

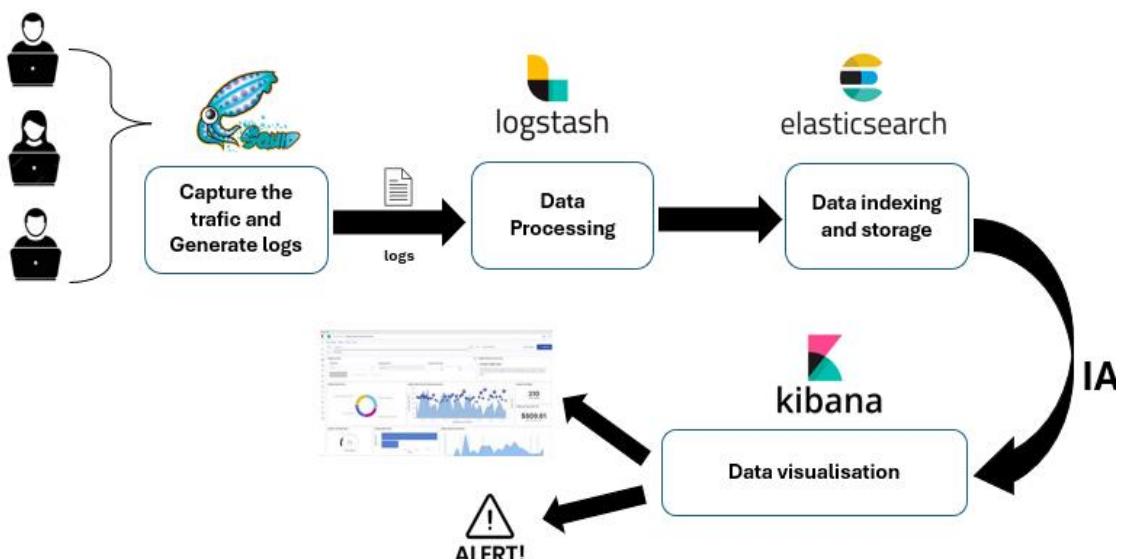
Rapport technique sur la configuration des outils du projet

Rappel de la problématique :

Dans le cadre de la surveillance du réseau d'entreprise, nous proposons une solution intégrée combinant **un proxy transparent**, **une analyse avancée des logs**, **un moteur de recherche et de visualisation des données**, ainsi qu'**un système d'intelligence artificielle et d'alerte**. Cette approche permet d'identifier en temps réel les activités suspectes tout en optimisant les performances du réseau.

L'architecture mise en place repose sur les éléments suivants :

- **Squid (Proxy Transparent)** : Capture et filtre le trafic réseau HTTP/HTTPS.
- **Logstash** : Collecte, traite et transmet les logs vers Elasticsearch.
- **Elasticsearch** : Stocke et indexe les logs pour une recherche rapide et efficace.
- **Kibana** : Fournit des outils de visualisation et d'analyse sous forme de tableaux de bord interactifs.
- **Intelligence Artificielle (IA)** : Analyse comportementale pour détecter les activités suspectes.
- **Système d'Alerte** : Génération de notifications en cas d'anomalies détectées.



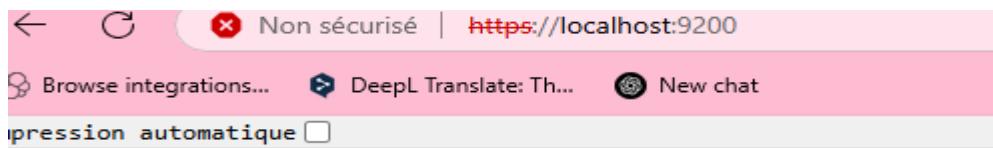
Configuration des Outils, Collecte des Données et Autoconfiguration de Squid Proxy

1. Configuration des Outils de Projet

Le projet repose sur une architecture d'analyse et de filtrage intelligent du trafic réseau via :

- **Elasticsearch** : Pour configurer Elasticsearch, on utilise la commande **elasticsearch.bat**, ce qui génère un mot de passe et un nom d'utilisateur.

- **Nom d'utilisateur** : elastic
 - **Mot de passe** : mDLu5n7azx_OxViC4=Fz
 - **Token de connexion pour Kibana** :
- eyJZXIiOiI4LjE0LjAiLCJhZHlOIlsIMTkyLjE2OC41Ni4xOjkyMDAiXSwiZmdyljoiZTk0MGVkNTZiNTc0YjlkYWY4MGlwYzI2N2MxNzEwMDU4NmUzYmFIMzgyZWUxZGM5NDliN2ZjYzZhNTc0ZWQwNCIsImtleSI6IkVnR0tHSIVCZGpDajd1NGhRSkltOmdub2VhbWVEUm55Z3NjZkNoTlVvX3cifQ==
- , puis on accède à l'interface via l'URL suivante :
 - [🔗 https://localhost:9200/](https://localhost:9200/)



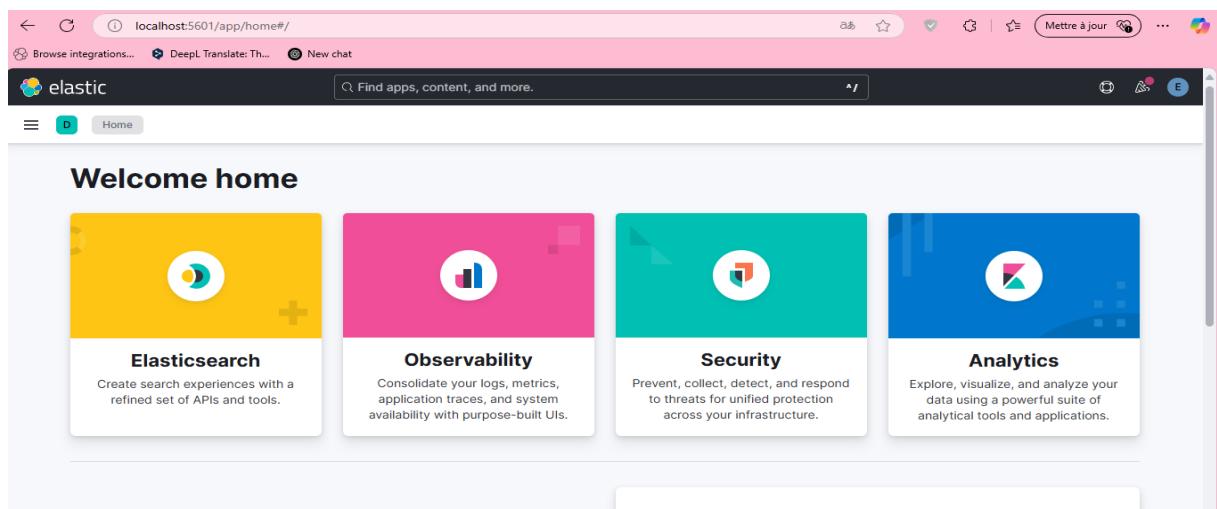
```

{
  "name" : "MOUNA",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "X-I-i0LuQOWPrx3HuS9Abg",
  "version" : {
    "number" : "8.17.2",
    "build_flavor" : "default",
    "build_type" : "zip",
    "build_hash" : "747663ddda3421467150de0e4301e8d4bc636b0c",
    "build_date" : "2025-02-05T22:10:57.067596412Z",
    "build_snapshot" : false,
    "lucene_version" : "9.12.0",
    "minimum_wire_compatibility_version" : "7.17.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "You Know, for Search"
}
  
```

- **Kibana** : Pour configurer Kibana, on utilise la commande **kibana.bat**, puis on accède à l'interface via l'URL suivante :

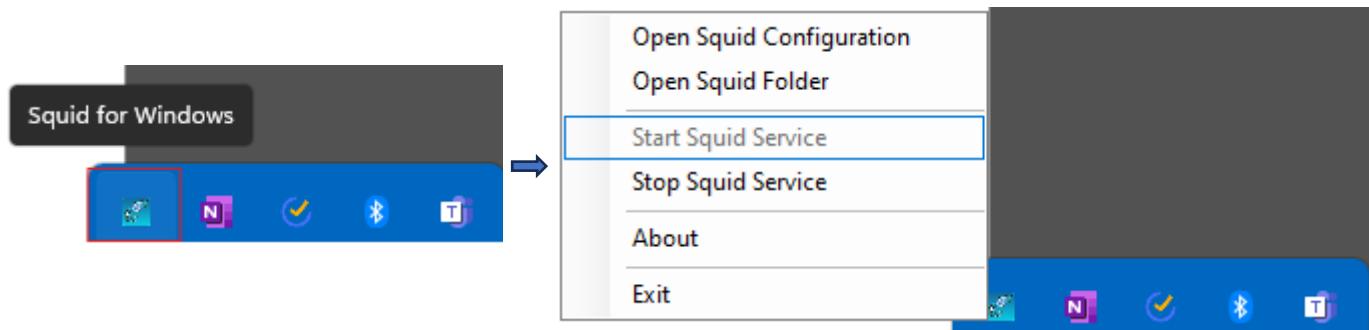
[🔗 http://localhost:5601/](http://localhost:5601/)

Cela permet d'interagir avec Elasticsearch et de visualiser les données indexées.

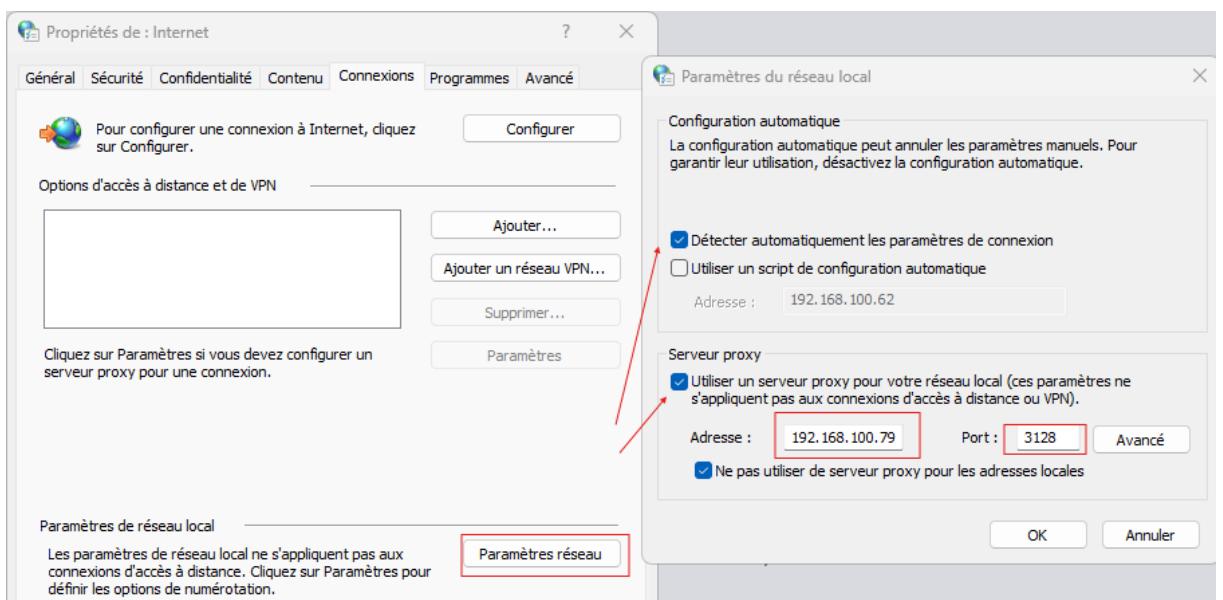


- **Squid Proxy** : pour la gestion du trafic web :

On redémarre Squid.



Ensuite configurer le proxy sur le poste client : Internet Explorer : Outils/Options Internet/Onglet Connexions/Paramètres réseau



On envisage de connecter un ordinateur à un second PC, qui fera office de proxy chargé de collecter et traiter les logs. Pour cela, nous avons attribué l'adresse IP 192.168.100.79 au proxy, puis nous nous y sommes connectés.

- **Logstash :**

Dans un premier temps, nous souhaitons **connecter Logstash**, qui collecte, filtre et transforme les logs générés par Squid, à **Elasticsearch** afin de les stocker et les indexer, puis les visualiser et les analyser sur Kibana pour le diagnostic.

➔ **Squid**, en tant que proxy cache, génère des logs détaillant les requêtes des utilisateurs, permettant d'analyser le trafic réseau, de détecter d'éventuelles anomalies et d'optimiser la bande passante. **Logstash** agit comme un outil d'ingestion de données, assurant le traitement et l'envoi des logs vers **Elasticsearch**, un moteur de recherche et d'analyse distribué qui permet un accès rapide et efficace aux données indexées. Enfin, **Kibana** offre une interface de visualisation intuitive qui facilite l'exploration et l'analyse des logs à travers des tableaux de bord interactifs et des graphiques.

```

input {
  file {
    # Utilisation d'un chemin sans espaces ou avec des guillemets doubles
    path => "C:/Program Files/k/squid/var/log/squid/access.log"
    start_position => "beginning"
    since_db_path => "NUL"
    codec => plain { charset => "UTF-8" }
  }
}

filter {
  grok {
    match => {
      "message" => "%{NUMBER:timestamp} %{NUMBER:duration} %{IP:client_ip} %{WORD:method} %{NUMBER:status} %{NUMBER:bytes} %{WORD:action} %{NOTSPACE:site}%{NOTSPACE:other_info} %{NOTSPACE:hierarchy} %{IP:server_ip} %{GREEDYDATA:additional_info}"
    }
    ecs_compatibility => disabled
  }

  # Filtrage de la date pour le timestamp UNIX
  date {
    match => ["timestamp", "UNIX"]
    target => "@timestamp"
  }

  # Ajout de la géolocalisation à partir de l'IP du client
  geoip {
    source => "client_ip"
    target => "geoip"
  }
}

```

Bloc input pour lire les logs depuis un fichier squid

```

output {
  # Envoi des logs vers Elasticsearch
  elasticsearch {
    hosts => ["https://localhost:9200"]
    user => "elastic"
    password => "mDLu5n7azx_0xViC4=Fz"
    index => "squid-logstash-%{+YYYY.MM.dd}"
    ssl => false
  }
}

```

section output de Logstash.conf qui permet envoi des données traitées à Elasticsearch.

puis activer logstash avec la commande :

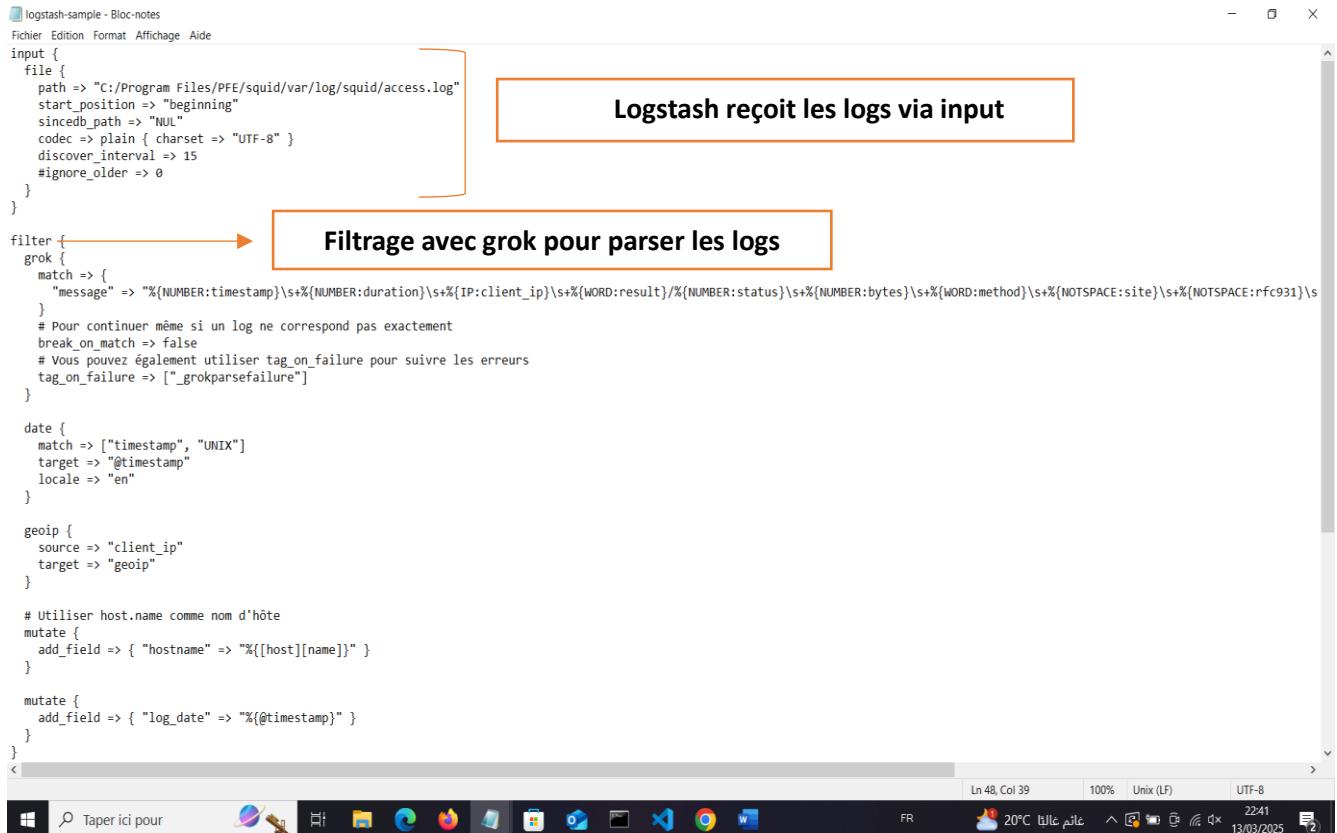
```
C:\Program Files\k\logstash-8.17.2\bin>logstash.bat -f "C:/Program Files/k/logstash-8.17.2/config/logstash.conf"
```

- On peut maintenant voir les log sur kibana

Name	Health	Status	Primaries	Replicas	Document...	Storage si...	Data stream
squid-logstash-2025.03.05	yellow	open	1	1	356	454.95kb	
squid-logstash-2025.03.06	yellow	open	1	1	200	91.55kb	
squid-logstash-2025.03.08	yellow	open	1	1	345	216.1kb	
squid-logstash-2025.03.09	yellow	open	1	1	1 090	500.82kb	

2. Étapes de Collecte des Données avec Logstash

✓ Première Méthode – Collecte via Logstash



```
logstash-sample - Bloc-notes
Fichier Edition Format Affichage Aide
input {
    file {
        path => "C:/Program Files/PFE/squid/var/log/squid/access.log"
        start_position => "beginning"
        since_db_path => "NUL"
        codec => plain { charset => "UTF-8" }
        discover_interval => 15
        #ignore_older => 0
    }
}

filter {
    grok {
        match => {
            "message" => "%{NUMBER:timestamp}\s+ %{NUMBER:duration}\s+ %{IP:client_ip}\s+ %{WORD:result}/%{NUMBER:status}\s+ %{NUMBER:bytes}\s+ %{WORD:method}\s+ %{NOTSPACE:site}\s+ %{NOTSPACE:rfc931}\s"
        }
        # Pour continuer même si un log ne correspond pas exactement
        break_on_match => false
        # Vous pouvez également utiliser tag_on_failure pour suivre les erreurs
        tag_on_failure => ["_grokparsefailure"]
    }

    date {
        match => ["timestamp", "UNIX"]
        target => "@timestamp"
        locale => "en"
    }

    geoip {
        source => "client_ip"
        target => "geoip"
    }

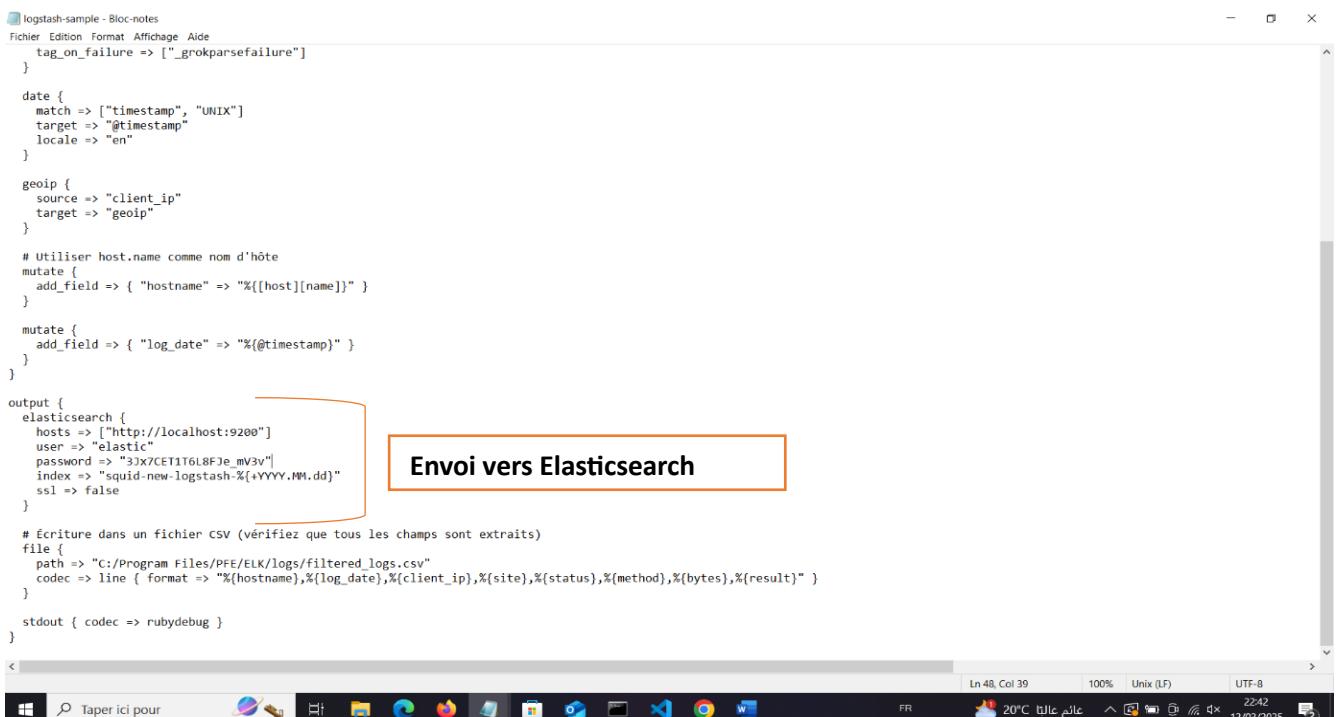
    # Utiliser host.name comme nom d'hôte
    mutate {
        add_field => { "hostname" => "%{[host][name]}" }
    }

    mutate {
        add_field => { "log_date" => "%{@timestamp}" }
    }
}

output {
    elasticsearch {
        hosts => ["http://localhost:9200"]
        user => "elastic"
        password => "33X7CET1T6L8FJe_mv3v"
        index => "squid-new-logstash-%{+YYYY.MM.dd}"
        ssl => false
    }

    # écriture dans un fichier CSV (vérifiez que tous les champs sont extraits)
    file {
        path => "C:/Program Files/PFE/ELK/logs/filtered.logs.csv"
        codec => line { format => "%{hostname},%{log_date},%{client_ip},%{site},%{status},%{method},%{bytes},%{result}" }
    }

    stdout { codec => rubydebug }
}
```



```
logstash-sample - Bloc-notes
Fichier Edition Format Affichage Aide
tag_on_failure => ["_grokparsefailure"]

date {
    match => ["timestamp", "UNIX"]
    target => "@timestamp"
    locale => "en"
}

geoip {
    source => "client_ip"
    target => "geoip"
}

# Utiliser host.name comme nom d'hôte
mutate {
    add_field => { "hostname" => "%{[host][name]}" }
}

mutate {
    add_field => { "log_date" => "%{@timestamp}" }
}

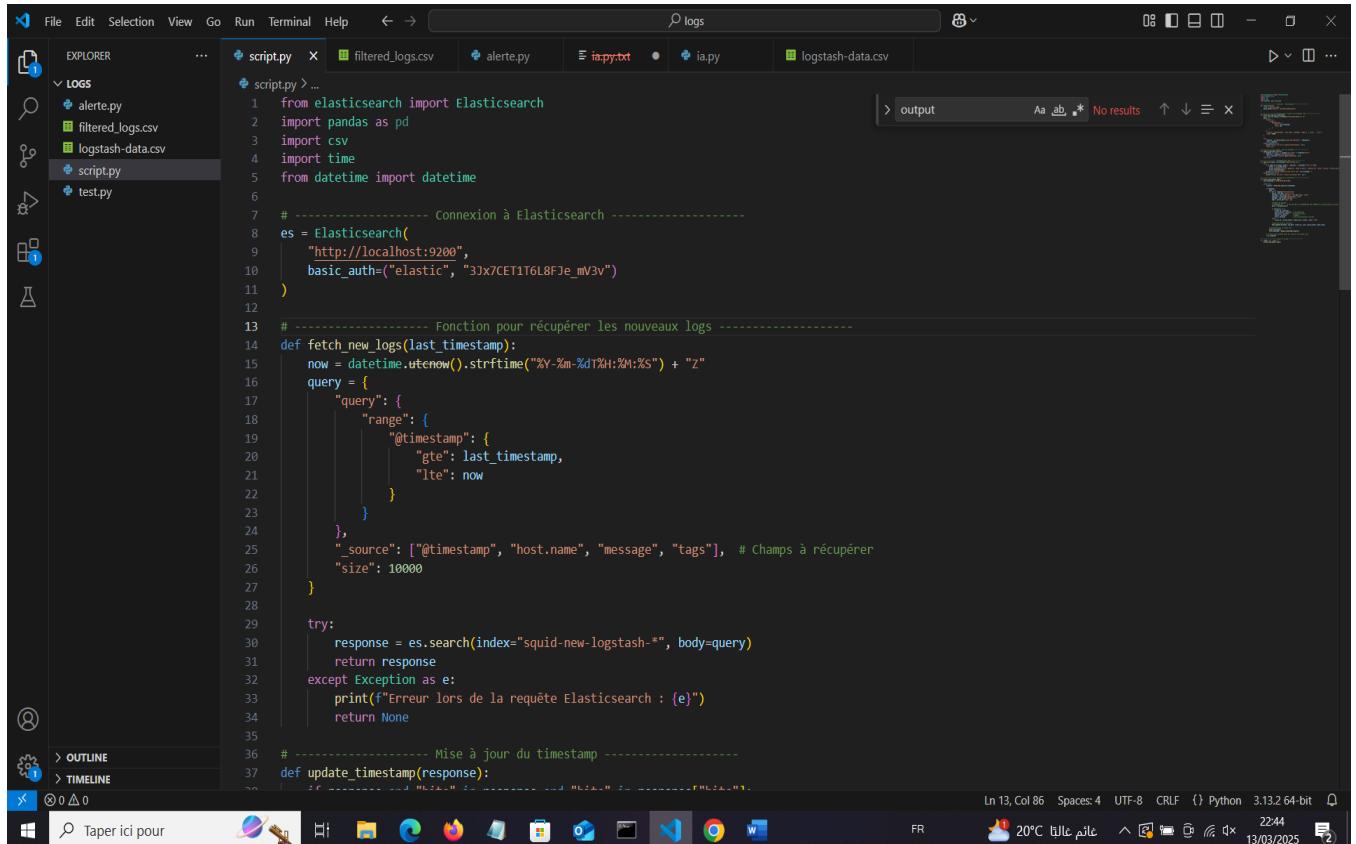
output {
    elasticsearch {
        hosts => ["http://localhost:9200"]
        user => "elastic"
        password => "33X7CET1T6L8FJe_mv3v"
        index => "squid-new-logstash-%{+YYYY.MM.dd}"
        ssl => false
    }

    # écriture dans un fichier CSV (vérifiez que tous les champs sont extraits)
    file {
        path => "C:/Program Files/PFE/ELK/logs/filtered.logs.csv"
        codec => line { format => "%{hostname},%{log_date},%{client_ip},%{site},%{status},%{method},%{bytes},%{result}" }
    }

    stdout { codec => rubydebug }
}
```

✓ Deuxième Méthode – Script Python d'Export des Logs depuis Elasticsearch

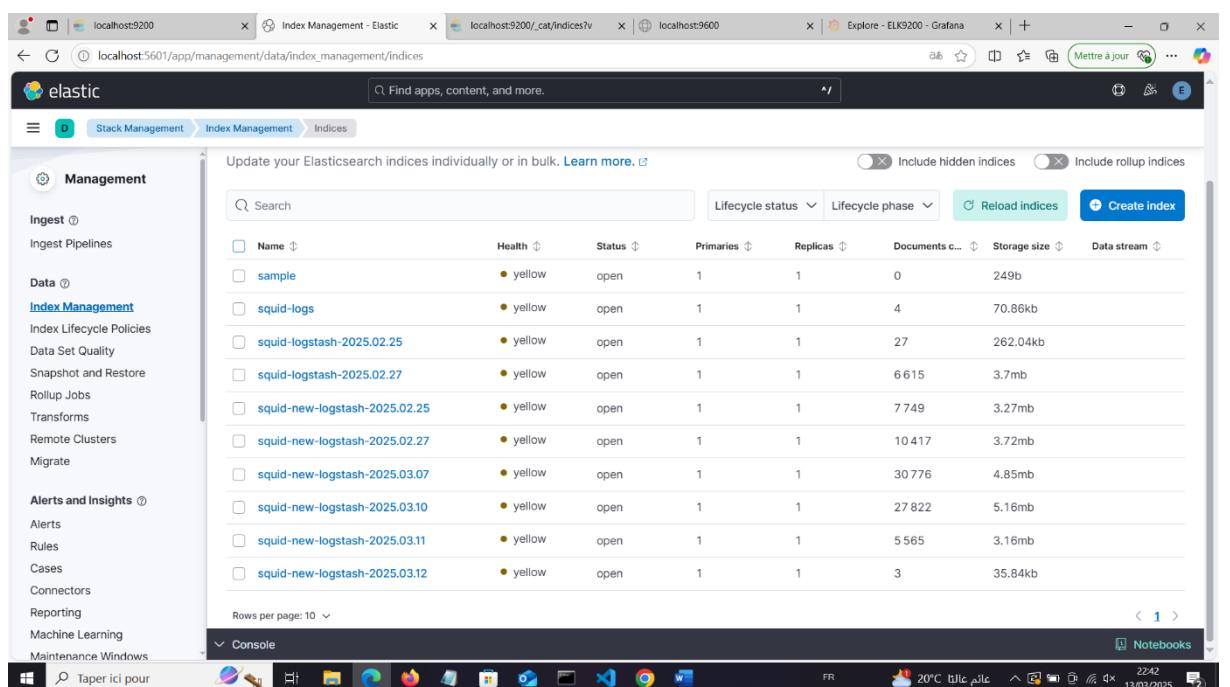
- Script pour exporter les logs indexés vers un fichier CSV pour traitement IA :



The screenshot shows a code editor interface with a dark theme. The left sidebar has a tree view under 'LOGS' containing 'alerte.py', 'filtered_logs.csv', 'logstash-data.csv', 'script.py', and 'test.py'. The main editor area displays a Python script named 'script.py'. The script imports Elasticsearch, pandas, csv, time, and datetime. It connects to Elasticsearch at 'http://localhost:9200' with basic authentication ('elastic' and 'mV3v'). A function 'fetch_new_logs' retrieves logs from Elasticsearch between 'last_timestamp' and 'now' (current timestamp). It uses a query with a range filter on '@timestamp' and specifies '_source' fields like '@timestamp', 'host.name', 'message', and 'tags'. The 'size' is set to 10000. A try-except block handles errors. Another function 'update_timestamp' is defined but not fully implemented. The status bar at the bottom shows 'Ln 13, Col 86' and 'Python 3.12.6-bit'. The taskbar at the bottom includes icons for various applications like File Explorer, Task Manager, and browser tabs.

```
File Edit Selection View Go Run Terminal Help ← → ⌘ logs
EXPLORER ... script.py filtered_logs.csv alerte.py ia.py.txt ia.py logstash-data.csv
LOGS
script.py > ...
1  from elasticsearch import Elasticsearch
2  import pandas as pd
3  import csv
4  import time
5  from datetime import datetime
6
7  # ----- Connexion à Elasticsearch -----
8  es = Elasticsearch(
9      "http://localhost:9200",
10     basic_auth=("elastic", "mV3v")
11 )
12
13 # ----- Fonction pour récupérer les nouveaux logs -----
14 def fetch_new_logs(last_timestamp):
15     now = datetime.utcnow().strftime("%Y-%m-%dT%H:%M:%S") + "Z"
16     query = {
17         "query": {
18             "range": {
19                 "@timestamp": {
20                     "gte": last_timestamp,
21                     "lte": now
22                 }
23             },
24             "_source": ["@timestamp", "host.name", "message", "tags"], # Champs à récupérer
25             "size": 10000
26         }
27     }
28
29     try:
30         response = es.search(index="squid-new-logstash-*", body=query)
31         return response
32     except Exception as e:
33         print(f"Erreur lors de la requête Elasticsearch : {e}")
34         return None
35
36 # ----- Mise à jour du timestamp -----
37 def update_timestamp(response):
38
FR 20°C Julie 22:44 13/03/2025
```

✓ 3. Résultats Obtenus



The screenshot shows the Elasticsearch Management interface. On the left, there's a navigation sidebar with 'Management', 'Ingest', 'Data', 'Index Management', 'Alerts and Insights', and 'Machine Learning'. The main area is titled 'Management' and shows a table of indices. The columns include Name, Health, Status, Primaries, Replicas, Documents, Storage size, and Data stream. The table lists several indices: 'sample' (yellow), 'squid-logs' (yellow), 'squid-logstash-2025.02.25' (yellow), 'squid-new-logstash-2025.02.25' (yellow), 'squid-new-logstash-2025.02.27' (yellow), 'squid-new-logstash-2025.03.07' (yellow), 'squid-new-logstash-2025.03.10' (yellow), 'squid-new-logstash-2025.03.11' (yellow), and 'squid-new-logstash-2025.03.12' (yellow). The status for all indices is 'open'. The storage sizes range from 3.7mb to 262.04kb. The bottom of the screen shows a taskbar with various application icons and a system tray with the date and time.

- Logs collectés en temps réel et affichés dans Kibana.

The screenshot shows a Windows desktop environment with several open browser tabs. The tabs include:

- localhost:9200 (Discover - Elastic)
- localhost:9200/_cat/indices?v (localhost:9600)
- localhost:9200/_cat/indices?v (Explore - ELK9200 - Grafana)
- localhost:5601/app/discover#?_g=(filters:[],refreshInterval(pause:1,value:60000),time:(from:now-15m,to:now))&_a=(columns:[],dataSource:(dataViewId:5460cd74-a63...))

The main window is Kibana, specifically the 'Discover' view for the 'squid-new-logstash-2025.03.12' index pattern. It shows a table of log documents with columns for '_index', '_id', '_score', '_type', '_id', '_index', '_score', '_type', and '_id'. The data is sorted by '_id'. A sidebar on the left shows available fields, empty fields, and meta fields. The bottom of the screen shows the Windows taskbar with various pinned icons like File Explorer, Task View, and Start.

- Exportation réussie dans un fichier CSV pour traitement IA.

Par script :

```

1 Nom de l'appareil,Date et heure,Adresse IP,Site,Action,Taille de données
2 DESKTOP-SNJF2KJ,2025-03-12T10:36:24.0352,192.168.100.75,play.google.com:443,TCP_TUNNEL_ABORTED/200,39
3 DESKTOP-SNJF2KJ,2025-03-12T10:36:25.9677,192.168.100.75,play.google.com:443,TCP_TUNNEL_ABORTED/200,39
4 DESKTOP-SNJF2KJ,2025-03-12T10:36:25.8892,192.168.100.75,www.google.com:443,TCP_TUNNEL/200,1531
5 DESKTOP-SNJF2KJ,2025-03-12T10:36:31.1592,192.168.100.75,ab.chatpt.com:443,TCP_TUNNEL_ABORTED/200,39
6 DESKTOP-SNJF2KJ,2025-03-12T10:36:36.8932,192.168.100.75,mail.google.com:443,TCP_TUNNEL/200,7087
7 DESKTOP-SNJF2KJ,2025-03-12T10:36:24.1412,192.168.100.75,mobile.events.data.microsoft.com:443,TCP_TUNNEL/200,7242
8 DESKTOP-SNJF2KJ,2025-03-12T10:36:24.1712,192.168.100.75,play.google.com:443,TCP_TUNNEL_ABORTED/200,39
9 DESKTOP-SNJF2KJ,2025-03-12T10:36:25.8902,192.168.100.75,www.google.com:443,TCP_TUNNEL/200,7192
10 DESKTOP-SNJF2KJ,2025-03-12T10:36:25.8902,192.168.100.75,www.google.com:443,TCP_TUNNEL/200,4878
11 DESKTOP-SNJF2KJ,2025-03-12T10:36:25.8902,192.168.100.75,a.net.cloudflare.com:443,TCP_TUNNEL/200,4120
12 DESKTOP-SNJF2KJ,2025-03-12T10:36:25.8902,192.168.100.75,play.google.com:443,TCP_TUNNEL/200,16759
13 DESKTOP-SNJF2KJ,2025-03-12T10:36:25.8932,192.168.100.75,www.google.com:443,TCP_TUNNEL/200,1531
14 DESKTOP-SNJF2KJ,2025-03-12T10:36:25.8992,192.168.100.75,www.google.com:443,TCP_TUNNEL/200,5417
15 DESKTOP-SNJF2KJ,2025-03-12T10:36:26.0982,192.168.100.75,play.google.com:443,TCP_TUNNEL_ABORTED/200,39
16 DESKTOP-SNJF2KJ,2025-03-12T10:36:28.9232,192.168.100.75,ing-s-msn.com.akamaiized.net:443,TCP_TUNNEL_ABORTED/200,39
17 DESKTOP-SNJF2KJ,2025-03-12T10:36:26.6672,192.168.100.75,chatgpt.com:443,TCP_TUNNEL_ABORTED/200,39
18 DESKTOP-SNJF2KJ,2025-03-12T10:36:30.1867,192.168.100.75,chatgpt.com:443,TCP_TUNNEL_ABORTED/200,39
19 DESKTOP-SNJF2KJ,2025-03-12T10:36:30.8942,192.168.100.75,signaler-pa.clients6.google.com:443,TCP_TUNNEL/200,2925
20 DESKTOP-SNJF2KJ,2025-03-12T10:36:30.9842,192.168.100.75,ab.chatpt.com:443,TCP_TUNNEL_ABORTED/200,39
21 DESKTOP-SNJF2KJ,2025-03-12T10:36:21.5472,192.168.100.75,www.google.com:443,TCP_TUNNEL_ABORTED/200,39
22 DESKTOP-SNJF2KJ,2025-03-12T10:36:21.3432,192.168.100.75,www.google.com:443,TCP_TUNNEL/200,39
23 DESKTOP-SNJF2KJ,2025-03-12T10:36:21.9922,192.168.100.75,play.google.com:443,TCP_TUNNEL_ABORTED/200,39
24 DESKTOP-SNJF2KJ,2025-03-12T10:36:19.4702,192.168.100.75,ing-googleapis.com:443,TCP_TUNNEL_ABORTED/200,39
25 DESKTOP-SNJF2KJ,2025-03-12T10:36:15.8742,192.168.100.75,signaler-pa.clients6.google.com:443,TCP_TUNNEL/200,13917
26 DESKTOP-SNJF2KJ,2025-03-12T10:36:14.1232,192.168.100.75,mobile.events.data.microsoft.com:443,TCP_TUNNEL/200,7121
27 DESKTOP-SNJF2KJ,2025-03-12T10:36:15.3942,192.168.100.75,th.bing.com:443,TCP_TUNNEL_ABORTED/200,39
28 DESKTOP-SNJF2KJ,2025-03-12T10:36:10.2192,192.168.100.75,www.bing.com:443,TCP_TUNNEL_ABORTED/200,39
29 DESKTOP-SNJF2KJ,2025-03-12T10:36:09.9432,192.168.100.75,www.bing.com:443,TCP_TUNNEL_ABORTED/200,39
30 DESKTOP-SNJF2KJ,2025-03-12T10:36:04.1032,192.168.100.75,mobile.events.data.microsoft.com:443,TCP_TUNNEL/200,7121
31 DESKTOP-SNJF2KJ,2025-03-12T10:36:00.5142,192.168.100.75,th.bing.com:443,TCP_TUNNEL_ABORTED/200,39
32 DESKTOP-SNJF2KJ,2025-03-12T10:35:59.8412,192.168.100.75,th.bing.com:443,TCP_TUNNEL_ABORTED/200,39
33 DESKTOP-SNJF2KJ,2025-03-12T10:35:59.0662,192.168.100.75,th.bing.com:443,TCP_TUNNEL_ABORTED/200,39
34 DESKTOP-SNJF2KJ,2025-03-12T10:35:59.3422,192.168.100.75,th.bing.com:443,TCP_TUNNEL_ABORTED/200,39
35 DESKTOP-SNJF2KJ,2025-03-12T10:35:59.4192,192.168.100.75,th.bing.com:443,TCP_TUNNEL_ABORTED/200,39
36 DESKTOP-SNJF2KJ,2025-03-12T10:35:58.6752,192.168.100.75,imr-s-men-crm.akamaiized.net:443,TCP_TUNNEL_ABORTED/200,39

```

Par logstsh :

```

EXPLORER
LOGS
alerte.py
filtered_logs.csv
logstash-data.csv
script.py
test.py

script.py filtered_logs.csv alerte.py ia.py logstash-data.csv
filtered_logs.csv

1 DESKTOP-5NJF2KJ,2025-02-25T09:31:38,732Z,192.168.100.57,api.msn.com:443,200,CONNECT,6716,TCP_TUNNEL
2 DESKTOP-5NJF2KJ,2025-02-25T09:31:53,728Z,192.168.100.57,http://ipv6.msftconnecttest.com/connecttest.txt,000,GET,0,TCP_MISS_ABORTED
3 DESKTOP-5NJF2KJ,2025-02-25T09:32:18,128Z,192.168.100.57,ssl.gstatic.com:443,200,CONNECT,7088,TCP_TUNNEL
4 DESKTOP-5NJF2KJ,2025-02-25T09:32:46,100Z,192.168.100.57,http://detectportal.firefox.com/success.txt?,000,GET,0,TCP_MISS_ABORTED
5 DESKTOP-5NJF2KJ,2025-02-25T09:32:46,102Z,192.168.100.57,push.services.mozilla.com:443,200,CONNECT,3721,TCP_TUNNEL
6 DESKTOP-5NJF2KJ,2025-02-25T09:32:57,431Z,192.168.100.57,push.services.mozilla.com:443,200,CONNECT,39,TCP_TUNNEL
7 DESKTOP-5NJF2KJ,2025-02-25T09:33:02,024Z,192.168.100.57,firefox.settings.services.mozilla.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED
8 DESKTOP-5NJF2KJ,2025-02-25T09:33:16,670Z,192.168.100.57,www.youtube.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED
9 DESKTOP-5NJF2KJ,2025-02-25T09:33:16,584Z,192.168.100.57,http://o.pki.gog/wr2,200,POST,800,TCP_MISS
10 DESKTOP-5NJF2KJ,2025-02-25T09:33:28,349Z,192.168.100.57,cdn.accounts.firefox.com:443,200,CONNECT,19179,TCP_TUNNEL
11 DESKTOP-5NJF2KJ,2025-02-25T09:33:39,583Z,192.168.100.57,http://r10.o.lencr.org/,200,POST,964,TCP_MISS
12 DESKTOP-5NJF2KJ,2025-02-25T09:33:42,498Z,192.168.100.57,www.youtube.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED
13 DESKTOP-5NJF2KJ,2025-02-25T09:33:47,761Z,192.168.100.57,http://r10.o.lencr.org/,200,POST,964,TCP_MISS
14 DESKTOP-5NJF2KJ,2025-02-25T09:33:58,388Z,192.168.100.57,http://r11.o.lencr.org/,200,POST,965,TCP_MISS
15 DESKTOP-5NJF2KJ,2025-02-25T09:34:13,315Z,192.168.100.57,clients4.google.com:443,200,CONNECT,3711,TCP_TUNNEL
16 DESKTOP-5NJF2KJ,2025-02-25T09:34:20,461Z,192.168.100.57,http://detectportal.firefox.com/success.txt?,200,GET,310,TCP_MISS
17 DESKTOP-5NJF2KJ,2025-02-25T09:34:38,425Z,192.168.100.57,ogads-pa.clients6.google.com:443,200,CONNECT,11920,TCP_TUNNEL
18 DESKTOP-5NJF2KJ,2025-02-25T09:34:44,051Z,192.168.100.57,assets.msn.com:443,200,CONNECT,8085,TCP_TUNNEL
19 DESKTOP-5NJF2KJ,2025-02-25T09:34:45,555Z,192.168.100.57,play.google.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED
20 DESKTOP-5NJF2KJ,2025-02-25T09:34:50,482Z,192.168.100.57,www.youtube.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED
21 DESKTOP-5NJF2KJ,2025-02-25T09:34:53,418Z,192.168.100.57,ssl.gstatic.com:443,200,CONNECT,3698,TCP_TUNNEL
22 DESKTOP-5NJF2KJ,2025-02-25T09:35:02,342Z,192.168.100.57,www.gstatic.com:443,200,CONNECT,108712,TCP_TUNNEL
23 DESKTOP-5NJF2KJ,2025-02-25T09:35:06,607Z,192.168.100.57,play.google.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED
24 DESKTOP-5NJF2KJ,2025-02-25T09:35:17,732Z,192.168.100.57,th.bing.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED
25 DESKTOP-5NJF2KJ,2025-02-25T09:35:23,444Z,192.168.100.57,ssl.gstatic.com:443,200,CONNECT,3538,TCP_TUNNEL
26 DESKTOP-5NJF2KJ,2025-02-25T09:35:24,834Z,192.168.100.57,www.youtube.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED
27 DESKTOP-5NJF2KJ,2025-02-25T09:35:33,498Z,192.168.100.57,play.google.com:443,200,CONNECT,4545,TCP_TUNNEL
28 DESKTOP-5NJF2KJ,2025-02-25T09:35:47,565Z,192.168.100.57,th.bing.com:443,200,CONNECT,39,TCP_TUNNEL_ABORTED

```

- Identification de certains comportements suspects : transferts de fichiers volumineux, consultation fréquente de sites freelance ou hébergement cloud.

4. Idée d'Extension : Autoconfiguration Intelligent de Squid Proxy (IA)

Principe

- Le script IA doit analyser en continu le fichier CSV.
- Si :
 - Un **transfert > 100MB** est détecté,
 - Ou si l'URL appartient à une **liste noire (freelance, hébergement, etc.)**,
- Alors :
 - Une **alerte est envoyée au manager** (email ou popup).
 - Le site ou la connexion est automatiquement **ajouté dans la blacklist de Squid**.

Développement IA – Autoconfiguration de Squid Proxy

Structure du répertoire IA :

/IA_Squid_Proxy/

|— data/

 |— logstash-data.csv

Nous sommes dans cette étapes

```
|── ia_config/
|   └── sites_freelance.txt
└── scripts/
    ├── analyse_ia.py
    └── update_blacklist.bat
└── alertes/
    └── alerte_log.txt
└── gui/
    └── ia_gui.py
```

1. Fichier sites_freelance.txt (liste noire initiale)

upwork.com

freelancer.com

fiverr.com

weworkremotely.com

2. Script Python IA : analyse_ia.py

Ce script :

- Analyse le fichier CSV des logs,
 - Déetecte les cas critiques (volume > 100MB ou site freelance),
 - Envoie une alerte (log + popup),
 - Met à jour le fichier de blacklist,
 - Relance Squid via un batch.
-

Avantages de cette solution IA

- Surveillance continue
- Détection proactive des abus réseau
- Blocage automatique sans intervention humaine
- Alerta directe au manager
- Facile à intégrer dans l'infrastructure actuelle