

# COMPRENDRE ET UTILISER PACKETBEAT DANS LA STACK ELK

## Introduction

---

Nous sommes maintenant confrontés à des entreprises avec des réseaux de plus en plus complexes qui doivent être protégés. De nos jours, il est absolument nécessaire de surveiller votre réseau, chaque organisation doit **collecter toutes les informations sur ses réseaux**. Si vous ne surveillez pas votre réseau, comment pourrez-vous détecter ce qui est un comportement normal ou une attaque ? Il existe des attaques malveillantes évidentes, mais une détection plus précoce est cruciale dans tout incident de cybersécurité. L'un des moyens les meilleurs et les plus pratiques de détecter une activité anormale est **la surveillance du réseau**. Dans cet article, nous allons voir comment est-il possible de surveiller votre réseau avec l'outil Packetbeat, puis d'agréger toutes les informations dans la pile ELK.

Dans le [chapitre précédent sur Metricbeat](#) , nous avons décrit comment utiliser Metricbeat pour envoyer des métriques systèmes dans la pile. Dans ce chapitre, nous allons **apprendre à utiliser Packetbeat** qui fait partie de la famille d'expéditeurs Beats les plus populaires.

## Qu'est-ce que Packetbeat ?

Packetbeat est un outil de surveillance réseau développé par Elastic qui utilise la bibliothèque [libpcap](#), c'est un expéditeur et un analyseur de données open source pour les paquets réseaux intégrés à la pile ELK (Elasticsearch, Logstash et Kibana) et membre de la famille des expéditeurs Beats (Filebeat, Libbeat, Winlogbeat, etc...). Il

fournit des métriques de surveillance réseau en temps réel sur du protocole HTTP, TLS, DNS et de nombreux autres protocoles réseaux.

Ainsi grâce à cet outil de surveillance de paquets de données avec la stack ELK, il peut vous aider à détecter des niveaux inhabituels de trafic réseau et des caractéristiques de paquets inhabituelles, à identifier les sources et les destinations des paquets, à rechercher des chaînes de données spécifiques dans les paquets et à créer un tableau de bord convivial avec des statistiques pertinentes prêtes à l'emploi. Ainsi cela peut contribuer à améliorer vos temps de réponse aux attaques malveillantes.

Dans cet article, nous allons démontrer la plupart des avantages mentionnés ci-dessus. Plus précisément, nous allons **apprendre à utiliser Packetbeat pour surveiller les transactions HTTP d'une application Web** et d'analyser les données en utilisant la stack ELK.

## Installation et configuration de Packetbeat

### Préparation de l'application web

Afin d'éviter d'installer de nombreux paquets et bibliothèques, nous allons déployer notre application web depuis la technologie Docker. Tout D'abord commencez par télécharger les sources du projet en cliquant [ici](#) et désarchivez ensuite le projet.

Ensuite, je vais supposer que vous disposez déjà d'un environnement Docker fonctionnel sur votre système, si ce n'est pas le cas merci de suivre mon [article sur l'installation de docker](#).

Une fois cette étape réalisée, placez-vous alors à la racine du projet et buildez ensuite votre image avec la commande suivante :

```
docker build -t myapp .
```

Ensuite toujours depuis la racine du projet, lancez la commande suivante pour exécuter vos conteneurs :

```
docker-compose up -d
```

On s'assure ensuite que nos conteneurs s'exécutent correctement sur les bons ports:

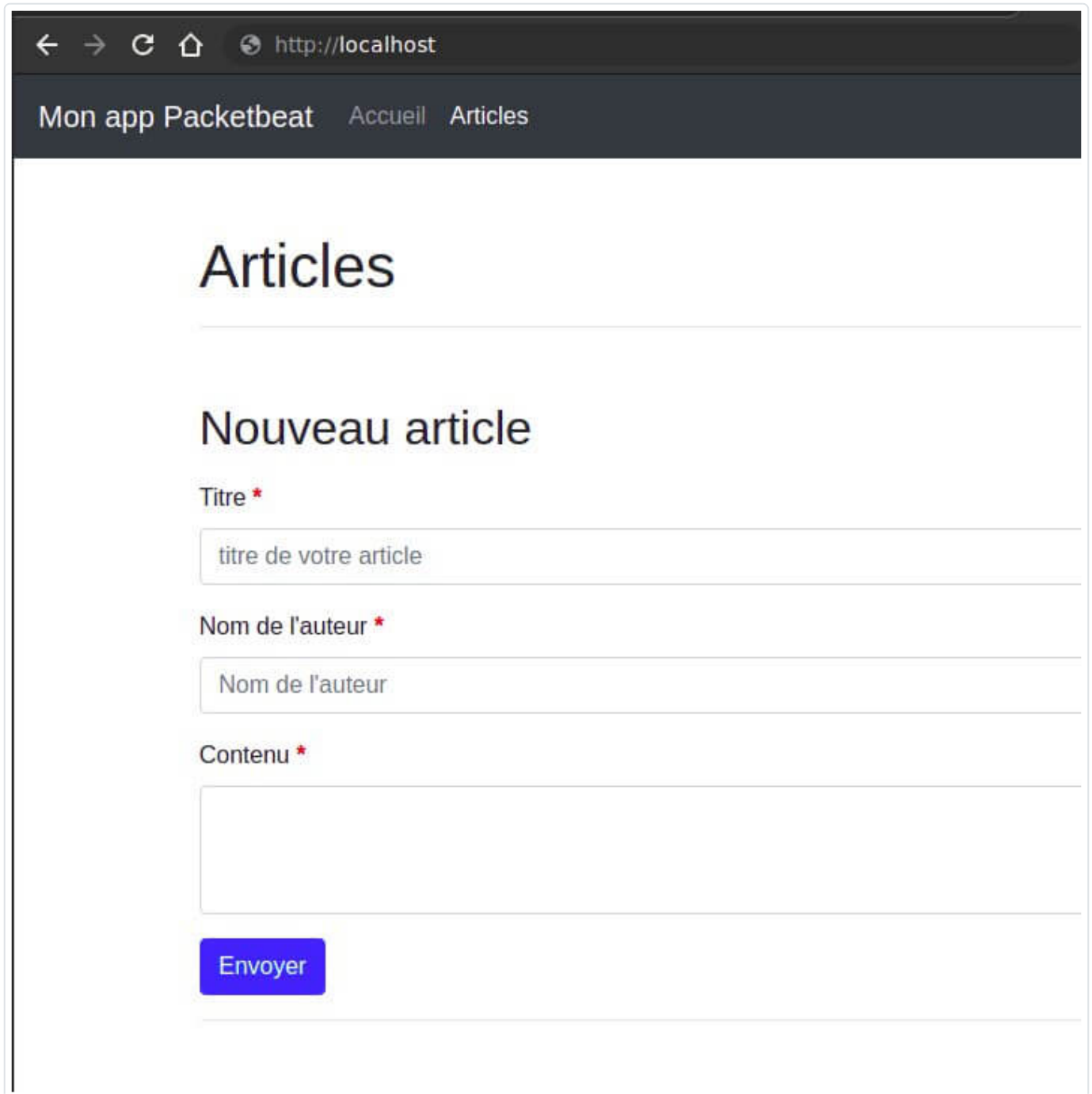
```
docker ps
```

### Résultat :

| CONTAINER ID | IMAGE     | COMMAND                  | CREATED        | STATUS        |
|--------------|-----------|--------------------------|----------------|---------------|
| 97e0bb711609 | myapp     | "docker-php-entrypoi..." | 12 minutes ago | Up 12 minutes |
| feae30be6007 | mysql:5.7 | "docker-entrypoint.s..." | 12 minutes ago | Up 12 minutes |

Ici nous avons un conteneur contenant un serveur web nommé "myapp\_c" écoutant sur le port 80 et un autre conteneur mysql écoutant sur le port 3306.

Si tout c'est bien passé, alors visitez la page suivante <http://localhost>, et vous obtiendrez le résultat suivant :



The screenshot shows a web browser window with the address bar displaying 'http://localhost'. The page has a dark header with the text 'Mon app Packetbeat' and two navigation links, 'Accueil' and 'Articles'. The main content area has a large heading 'Articles' followed by a horizontal line. Below this is the heading 'Nouveau article'. There are three form fields: 'Titre \*' with a placeholder 'titre de votre article', 'Nom de l'auteur \*' with a placeholder 'Nom de l'auteur', and 'Contenu \*' which is a larger text area. A blue button labeled 'Envoyer' is positioned below the 'Contenu' field.

← → ↻ 🏠 🌐 http://localhost

Mon app Packetbeat Accueil Articles

# Articles

---

## Nouveau article

Titre \*

Nom de l'auteur \*

Contenu \*

Envoyer

## Installation de Packetbeat

Avant de commencer l'installation, assurez-vous d'avoir installé Elasticsearch pour stocker et rechercher nos données, et d'avoir installé Kibana pour les visualiser et les gérer (mon tuto d'installation est disponible [ici](#)).

Comme pour Metricbeat, il existe plusieurs façons d'installer Packetbeat. Dans notre cas nous allons comme pour, soit **installer Packetbeat depuis le gestionnaire de paquets par défaut** de notre distribution. Pour ce faire, vous devrez d'abord mettre à niveau votre système et vos paquets:

```
sudo apt update -y && sudo apt upgrade -y
```

Pour les machines appartenant à la famille debian, vous devrez peut-être installer le paquet `apt-transport-https` avant de continuer:

```
sudo apt-get install apt-transport-https
```

Téléchargez et installez ensuite la clé de signature publique (*étape non obligatoire, si vous avez suivie le chapitre précédent*) :

Sous la famille debian:

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

Sous la famille redhat:

```
sudo rpm --import https://packages.elastic.co/GPG-KEY-elasticsearch
```

L'étape suivante consiste à ajouter le dépôt Elastic sur votre système (*étape non obligatoire, si vous avez suivie le chapitre précédent*) :

Sous la famille debian:

```
echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic.list
```

Sous la famille redhat, créez un fichier et nommez le par exemple `elastic.repo` dans le répertoire `/etc/yum.repos.d/`, contenant:

```
[elastic-7.x]
name=Elastic repository for 7.x packages
baseurl=https://artifacts.elastic.co/packages/7.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
type=rpm-md
```

Il ne vous reste plus qu'à mettre à jour vos référentiels et d'installer Packetbeat:

Sous la famille debian:

```
sudo apt-get update && sudo apt-get install packetbeat
```

Sous la famille redhat:

```
sudo yum install packetbeat
```

Pour exécuter Packetbeat, utilisez la commande suivante:

```
sudo systemctl start packetbeat
```

Si jamais vous rencontrez des problèmes d'initialisation, veuillez vérifier les logs du service Packetbeat à l'aide de la commande suivante :

```
sudo journalctl -f -u packetbeat
```

## Configuration de Packetbeat

Le fichier de configuration de Packetbeat se retrouve dans **`/etc/packetbeat/packetbeat.yml`** . Dans ce fichier, assurez-vous bien que la configuration Packetbeat possède les bonnes informations pour communiquer avec Kibana et Elasticsearch, si besoin décommentez les lignes suivantes :

```

packetbeat.interfaces.device: any

...

output.elasticsearch:
  hosts: ["localhost:9200"]
  username: "" #(si pas de login/mot de passe ne rien mettre)
  password: "" #(si pas de login/mot de passe ne rien mettre)

...

setup.kibana:
  host: "localhost:5601"

...

packetbeat.protocols:
- type: icmp
  enabled: false

- type: dns
  ports: [53]

- type: http
  ports: [80]

- type: mysql
  ports: [3306]

```

Ci-dessous une explication détaillée sur **les options de configuration Packetbeat** utilisées:

- **packetbeat.interfaces.device** : ici on détermine quelle interface réseau surveiller. Dans notre cas, nous allons écouter tous les paquets envoyés ou reçus par le serveur, mais vous pouvez choisir le nom d'une interface réseau spécifique si besoin.
- **packetbeat.protocols** : dans cette section Protocoles, nous devons configurer les ports sur lesquels Packetbeat peut trouver chaque protocole. Habituellement, les valeurs par défaut dans le fichier de configuration suffiront, mais si vous utilisez des ports non standard, c'est l'endroit pour les ajouter.

Dans notre cas notre application est desservie par un serveur web et une base de données MySQL.

- **setup.kibana** : pour que les tableaux de bord fonctionnent, nous devons spécifier le point de terminaison Kibana. Vous devrez entrer l'URL de votre hôte Kibana et vos informations d'identification (nom d'utilisateur/mot de passe) si nécessaire.
- **output.elasticsearch** : spécifie la sortie à laquelle nous envoyons les métriques Packetbeat. Nous utilisons Elasticsearch, vous devrez donc fournir l'hôte, le protocole et les informations d'identification Elasticsearch si nécessaire.

Une fois terminé, relancez Packetbeat:

```
sudo systemctl restart packetbeat
```

## Visualisation des dashboards

Avant de visualiser nos dashboards, vérifiez au préalable qu'Elasticsearch et Kibana sont en cours d'exécution et qu'Elasticsearch est prêt à recevoir des données de Packetbeat:

```
sudo systemctl status elasticsearch kibana
```

### Résultat :

```
? elasticsearch.service - Elasticsearch
  Loaded: loaded (/lib/systemd/system/elasticsearch.service; disabled; vendor preset: enabled)
  Active: active (running) since Wed 2021-07-14 12:29:01 CEST; 1h 29min ago

? kibana.service - Kibana
  Loaded: loaded (/etc/systemd/system/kibana.service; disabled; vendor preset: enabled)
  Active: active (running) since Wed 2021-07-14 12:28:42 CEST; 1h 30min ago
```



Ensuite lancez la sous-commande `setup` pour charger les tableaux de bord dans Kibana (la commande peut prendre un peu de temps pour se terminer):

```
sudo packetbeat setup
```

### Résultat :

```
Overwriting ILM policy is disabled. Set `setup.ilm.overwrite: true` for enabling.  
  
Index setup finished.  
Loading dashboards (Kibana must be running and reachable)  
Loaded dashboards
```

Afin de charger quelques données pour Packetbeat, visitez plusieurs fois la page <http://localhost> et ajoutez-y quelques articles :

← → ↻ 🏠 🌐 http://localhost

Mon app Packetbeat Accueil Articles

# Articles

## Nouveau article

Titre \*

Nom de l'auteur \*

Contenu \*

Envoyer

Vous pouvez également simuler plusieurs visites, en utiliser la commande `curl` avec une boucle `for` :

```
for i in {1..20}; do curl http://localhost -s > /dev/null; done
```

Une fois les données chargées, tout pour Metricbeat pour découvrir vos logs, rendez-vous dans <http://localhost:5601/> et sur le menu à gauche cliquez sur Discover:



Home



Home

## Recently viewed



No recently viewed items



Kibana



Discover

Dashboard

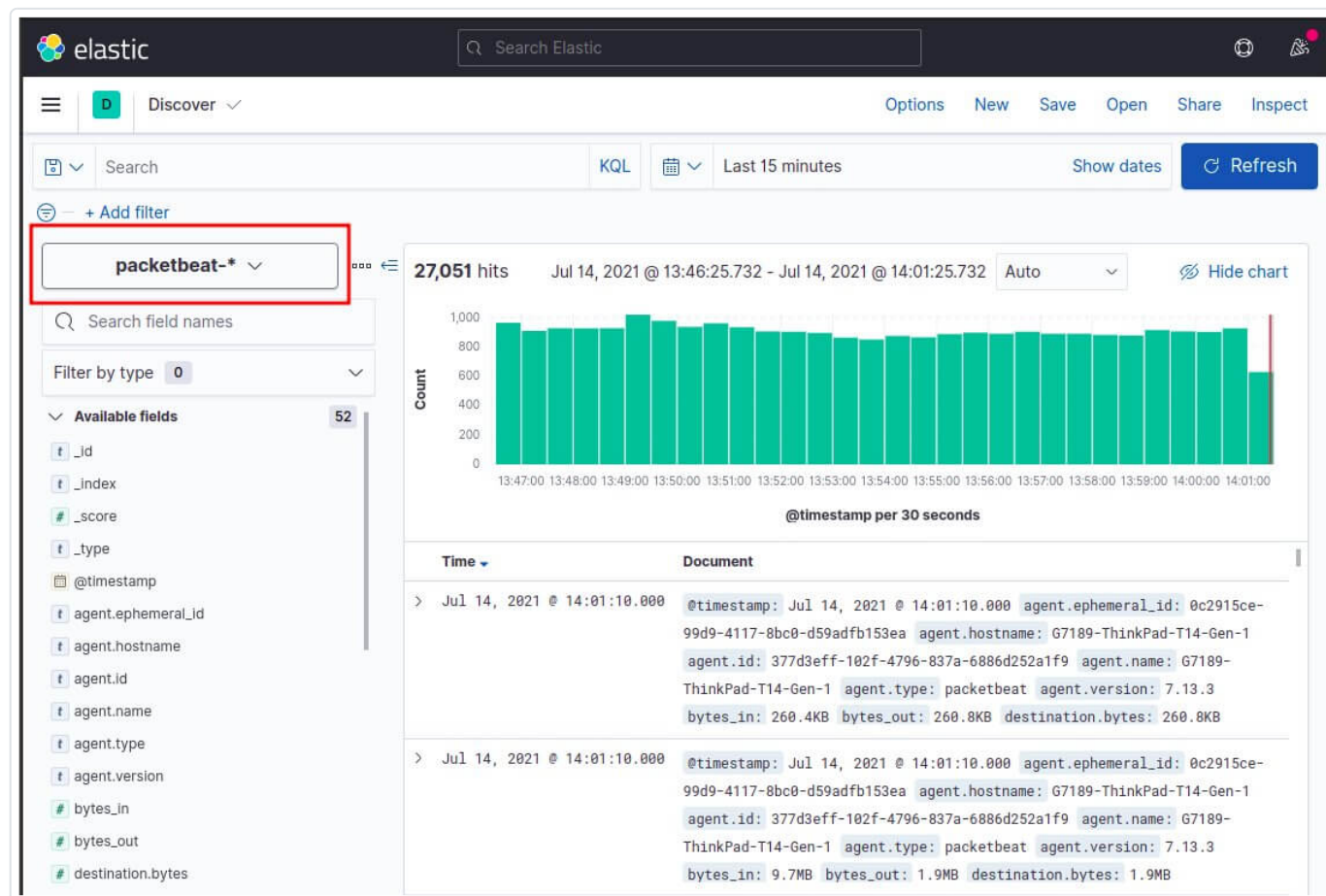
Canvas

Maps

Machine Learning

Visualize

Choisissez ensuite le pattern index **packetbeat-\*** pour visualiser les logs de Packetbeat:



L'étape suivante est de visualiser notre tableau de bord afin de visualiser la collection de visualisations en temps réel issue par notre expéditeur Packetbeat. Pour ce faire, sur le menu à gauche cliquez sur Dashboard:



**Kibana**



Discover

Dashboard

Canvas

Maps

Machine Learning

Visualize



**Observability**



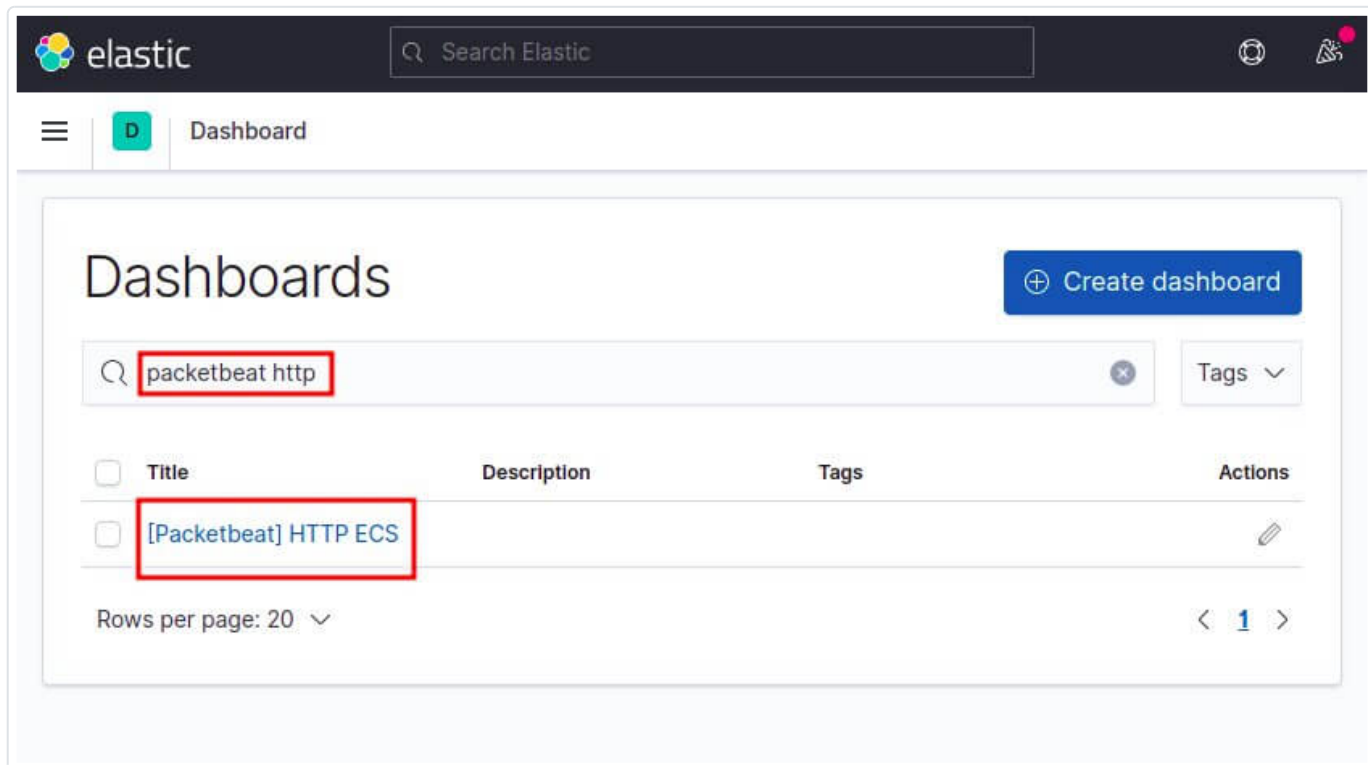
Logs

Metrics

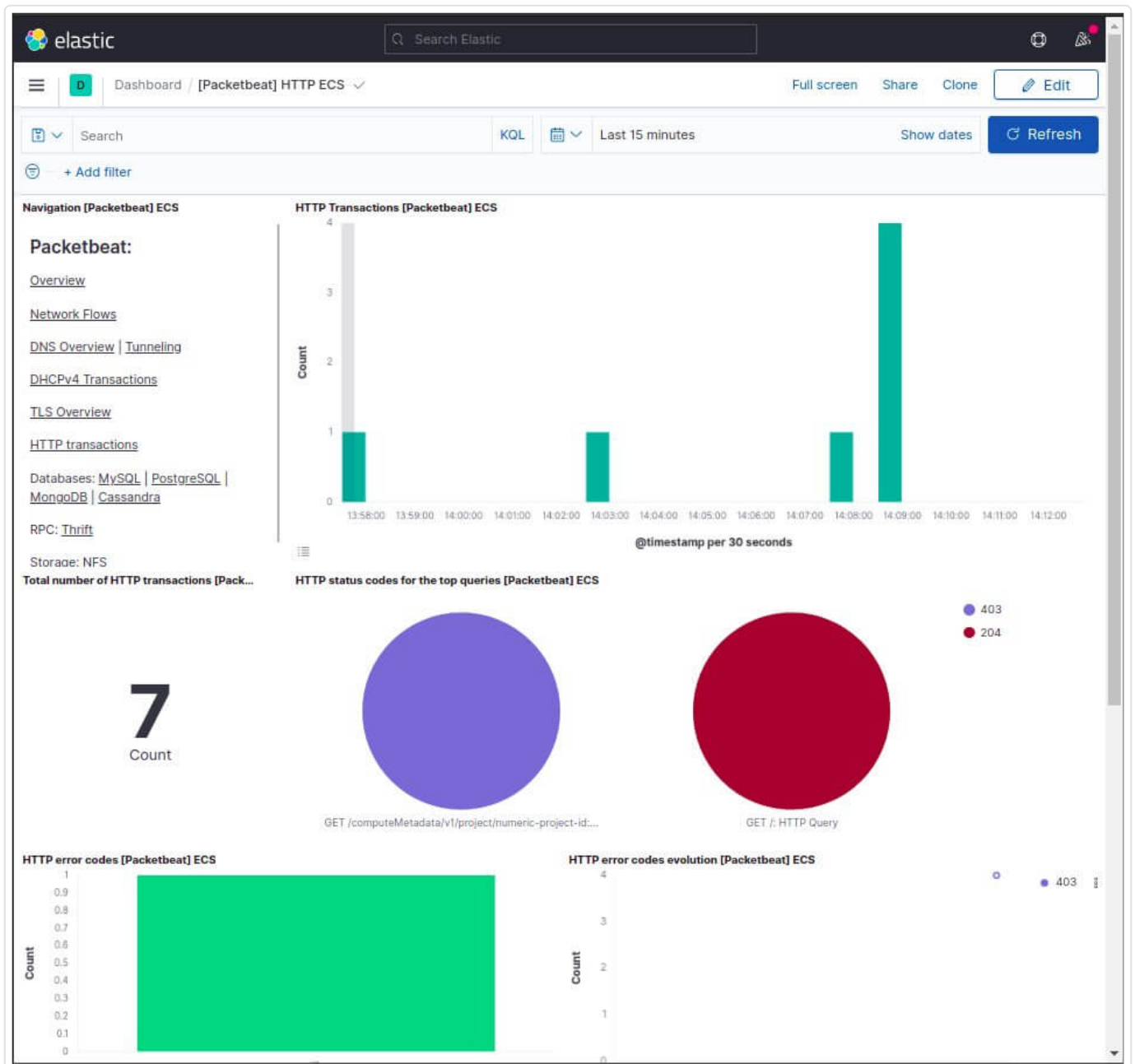
APM

Uptime

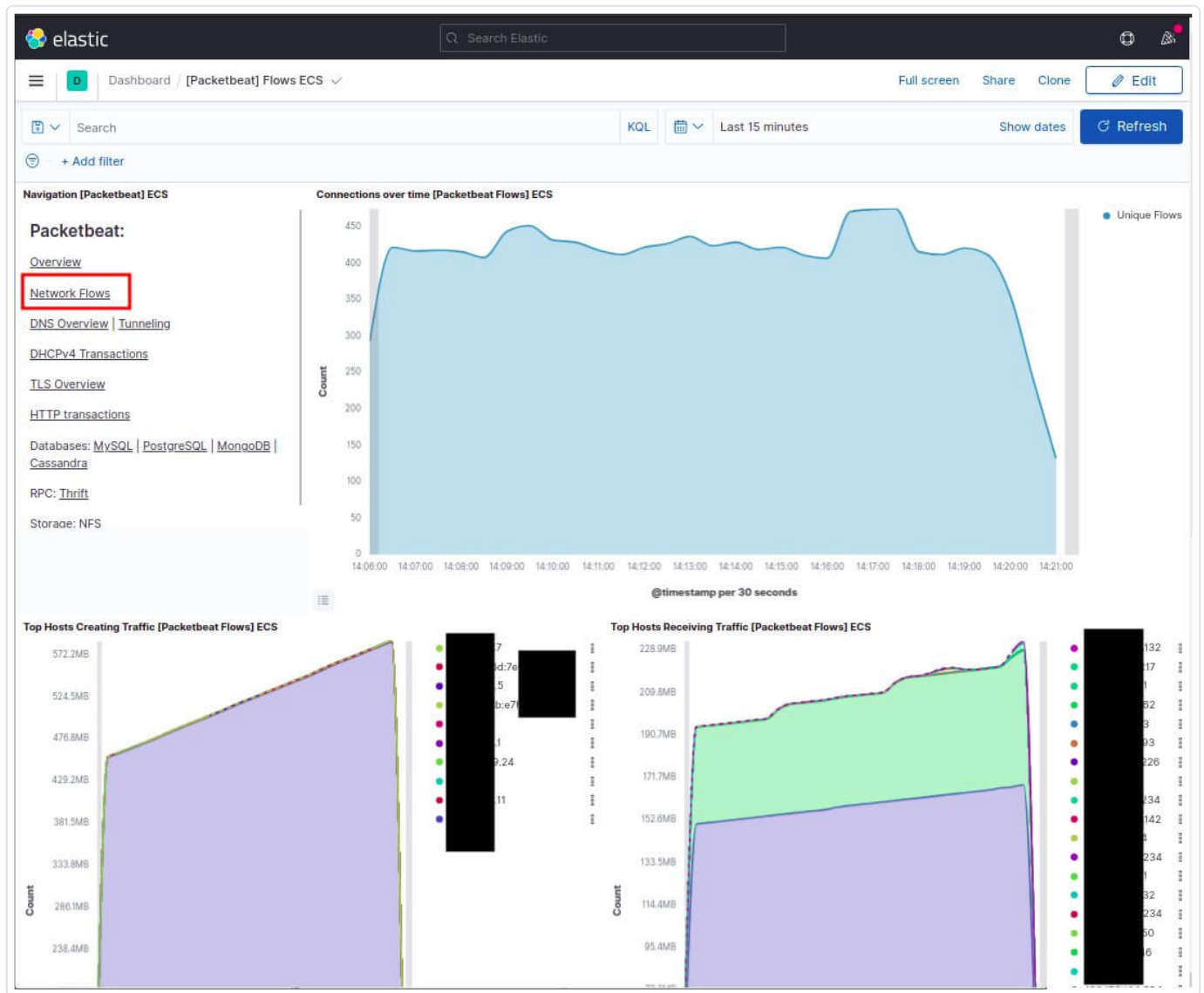
Recherchez et cliquez sur les Dashboards de Packetbeat:



Vous obtenez ainsi plusieurs dashboards. Le premier type nommé "HTTP Transactions" ressemble à l'image ci-dessous et vous donne le taux de code de réponse et requête HTTP:

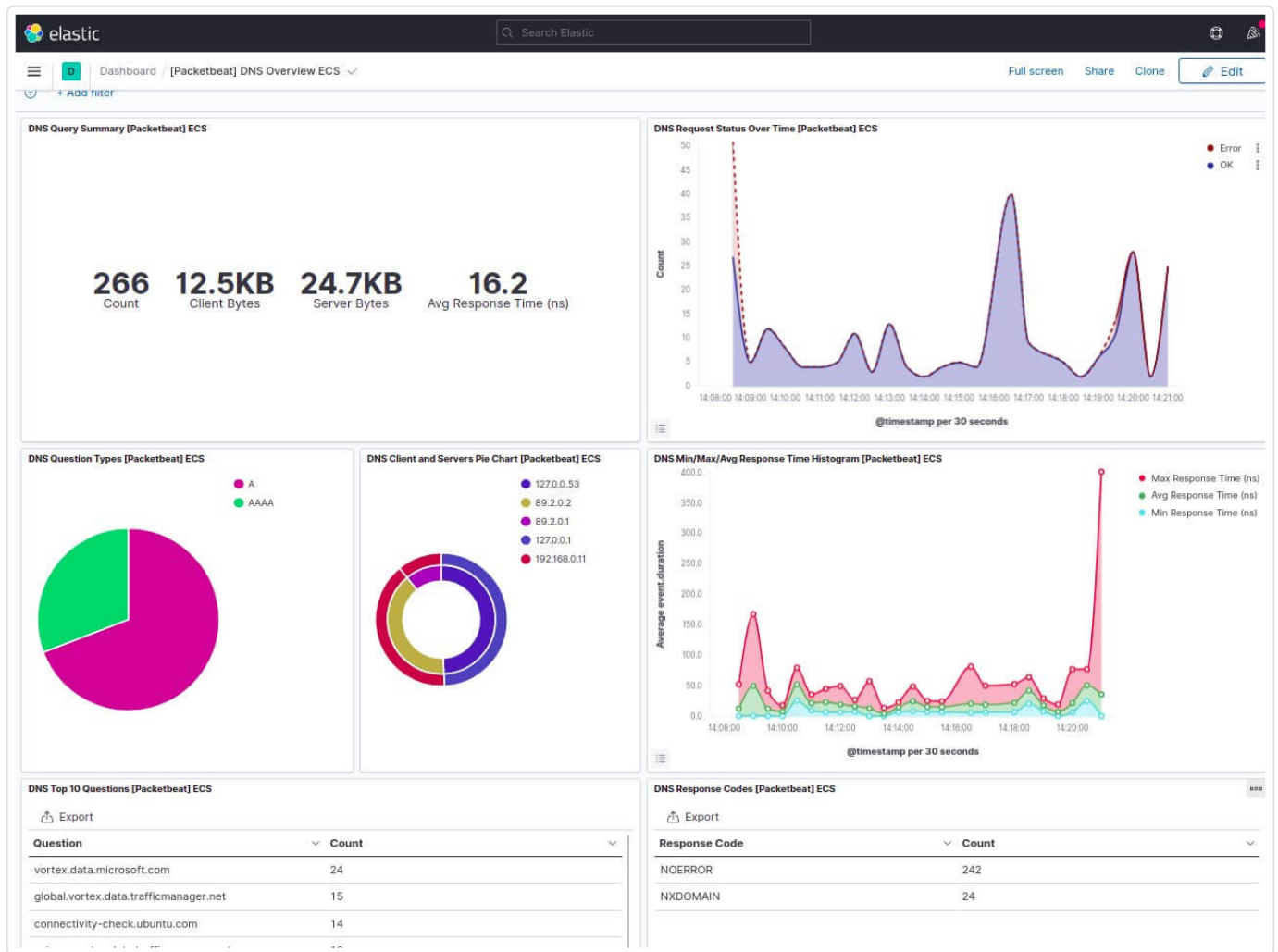


Dans le second nommé "Network Flows", vous avez des informations sur le flux de réseau avec le nombre de connexions dans le temps et les hôtes créant/recevant du trafic incluant également leur consommation réseaux:

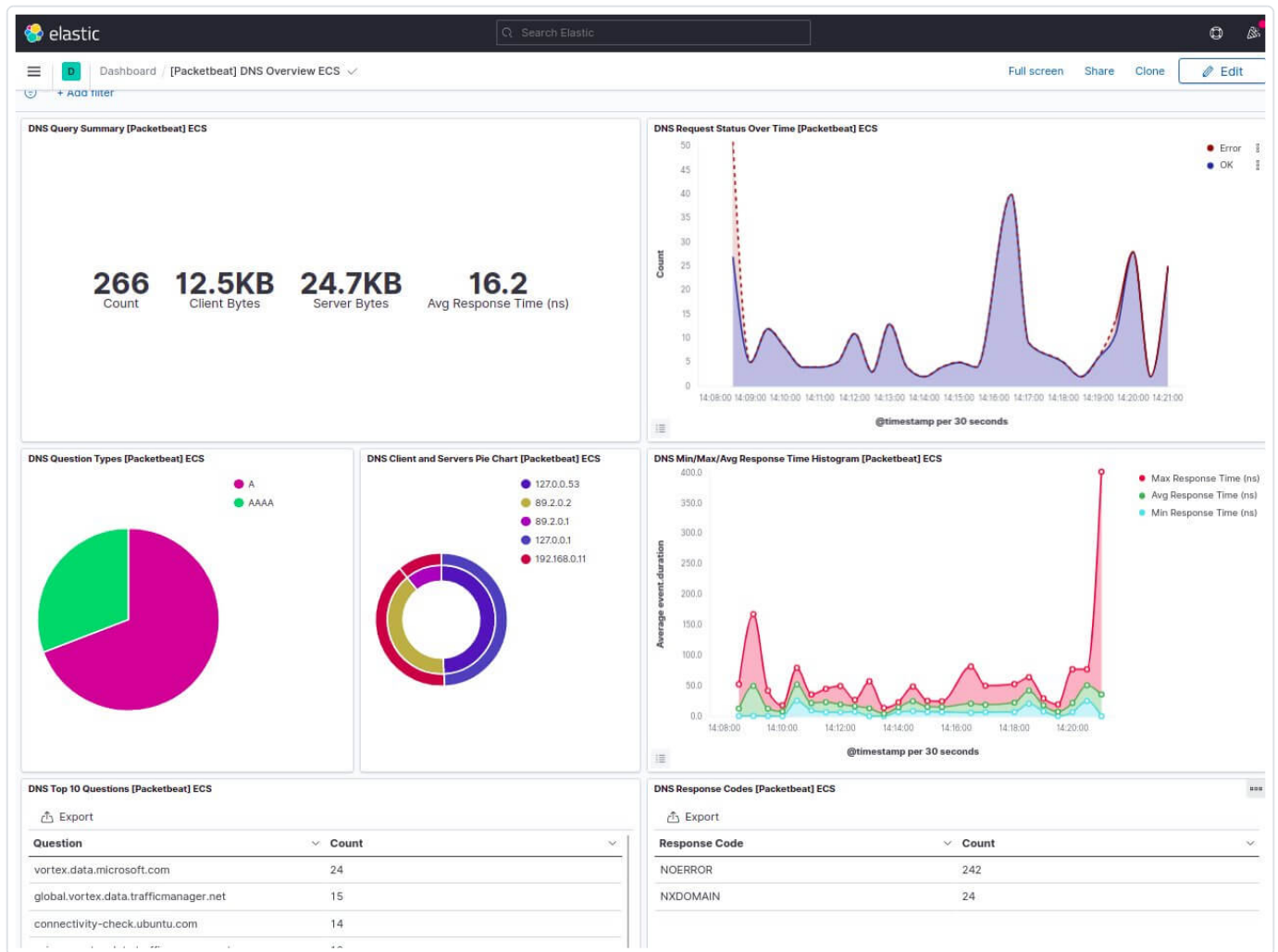


Dans le troisième nommé "DNS Overview", vous avez un résumé sur les requêtes DNS avec le temps de réponse moyen, le type de question de votre serveur DNS, histogramme du temps de réponse DNS, etc... :





Dans le quatrième nommé "Databases: MySQL", vous avez le nombre du type de requête SQL utilisé (SELECT, INSERT, etc ...), le débit de votre serveur MySQL, etc... :



Vous avez également d'autres tableaux de bord prêt à l'emploi que vous pouvez visualiser par vous-même.

## Conclusion

Maintenant que les métriques réseaux de votre application sont centralisées grâce à Packetbeat, et que vous êtes en mesure de les visualiser avec Kibana, vous devriez pouvoir voir ce que font vos serveurs en un coup d'œil.

Grâce à Packetbeat nous avons pu **avoir rapidement des tableaux de bord de surveillance du réseau opérationnel**, nous donnant une idée en temps réel des paquets transmis sur le fil.