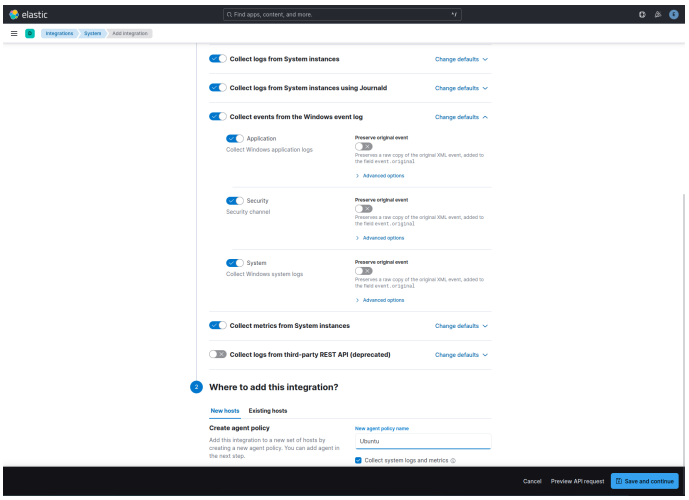

Figure 3 – *Intégration de System prt. 3*

Une fois coché, validons, et l'intégration System est configurée.

Maintenant, il faut ajouter un agent. Pour cela, cliquons sur *Add Elastic Agent to your hosts*, puis *Add Agent*, depuis la section *Integration policies* du module *System*.

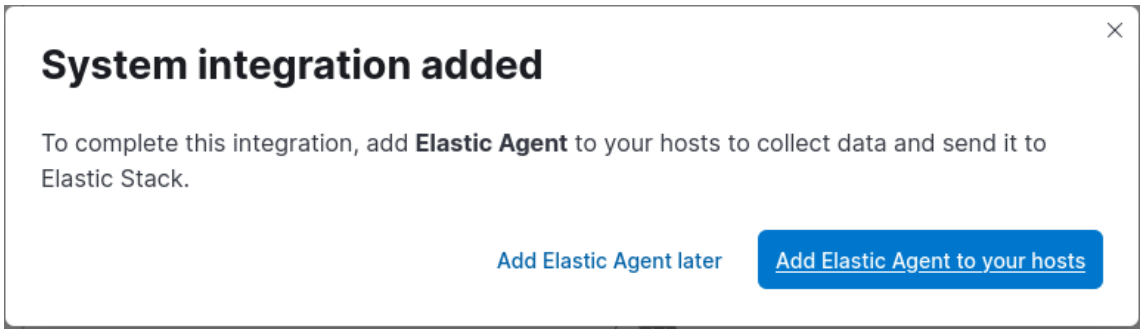

Figure 4 – *Intégration de System prt. 4*

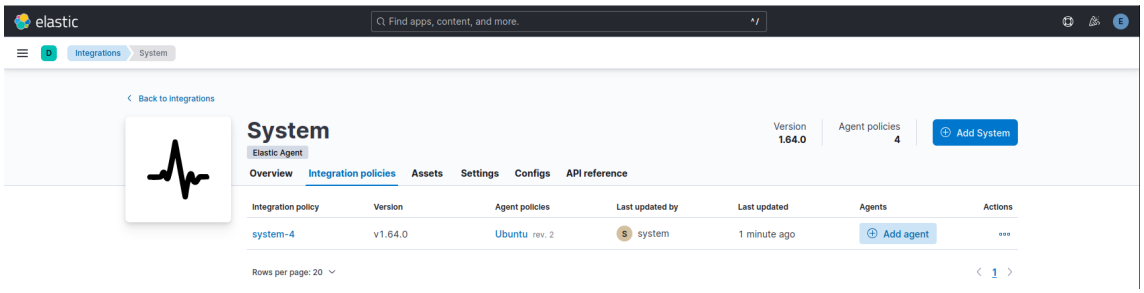

Figure 5 – *Intégration de System prt. 5*

Une fenêtre apparaîtra sur la droite avec les indications et le contenu du fichier */etc/elastic-agent/elastic-agent.yml*, il faudra le copier-coller et le garder en mémoire pour le coller dans le fichier au moment venu.

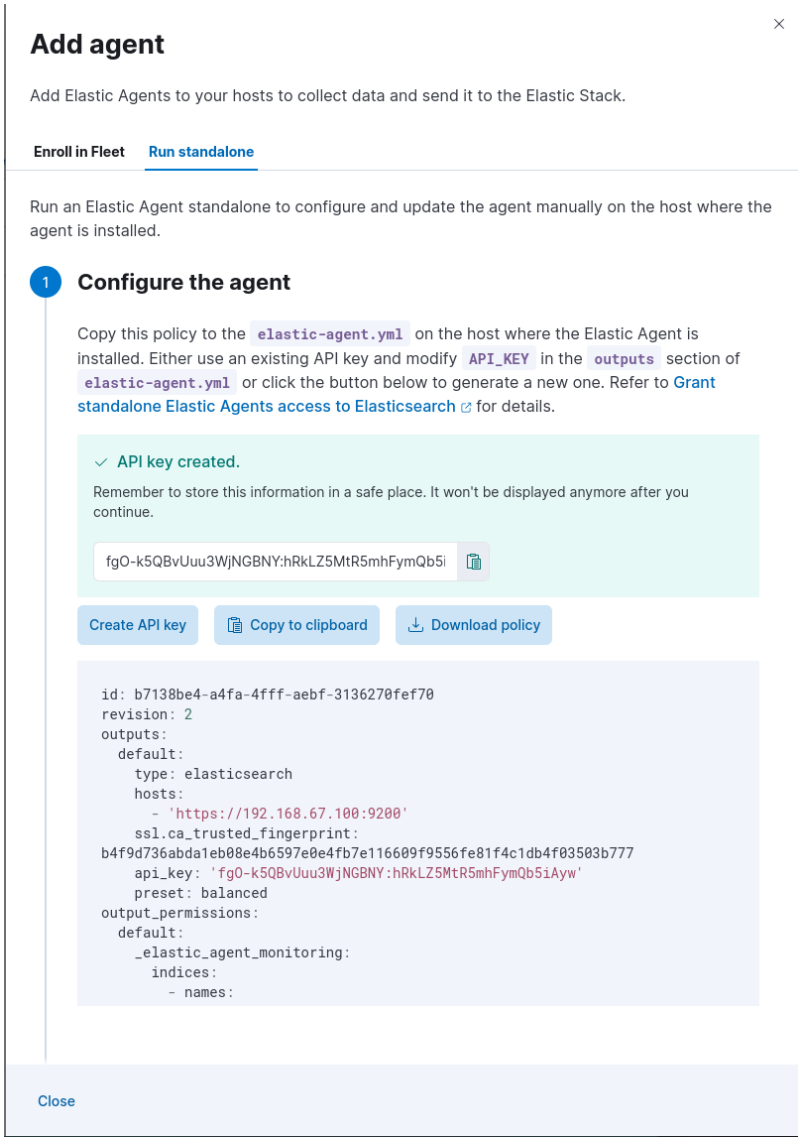

Figure 6 – *Ajouter un agent prt. 1*

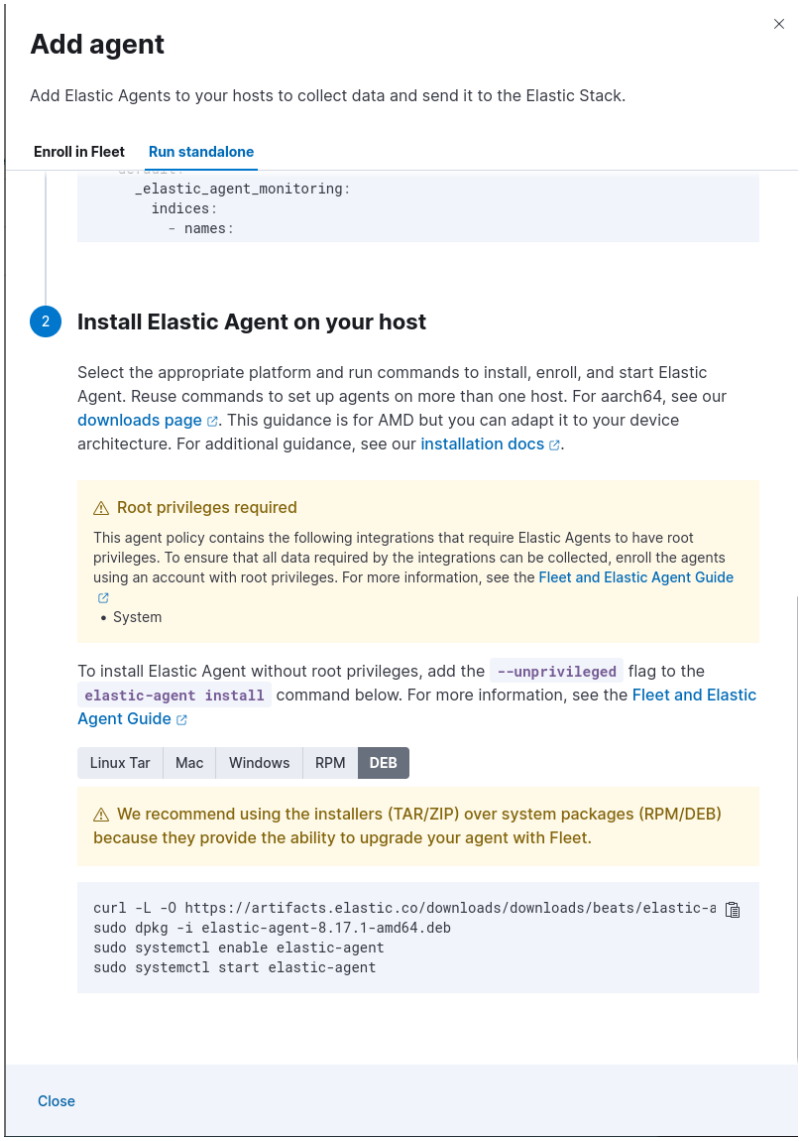

Figure 7 – *Ajouter un agent prt. 2*

Nous pouvons constater que Kibana rend l'agent Elastic simple d'utilisation avec le contenu du fichier directement créer.

Sur la machine Linux, à superviser, installons l'agent depuis le dépôt d'Elastic :

Linux 1

```
curl -L -O https ://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-8.17.1-amd64.deb
dpkg -i elastic-agent-8.17.1-amd64.deb
systemctl start elastic-agent
systemctl enable elastic-agent
```

Modifions la configuration de l'agent avec le texte que nous pouvions copier-coller depuis Kibana lors de l'ajout d'agent (ci-dessous figure le début du fichier, il faut copier-coller dans le fichier son intégralité) :

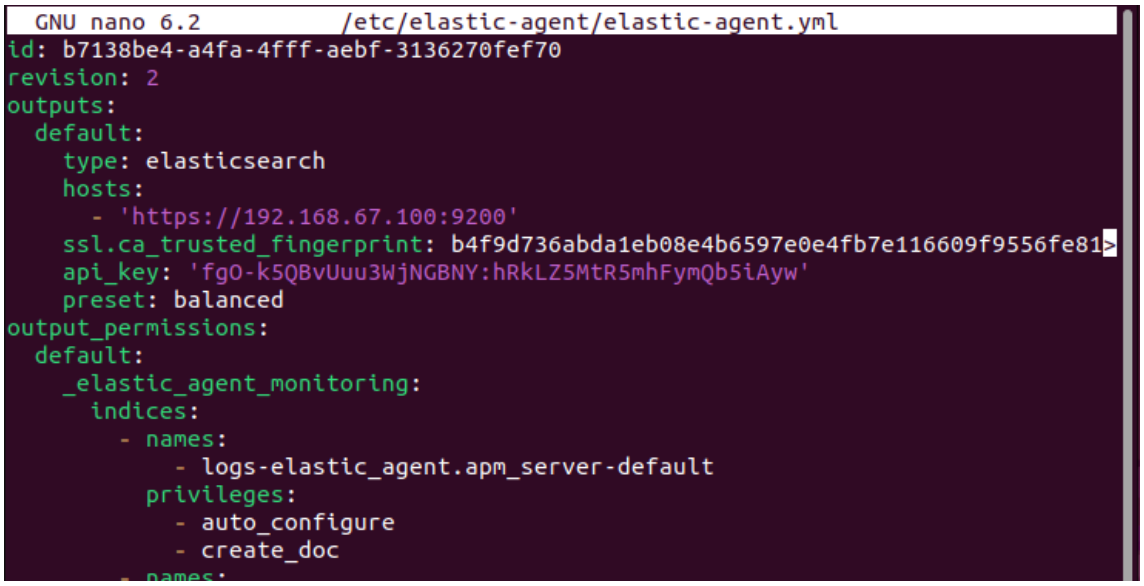

Figure 8 – */etc/elastic-agent/elastic-agent.yml*

Redémarrons le service et regardons sur Kibana si des données sont collectées :

Linux 2

```
systemctl restart elastic-agent.service
```

2 Résultats

Sur Kibana, nous pouvons voir que dans la section *Discover*, par exemple, les métriques apparaissent :

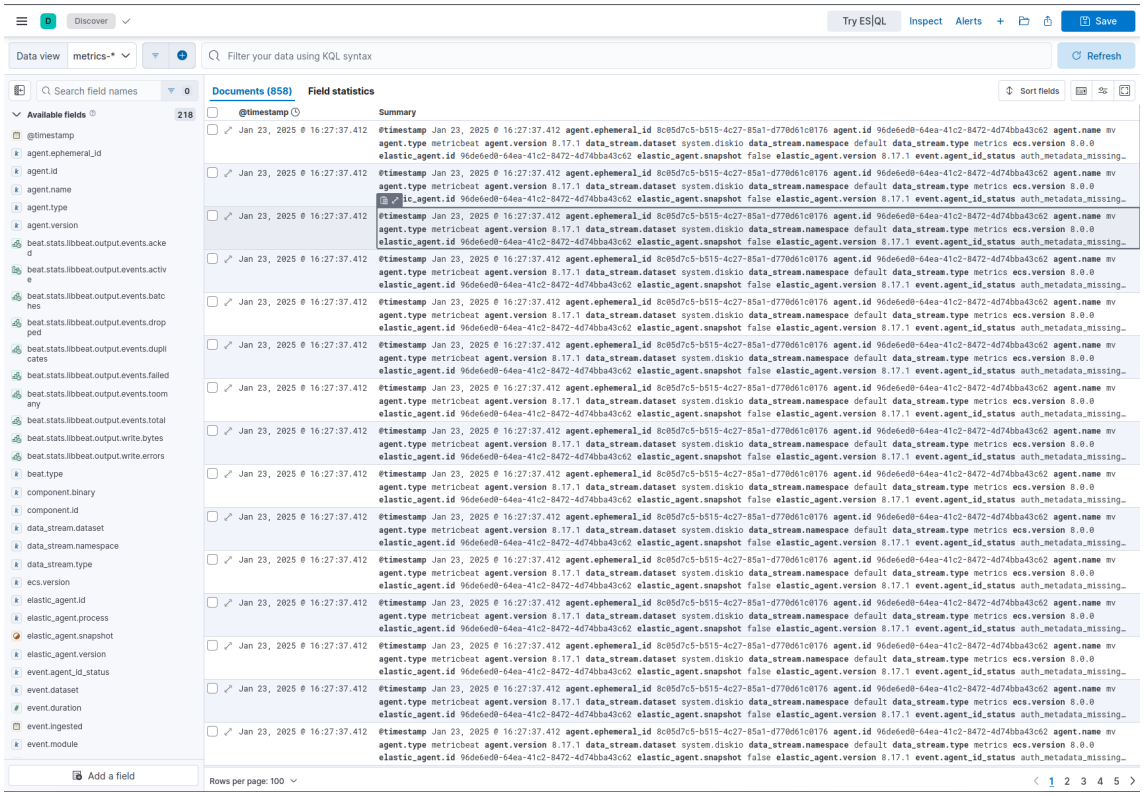

Figure 9 – *Métriques de la machine Linux*

Ainsi que les logs par exemple :

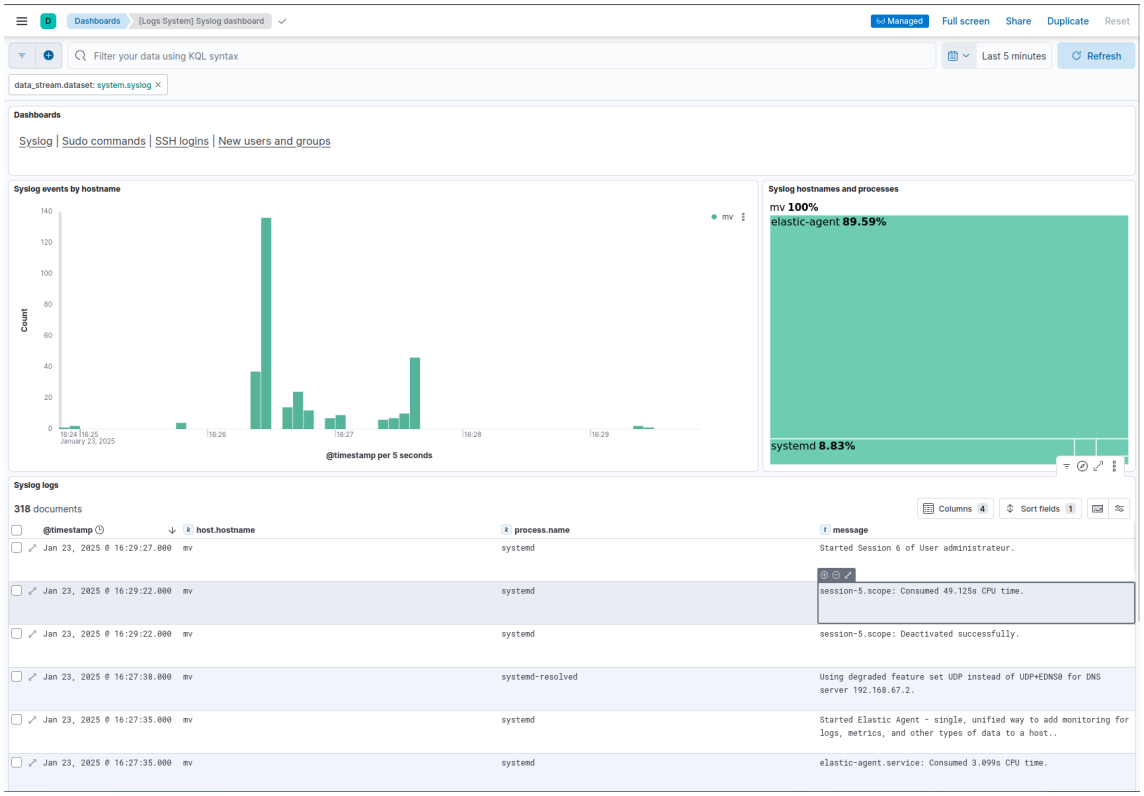

Figure 10 – *Logs de la machine Linux*

Kibana fournit également des dashboards des ressources en réel de la machine supervisé :

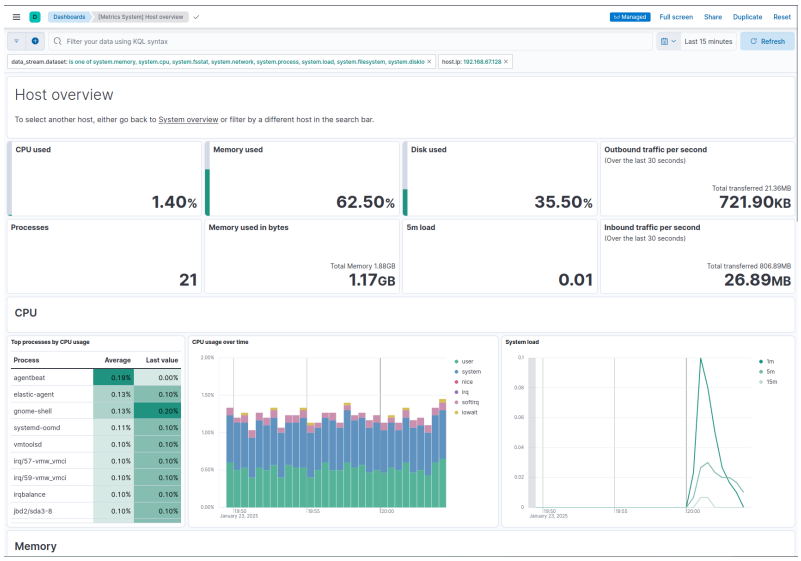

Figure 11 – *Dashboard des ressources de la machine Linux*

Ainsi que les connections SSH par exemple :

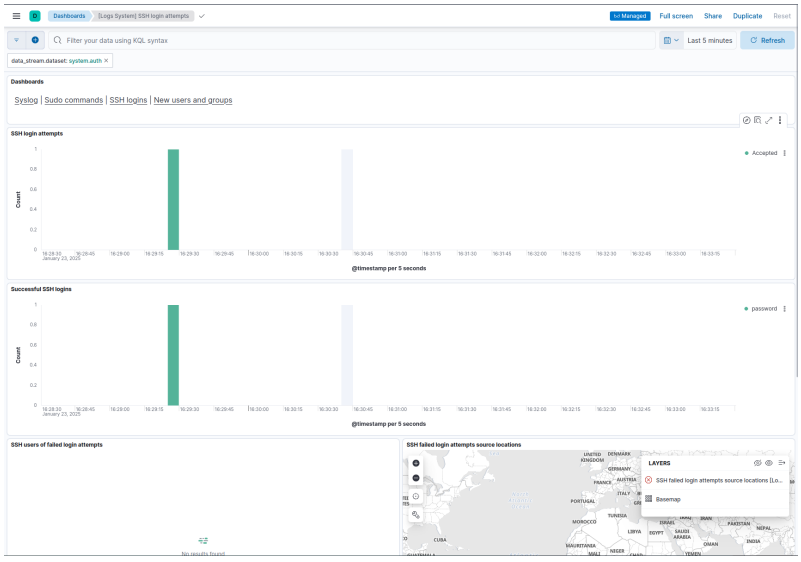

Figure 12 – *Dashboard des connections SSH de la machine Linux*

Université d'Artois

IUT de Béthune

Département Réseaux et Télécommunications

SAÉ5.CYBER03

Supervision d'un équipement Cisco

par

Eliot HULEUX

Table des matières

1 Configuration du routeur Cisco

1.1 Détails du réseau

Pour cette supervision, nous utiliserons GNS3 avec un routeur Cisco 3660, qui sera relié à la carte réseau *vmnet8*, qui permettra d'accéder au sous-réseau *192.168.67.0/24*, où est situé la machine ELK.

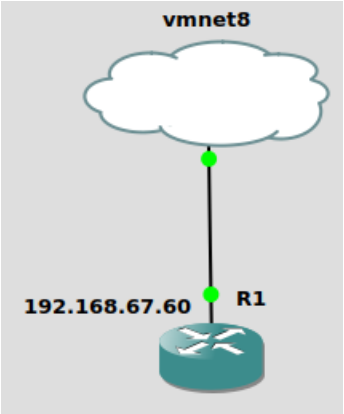

Figure 1 – *Schéma réseau GNS3*

Le routeur est relié au sous-réseau par son interface *FastEthernet0/0*, avec une adresse IPv4 *192.168.67.60/24*

1.2 Configuration partie routeur

Commençons par configurer SNMP sur le routeur, qui permettra de collecter les métriques du routeur :

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#snmp-server community public RO
R1(config)#snmp-server host 192.168.67.100 version 2c public
```

Figure 2 – *Commandes Cisco pour le protocole SNMP*

— *public* : communauté SNMP
— *RO* signifie *Read-Only*
— *192.168.67.100* étant l'adresse IP de la machine ELK

Ensuite, configurons *syslog* pour récupérer les logs système du routeur :

```
R1(config)#logging host 192.168.67.100 transport udp port 9002
R1(config)#logging trap informational
R1(config)#logging origin-id hostname
R1(config)#logging facility local5
```

Figure 3 – *Commandes Cisco pour le serveur syslog*

— *192.168.67.100* étant l'adresse IP de la machine ELK
— *informational* : niveau de détail des logs

Et pour finir, configurons la surveillance du trafic réseau, avec NetFlow :

```
R1(config)#interface FastEthernet0/0
R1(config-if)#ip flow ingress
R1(config-if)#ip flow egress
R1(config-if)#exit
R1(config)#ip flow-export destination 192.168.67.100 9995
R1(config)#ip flow-export version 5
```

Figure 4 – *Commandes Cisco pour NetFlow*

1.3 Configuration côté ELK

Maintenant, il faut configurer un agent de collecte, pour cela, nous utiliserons filebeat :

Linux 1

```
apt-get install filebeat
```

Filebeat possède un module Cisco, utilisons le pour cette supervision :

Linux 2

```
sudo filebeat modules enable cisco
```

Et configurons le fichier de configuration du module :

```
  GNU nano 6.2              /etc/filebeat/modules.d/cisco.yml *
  ios:
    enabled: true

    var.input: syslog
    var.syslog_host: 192.168.67.100
    var.syslog_port: 9002
    var.syslog_protocol: udp
```

Figure 5 – */etc/filebeat/modules.d/cisco.yml*

Ainsi que la sortie de Filebeat :

```
  GNU nano 6.2                /etc/filebeat/filebeat.yml *
setup.kibana:

  host: "192.168.67.100:5601"

# ------------------------------ Elasticsearch Output ------------------------------
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["https://192.168.67.100:9200"]

  # Performance preset - one of "balanced", "throughput", "scale",
  # "latency", or "custom".
  preset: balanced

  ssl.certificate_authorities: ["/etc/filebeat/http_ca.crt"]
  username: elastic
  password: RFD5Wbqa5HZZa_l4sPfq
```

Figure 6 – */etc/filebeat/filebeat.yml*

Et il n'y a plus qu'à démarrer le service :

Linux 3

systemctl start filebeat.service

systemctl enable filebeat.service

systemctl filebeat setup

Sur le routeur par exemple, nous avons effectué un *no shutdown* sur l'interface *FastEthernet1/0* et *FastEthernet2/0*, et nous collectons bien l'événement sur Kibana.

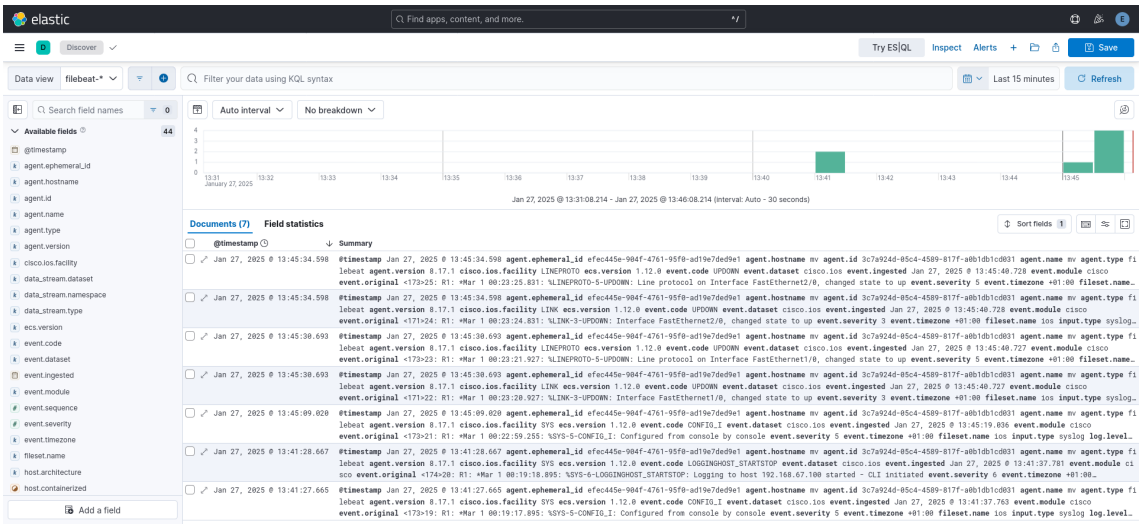

Figure 7 – *Syslog du routeur Cisco*

Université d'Artois

IUT de Béthune

Département Réseaux et Télécommunications

SAÉ5.CYBER03

Supervision d'un Windows Server 2025

par

Eliot HULEUX

Table des matières

1 Explication de l'environnement

Dans cette supervision, nous utiliserons une machine Windows Server 2025 Datacenter Edition, dans laquelle nous avons installé les services de domaine Active Directory (domaine : *elastic.tp* et DNS. Nous utiliserons l'agent Filebeat qui collectera les journaux d'audits, de sécurité, d'installation, système et application Windows.

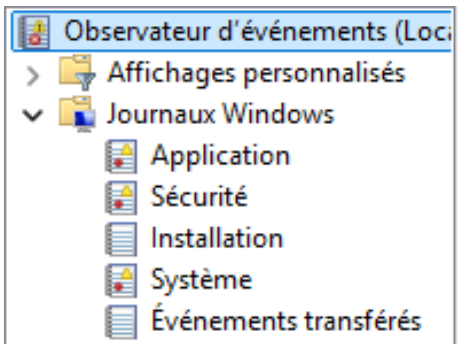

Figure 1 – *Journaux Windows*

2 Installation de l'agent

2.1 Côté Elasticsearch et Kibana

La différence entre les Beats et l'agent est expliqué dans le TP de supervision d'une machine Ubuntu.

La suite ELK possède une intégration nommée *Windows* qui permettra de collecter les journaux de la machine.

Pour cela, ajoutons l'intégration, et formulons une police d'agent :

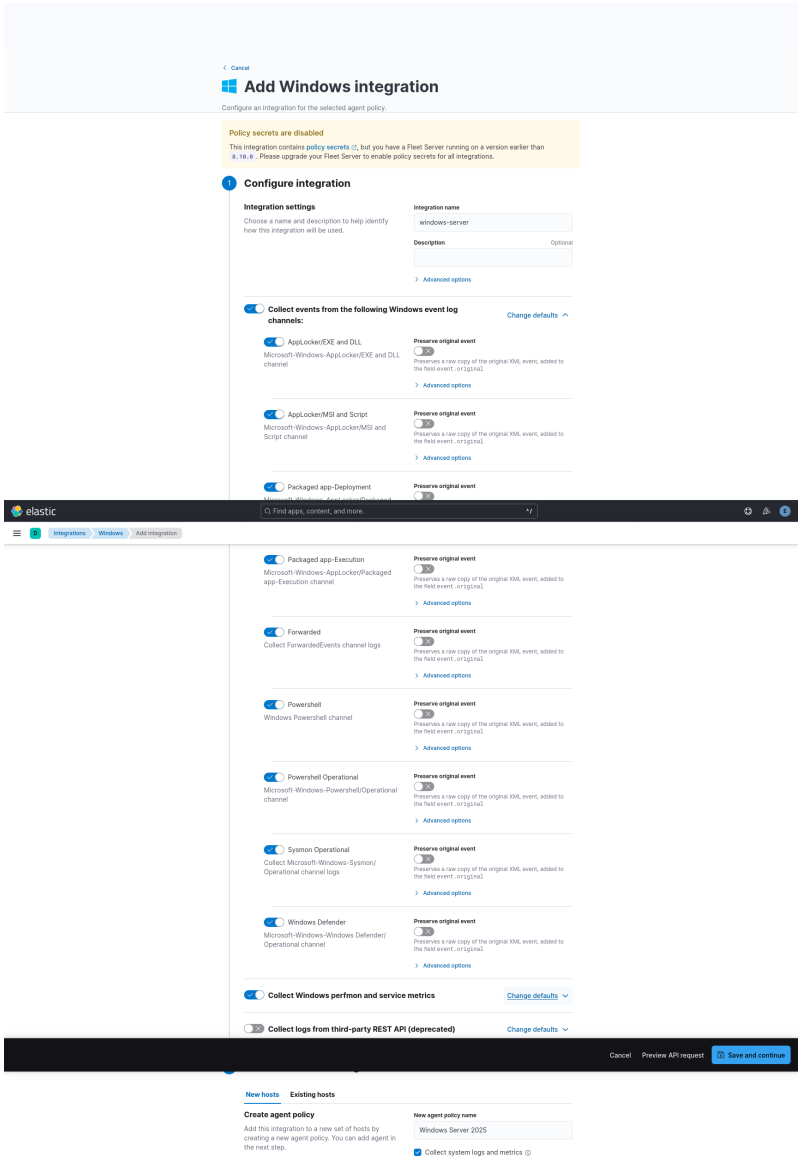

Figure 2 – *Intégration du module Windows prt. 1*

Ensuite il faut ajouter créer le fichier pour l'agent, ajoutons donc un nouvel agent :

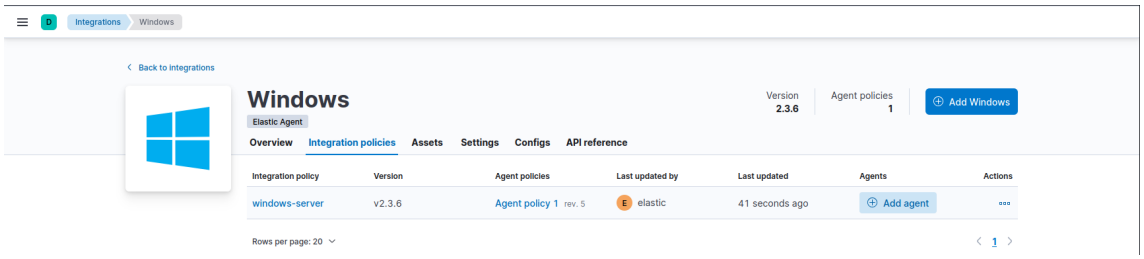

Figure 3 – *Intégration du module Windows prt. 2*

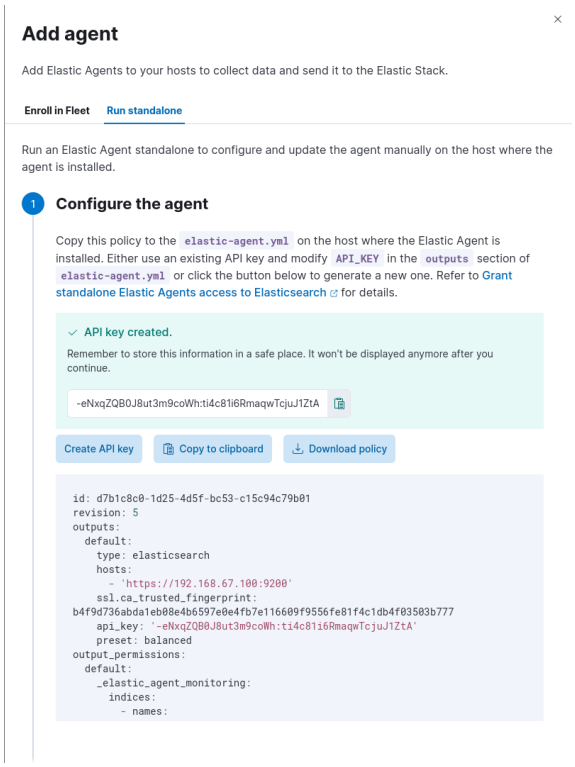

Figure 4 – *Intégration du module Windows prt. 2*

Gardons en mémoire ce fichier, en le copiant dans le presse papier. Nous le collerons dans le fichier *elastic-agent.yml* sur la machine Windows.

2.2 Côté Windows Server 2025

Pour installer l'agent, nous devons télécharger l'agent disponible avec ce lien :

https ://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-8.17.1-windows-x86_64.zip

Ensuite, après l'avoir désarchiver, déplaçons nous dans le dossier et modifions le fichier *elastic-agent.yml* avec ce que nous devions copier-coller précédemment.

Ensuite, exécutons le fichier .exe pour installer l'agent.

Powershell 1

```
./elastic-agent.exe install
```

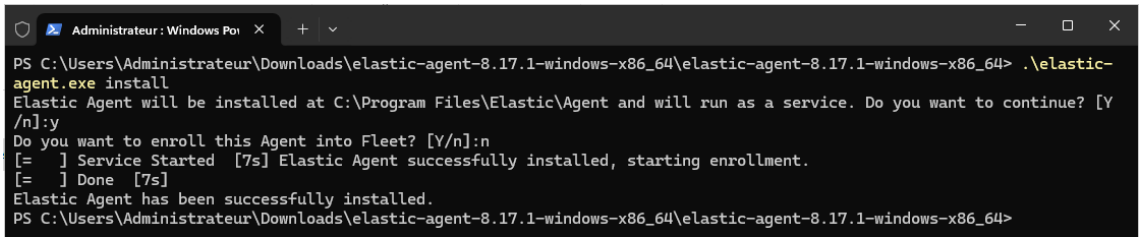

Figure 5 – *Intégration du module Windows*

3 Résultats sur Kibana

Sur Kibana, nous pouvons que les journaux de logs remontent par exemple :

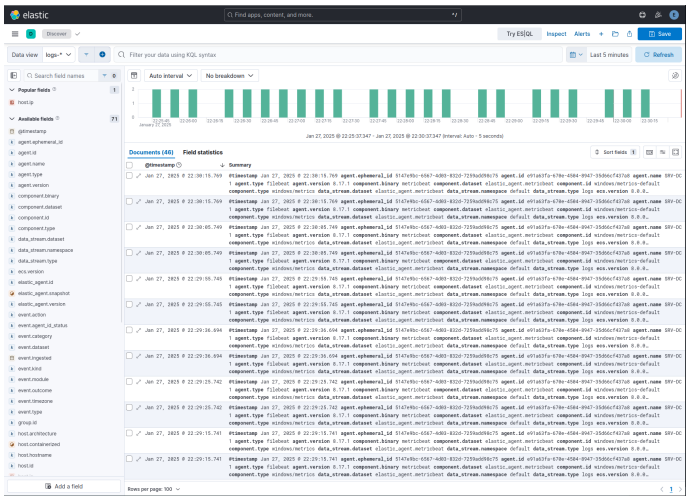

Figure 6 – *Journaux de logs Windows*

Sur les dashboards disponibles, on peut voir un dashboard qui remontent les ID des journaux :

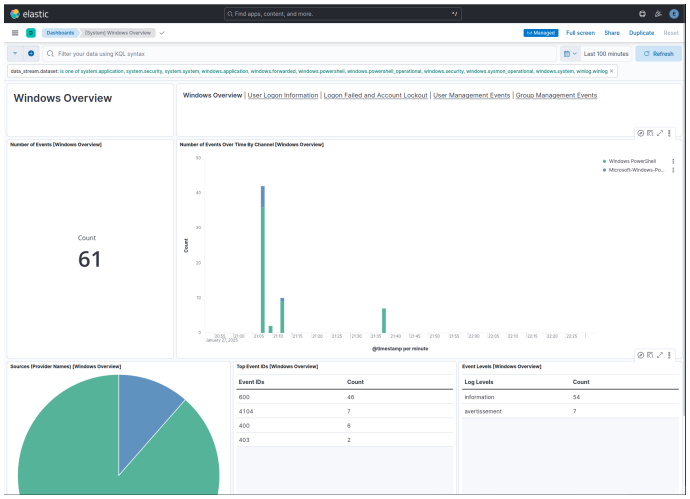

Figure 7 – *Dashboard de logs Windows*