Institut Universitaire de Technologie,
Aix-Marseille Université

RAPPORT DE STAGE
Diplôme Universitaire de Technologie
Spécialité Réseaux et Télécommunications

Solution de cross-plateformes et Anonymisation des données

Pape Mademba DIEYE

Institut Neurosciences de la Timone

Responsable entreprise : Dipankar BACHAR

Responsable académique : Djamal MERAD

2019

Table des matières

1 Introduction

Après deux années de formation sur les technologies d'aujourd'hui, leur mode de fonctionnement et les connaissances techniques qui permettent de les mettre en place, il est nécessaire de passer à la mise en pratique des compétences acquises dans le domaine professionnel. C'est dans ce cadre que je réalise un stage de dix semaines allant du 8 avril au 14 juin à l'Institut de Neurosciences de la Timone (INT).

L'INT est un laboratoire de recherche dans le domaine des neurosciences. Il est basé à Marseille, dans le cinquième arrondissement au sein de la Faculté de médecine de la Timone. Il a pour mission de réaliser des travaux de recherche fondamentale et clinique. La collaboration des chercheurs sur différents projets a été depuis longtemps le secret de l'aboutissement de pas mal de recherches dans des domaines aussi variés que les neurosciences. Mais cette collaboration implique un partage de données médicales qui demande une certaine confidentialité d'après le règlement général européen sur la protection de données (RGPD). C'est dans ce sens que l'institut neurosciences de la Timone débouche des moyens pour réaliser la mise en commun et à disposition des chercheurs des données de ses différentes plateformes de stockage tout en gardant l'anonymat de ces dernières. Ce projet va permettre aux chercheurs d'avoir accès aux données des différentes plateformes notamment aux données anonymisées du même patient.

Dans les lignes qui suivent on va essayer en premier lieu de présenter l'institut neuroscience de la Timone tout en mettant l'accent sur ses activités, ses partenaires.., ensuite prendre connaissance des différentes plateformes qu'il utilise pour le stockage de ses données ainsi que la problématique soulevée, après cela on passera à l'étude de la solution et son déploiement pour la mise en commun des données et enfin terminer par l'anonymisation des données qui est une étape essentielle du projet.

2 Présentation de L'INT

2.1 Historique

L'INT a été créé officiellement le 1er janvier . L'INT est le fruit d'une restructuration majeure des neurosciences initiée en 2000 par le centre national de la recherche scientifique (CNRS) et les trois universités d'AIX Marseille.
L'INT a pour objectif d'explorer, comprendre et modéliser le fonctionnement du cerveau. Au cours de son développement, il s'est fixé de nouveaux objectifs à savoir développer des recherches de haut niveau en neurosciences fondamentales, du cellulaire au **cognitif*** et de faire tomber les frontières entre la recherche fondamentale et la recherche **clinique*** c'est-à-dire toute recherche menée sur l'homme. La recherche fondamentale est définie comme étant des Travaux expérimentaux ou théoriques entrepris essentiellement en vue d'acquérir de nouvelles connaissances. L'INT se dote de 4500 m² de bâtiments au sein du campus de la faculté de médecine d'Aix Marseille à la Timone afin de favoriser les interactions entre les équipes de recherche et le regroupement des équipements scientifiques au seins de grands plateaux techniques mutualisés.

Il accueille onze équipes de recherche entre autres la SpiCCI qui est spécialisée dans la moelle épinière et interfaces avec le liquide cérébro-spinal, l'ImaPath qui évolue dans « l'imagerie in vivo » des interactions cellulaires dans les pathologies du système nerveux central et d'autres équipes de recherche chacune dans un domaine bien défini. A côté de ses équipes de recherche, l'INT dispose aussi des entités qu'on appelle des plateformes technologiques qui ont pour but de proposer des services qui permettent aux équipes de recherche de mener à bien leurs travaux. Parmi ses dernières le NIT (Neuroinformatics and Information Technology) l'ancienne CRISE (Cellule réseau et Informatique), plateforme dans laquelle j'ai réalisé ce stage, occupe une place très importante au sein de l'INT. Le NIT s'occupe de tout ce qui est service réseau, informatique, Stockage de données et calculs scientifiques. Il a à sa tête monsieur Olivier COULON (Chercheur), responsable Scientifique et monsieur Sylvain Takerkart (ingénieur de recherche) responsable opérationnel qui dirigent une équipe dont les membres sont LABEJOF Jimmy (administrateur système et réseaux CNRS) qui s'occupe de tout ce qui est administration système et réseau, il est aidé par ROUQUET Pascal (Technicien) gestionnaire de parc informatique, ensuite David MEUNIER (Ingénieur de recherche en bio-informatique), BACHAR Dipankar (Ingénieur de recherche en bio-informatique) qui m'a encadré durant toute la durée de mon stage. (voir organigramme **figure1)**

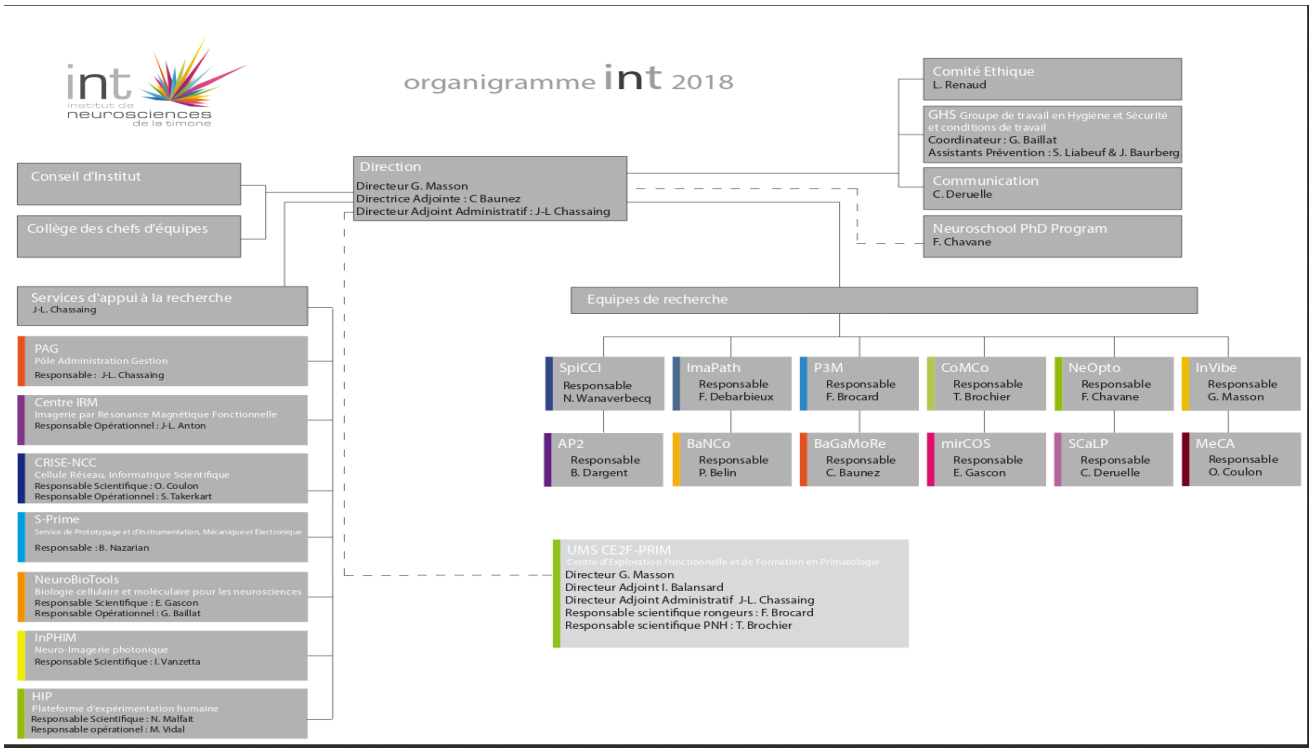

Figure 1 : Organigramme de l'INT

2.2 Activités

L'INT s'engage dans diverses activités toujours dans le cadre des neurosciences. Il donne les moyens à ses chercheurs et cliniciens pour explorer, comprendre et modéliser le fonctionnement normal et pathologique du cerveau et de la moelle épinière avec une approche intégrée allant du neurone aux comportements grâce à la neurophysiologie (étude des fonctions du système nerveux) et l'imagerie multi-échelle. Il les aide aussi à élucider le code neuronal c'est à dire d'essayer de comprendre comment la dynamique des grands et petits réseaux neuronaux explique les comportements les plus intégrés comme la perception, les émotions ou encore le contrôle des mouvements de la main ou des yeux. Essayer aussi de comprendre comment les perturbations ou la mort des neurones provoquent des troubles neurologiques ou psychiatriques.

Et enfin l'INT forme des jeunes scientifiques et cliniciens sur la recherche dans le domaine des neurosciences. Il est aidé sur ses activités par ses différentes plateformes technologiques dont le NIT qui joue un rôle très important au sein de l'INT.

Le NIT (Neuroinformatics and Information Technology) est la plateforme qui s'occupe de tout ce qui est informatique au service de l'INT. Il est structuré en deux cellules : une cellule « calculs scientifiques et données » et une cellule « infrastructure système, réseau et calculs haute performance ». Il a pour mission la gestion du parc d'ordinateurs de bureau des membres de l'INT afin de garantir le bon fonctionnement de ses derniers ainsi que les ordinateurs qui pilotent les postes expérimentaux.

Le NIT met à disposition des équipes de recherches des outils puissants de calcul pour les neurosciences, garantit la disponibilité et les performances du système d'information au sein de l'institut et la sécurité de l'infrastructure. Il a aussi la charge de la mise en œuvre d'un support aux projets de l'institut dans le domaine de l'organisation et du traitement des données : traitement du signal et des images, bio-informatique, apprentissage automatique. Il garantit aussi la bonne gestion des données en assurant leur stockage et leur sécurité.

2.3 Partenaires

L'institut des neurosciences de la Timone est en partenariat avec beaucoup d'entreprises et laboratoires de recherche. Entre autres l'institut national de la santé et de la recherche médicale (l'INSERM) qui a d'ailleurs participé à la rénovation de l'INT entre 2008 et 2011 avec la CNRS (Centre national de la recherche scientifique), la région PACA, le conseil général des Bouches du Rhône, la ville de Marseille et le fond européens pour le développement régional (FERDER). Ce projet a été conduit par l'université de la méditerranée. L'INT est aussi en partenariat avec Orange, Amidex etc.

3 Présentation du sujet de stage

3.1 Objectif du stage

L'Institut Neurosciences de la Timone dispose actuellement de deux logiciels de bases de données. Un logiciel de base de données REDCap (Research Electronic Data Capture) où on stocke les données cliniques et un autre logiciel de base de données XNAT pour les données d'imagerie. Ces deux logiciels de bases de données servent à stocker les données et **métadonnées*** issues des expérimentations au cours des différentes phases de recherche.

Durant mes dix semaines à l'INT, j'ai travaillé en premier lieu sur une solution de « cross-plateformes ». Ma mission était de mettre en place une solution qui permet d'avoir une vue globale sur l'ensemble des données des deux logiciels REDCap et XNAT. En second lieu j'ai élaboré une solution de protection de données personnelles par la génération d'identifiant global unique (GUID) qui va permettre d'identifier les données d'un sujet sans pour autant avoir recours à ses informations personnelles.

3.2 Aperçu sur les logiciels de bases de données

3.2.1 Research Electronic Data CaptureREDCap

REDCap (Research Electronic Data Capture) est un outil de collecte de données développé en 2004 par des informaticiens de l'université de Vanderbilt. Il dispose d'une interface web conviviale (figure 2) permettant aux chercheurs d'avoir le contrôle total sur leurs travaux sans aucune connaissance de base en informatique. Les chercheurs peuvent gérer directement leurs projets comme ils le souhaitent. La base de données REDCap est destinée spécialement aux données cliniques. L'ensemble de ses données sont stockées dans une base MySQL. De nombreuses bibliothèques médicales ont commencé à utiliser REDCap pour évaluer et saisir des données pour des projets.

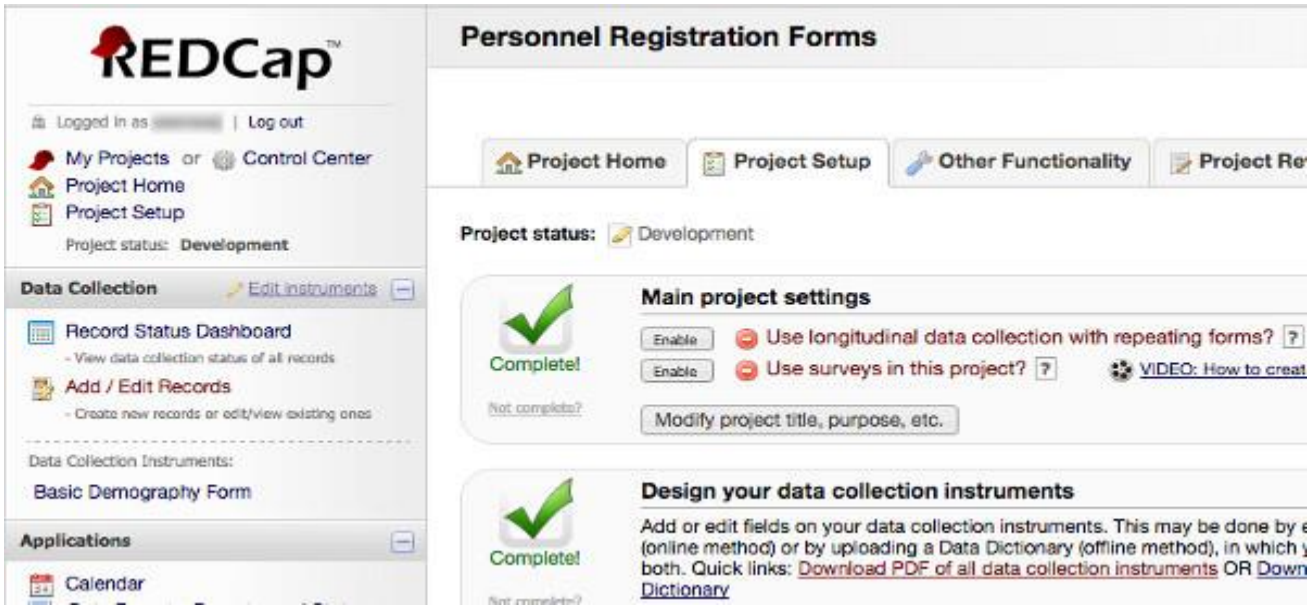

Figure 2 : interface utilisateur de REDCap

3.2.2 XNAT

XNAT est un logiciel « open-source », c'est-à-dire il nous donne la possibilité de développer nos propres fonctionnalités afin de l'améliorer, dédié à l'imagerie qui se définit comme étant l'ensemble des moyens d'acquisition et de restitution d'image du corps humain à partir de différents phénomènes physiques. Il est développé par une équipe de développeurs du laboratoire «NRG lab» de l'université de Washington. Les données de XNAT sont stockées dans une base de données PostgreSQL. XNAT dispose d'une interface très développée avec différentes fonctionnalités (figure 3).

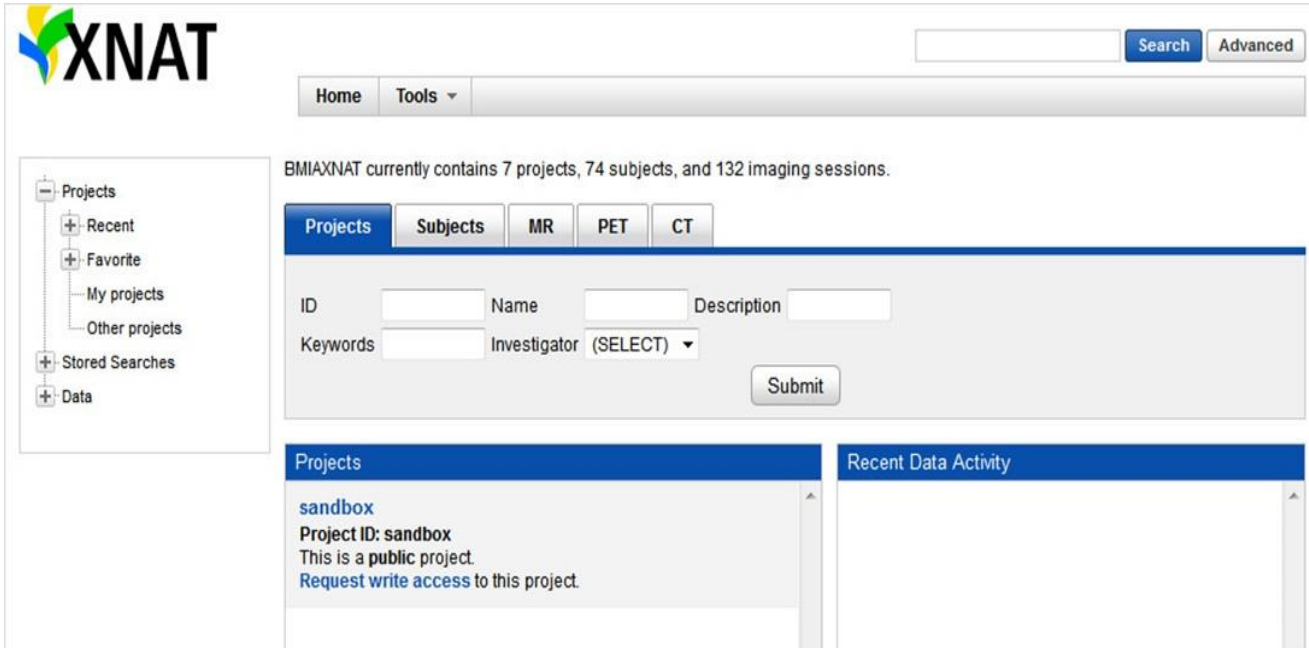

Figure 3 : Interface utilisateur XNAT

3.3 Problématique

Pour mener à bien des recherches dans le domaine des neurosciences, les équipes de recherche ont besoin d'établir des corrélations entre éléments très différents contenus dans le dossier d'un patient stocké dans les bases de données REDCap et XNAT. Le stockage des données rend difficile la mise en place de filtres d'analyse du dossier du patient. Le chercheur est obligé de faire cela manuellement pour chaque patient, parcourir le dossier manuellement. L'étude doit se faire généralement sur plusieurs dossiers, ce qui devient quasi impossible. Le besoin exprimé est de pouvoir réunir toutes les données contenues dans les différentes bases de données.

Lors d'une réunion avec les membres de l'INT, je leur ai proposé un logiciel nommé Elasticsearch, recherche élastique comme son nom l'indique, qui est un moteur de recherche reposant sur une base de données non SQL (Structured Query Langage). Ce logiciel sera couplé avec une technologie qu'on appelle Logstash qui permet de faire le lien entre la base de données elasticsearch et les différents logiciels de bases de données que sont REDCap et XNAT. Chaque table des deux bases de données sera récupérée par Logstash et passée à elasticsearch sous forme d'index. Enfin une interface graphique Kibana proposant diverses fonctionnalités sera mise en place pour visionner les données sur elasticsearch. Ces trois logiciels forment une solution appelée «la suite ELK (Elasticsearch Logstash Kibana » qui est une solution logicielle développée pour intégrer des données dans une base de données elasticsearch. Cette solution présente des aspects particuliers et nécessite une étude détaillée avant de passer à sa mise en place.

4 Présentation du travail réalisé

Partie I : Etude de la solution ELK et mise en place

4.1 Introduction à Elasticsearch

Elasticsearch est une évolution du projet apache Lucene, projet destiné à créer de puissants moteurs de recherche orientés texte, créé par Shay Bannon.
Shay Bannon est le président du conseil et chef de la direction Elastic NV et administrateur et chef de la technologie chez Elasticsearch, une filiale de Elastic NV. Aujourd'hui ce moteur Lucene est devenu un sous ensemble des fonctionnalités d'elasticsearch.

Elasticsearch est un logiciel open source sous licence logiciel libre apache, c'est un puissant moteur de recherche capable de stocker une grande quantité de données que l'on peut interroger à temps réel. Un moteur de recherche est une solution technologique capable d'intégrer un grand nombre de données permettant aux utilisateurs d'y accéder en une fraction de temps. Elasticsearch est aujourd'hui utilisé par de nombreux start-ups web grâce à sa stabilité et son élasticité. Par exemple Xing, un réseau social professionnel de 14 millions de membres, l'intègre déjà pour une recherche en temps réel afin de satisfaire ses utilisateurs. Netflix aussi, une entreprise américaine implémentée à travers le monde proposant des films et séries télévisées sur internet, fait recours aux services d'elasticsearch pour l'analyse de ses données, pour l'étude de son journal informatique et des erreurs. Étant une base de données qui n'utilise pas le langage SQL, elasticsearch repose sur une technique d'indexation des données sous forme de documents textes et l'appel à ses données se fait à travers un langage dénommé REST (Représentationnal State Transfer) qui offre d'intéressantes possibilités d'interrogation.

4.1.1 Architecture Elasticsearch

Elasticsearch stocke les données sous forme d'index. Un **index*** est un espace de nom logique semblable à une base de données. Les données qui se trouvent dans un index elasticsearch se présentent sous forme de documents de format JSON (JavaScript Object Notation). De plus chaque index possède également des types qui correspondent à des tables dans une base de données et qui permettent de partitionner logiquement les données dans un index en un ou plusieurs fragments (Shards) qui résident dans des nœuds (instances elasticsearch) différents. Pour chaque demande de recherche, tous les segments (constituant d'un fragment) d'un index sont recherchés et chaque segment consomme des cycles du processeur. Cela veut dire que plus le nombre de segments est élevé plus les performances de la recherche seront fiables.

Elasticsearch fonctionne en plusieurs clusters (figure 4), un ensemble d'instances elasticsearch (appelées Nœuds) qui communiquent entre eux pour lire et écrire dans un index. Il existe trois nœuds différents. D'une part le nœuds maître (Master Node) qui est responsable de la coordination des taches du cluster telles que la distribution des fragments entre les nœuds. D'autre part, le nœuds de données qui stocke les données sous forme de fragments (Shards) et effectue des actions liées à l'indexation, à la recherche et l'agrégation des données. Enfin, le nœud client dont les propriétés « Node. Master » et « Node. Data » sont définies sur « false » et agit en tant qu'équilibreur de charge facilitant le routage des requêtes d'indexation et de recherche. Donc, cela peut ne pas être nécessaire pour un cluster.

Les données elasticsearch sont stockées sous forme de documents JSON (JavaScript Object Notation) dans des partitions de l'index appelées « Primary Shards » ou fragments primaires. Par défaut l'index est découpé en cinq « Primary Shards » qui dispose chacun d'une copie appelée « Secondary Shards » ou fragments secondaires par mesure de sécurité des données. La recherche de données sur elasticsearch se fait à l'aide d'un langage puissant appelé REST (Representationnal State Transfer).

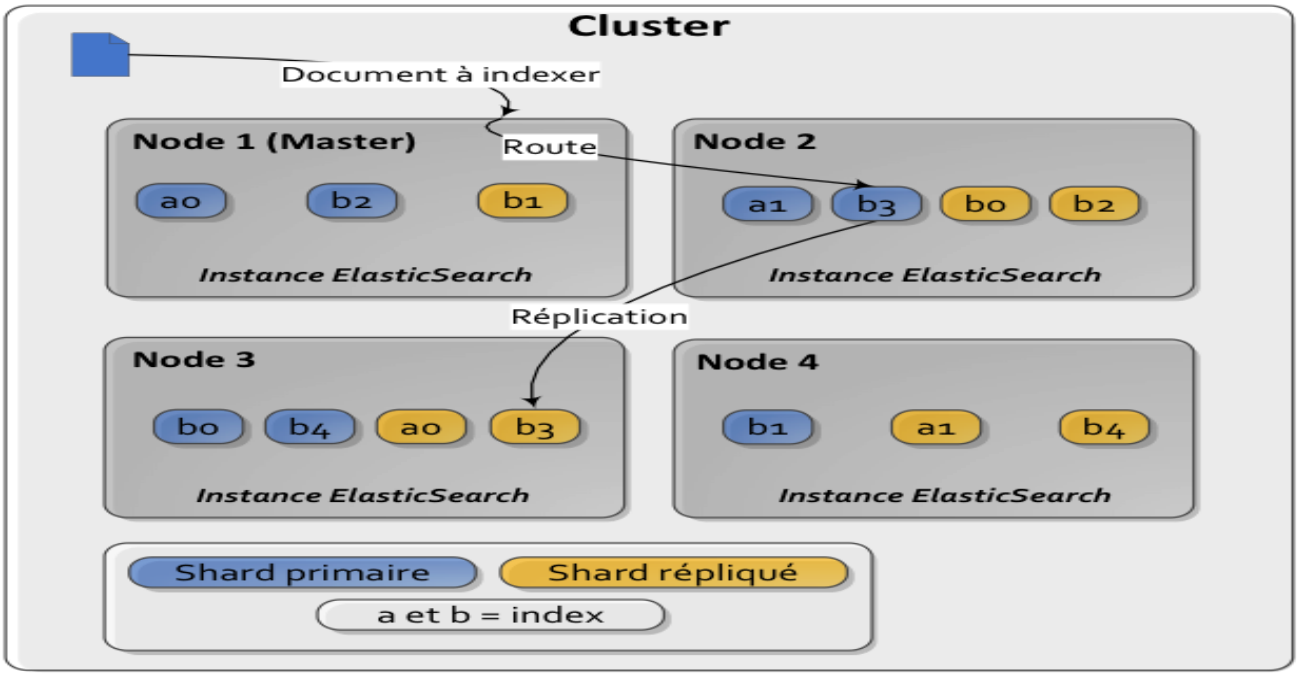

Figure 4 : Cluster elasticsearch

4.1.2 API REST (Representationnal State Transfer)

Une API (Application Programming Interface) est la partie du programme qu'on expose officiellement au monde extérieur pour manipuler celui-ci. Elle regroupe un ensemble de fonctions ou méthodes permettant d'entrer des données, de les modifier ou de les récupérer. L'API REST est sans état la communication entre le client et le serveur ne dépend d'un quelconque contexte du serveur. Ainsi chaque requête contient toutes les informations nécessaires à son traitement. REST est un protocole d'accès aux services web développé par Microsoft. Elle présente beaucoup de similitudes avec le protocole http (HyperText Transfer Protocole) qui signifie littéralement un protocole transfert de texte.

L'API REST présente quatre méthodes qui permettent de lire ou écrire (figure 5). Ces méthodes sont utilisées généralement par le protocole http. La méthode « POST » permet de créer un index dans la base de données elasticsearch en lui donnant un nom. Pour écrire sur cet index on utilise la méthode « PUT ». Ensuite pour obtenir le contenu de l'index on utilise la méthode « GET ». Enfin la méthode « DELETE » permet de supprimer l'index.

A côté de ces méthodes, REST se base sur les URI (Uniforme Resource Identifier) qui permettent d'identifier l'endroit où se trouve le document à lire ou à écrire. Ainsi chaque application utilisant le protocole REST pour les appels à ses ressources se doit de construire ses URI de manière précise.

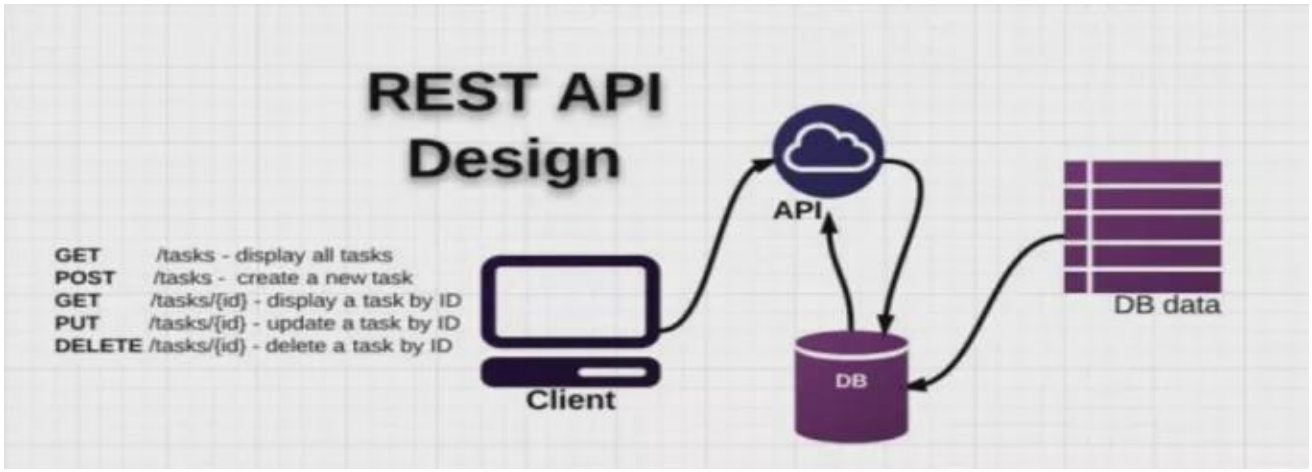

Figure 5 : Fonctionnement de l'API REST

4.1.3 Configuration elasticsearch

Même en étant puissant, elasticsearch est facile à mettre en place (Annexe I & II). Sa configuration repose sur un fichier d'extension YAML (Yet another Marcket Langage). Ce fichier elasticsearch.yml (figure 6) qui sera chargé au démarrage contient des informations essentielles au fonctionnement du logiciel notamment le numéro de port d'écoute 9200 par défaut, l'adresse d'écoute, le nombre de nœuds à utiliser (trois par défaut) etc.

```
  GNU nano 2.7.4    Fichier : /etc/elasticsearch/elasticsearch.yml

#
# Elasticsearch performs poorly when the system is swapping the memory.
#
# ---------------------------------- Network -----------------------------------
#
# Set the bind address to a specific IP (IPv4 or IPv6):
#
network.host: 10.164.4.26
#
# Set a custom port for HTTP:
#
http.port: 9200
#
# For more information, consult the network module documentation.
#
# --------------------------------- Discovery ----------------------------------
#
# Pass an initial list of hosts to perform discovery when new node is started:
# The default list of hosts is ["127.0.0.1", "[::1]"]
```

Figure 6: fichier elasticsearch.yml

Étant donné qu'Elasticsearch est développé en java, il nécessite donc une machine virtuelle java pour fonctionner. La mémoire allouée à la machine virtuelle est définie dans le fichier jvm.options où on indique la mémoire minimale utilisée mais aussi la mémoire maximale que dois utiliser la machine virtuelle java.

```
  GNU nano 2.7.4       Fichier : /etc/elasticsearch/jvm.options

## -Xms4g
## -Xmx4g
##
## See https://www.elastic.co/guide/en/elasticsearch/reference/current/heap-siz$
## for more information
##
################################################################

# Xms represents the initial size of total heap space
# Xmx represents the maximum size of total heap space

-Xms2g
-Xmx2g

################################################################
## Expert settings
################################################################
```

Figure 7: Allocation de mémoire de la machine virtuelle java

4.2 Introduction à Logstash

Logstash est un composant de la suite ELK : elasticsearch Logstash et Kibana. Pour rappel, la suite ELK a pour objectif de faciliter l'injection de données dans elasticsearch. Logstash est un logiciel open source coté serveur permettant de collecter, traiter et transférer des données. Il

fonctionne à base de **Plugins***. La collecte de données s'effectue via des plugins d'entrée qui ont pour rôle de récupérer les données auprès de différentes sources qui peuvent être des fichiers, des bases de données, des systèmes de messagerie etc. Une fois que les données sont collectées par les plugins d'entrée, celles-ci sont traitées par différents filtres afin de garantir leur bon format et prêtes à être acheminées vers les plugins de sortie. Enfin Logstash transmet les données aux plugins de sortie qui a leur tour les transmettent à des programmes externes très variés notamment elasticsearch ce qui lui permet d'être un outil très polyvalent, c'est un outil très utilisé pour l'alimentation de données en temps réel. Tout l'intérêt de Logstash réside sur sa compatibilité avec différentes sources de données.

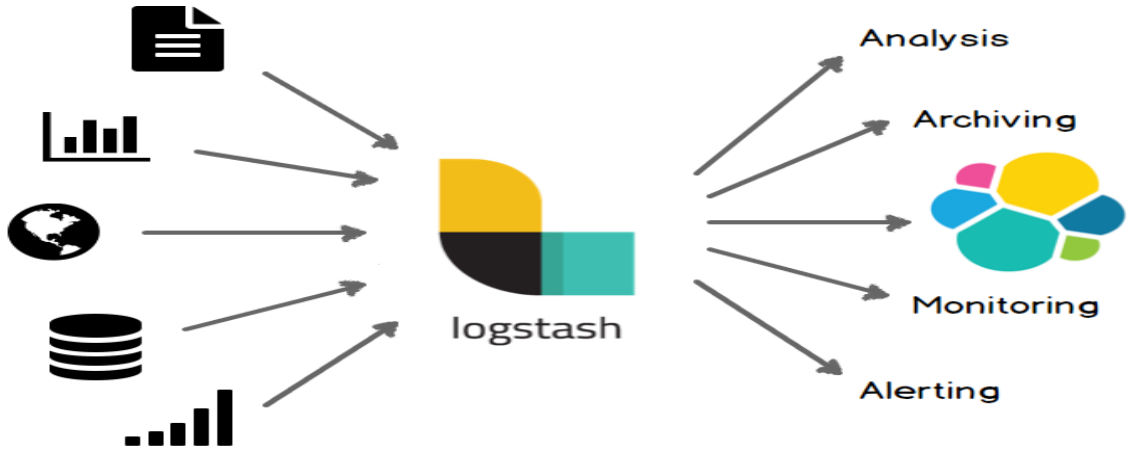

Figure 8: diversité des sources de Logstash

4.2.1 Fonctionnement de Logstash

Logstash traite les données de façon très organisée et optimale. Le traitement est géré par un ou plusieurs pipelines. Par définition un pipeline est un tunnel servant à transporter des données depuis une source vers une destination. Chaque pipeline contient un ou plusieurs plugins (extensions) qui sont chargés de récupérer les données, les soumettre à des filtres ensuite de les transporter à un grand nombre d'architectures différentes (figure 9).

Avant la version six de Logstash, chaque pipeline prend en charge un seul ficher de configuration où on définit la source des données (bloc input), le filtrage (bloc filter) utilisé et la destination (bloc output). Mais à partir de la version six de Logstash, il est maintenant possible de définir un seul pipeline pour plusieurs fichiers de configuration en déclarant au préalable le chemin de ces fichiers dans un fichier pipelines.yml dans le dossier de configuration de Logstash. L'utilisation d'un pipeline par fichier de configuration permet de ne pas interrompre les traitements si toute fois une sortie est bloquée. Elle permet aussi d'avoir les paramètres de performance nécessaires en fonction de la quantité de données traitée.

Après récupération des données par les plugins d'entrée, chaque pipeline dispose d'une file d'attente interne appelée queue qui permet le stockage des données avant d'être traitées par micro-lots. Cette file d'attente est petite par défaut, il est donc important de la modifier afin d'améliorer la fiabilité de Logstash.

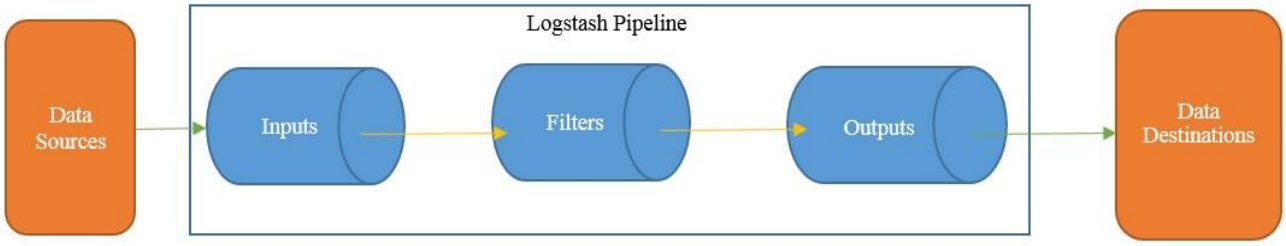

Figure 9: pipeline Logstash

Logstash est connu pour sa compatibilité avec de nombreux types d'architectures qui peuvent recevoir les données après ses différentes phases de traitement. Mais cette compatibilité est possible grâce à sa variété de filtres qui assurent la bonne structure des données qui seront transmises à l'architecture cible. Ce formatage des données permet de les analyser et de les visualiser plus facilement. Un grand nombre de plugins de filtre est disponible mais le plugin qui va être utilisé dépendra bien sûr du type de donnée qui va être traité mais aussi de l'architecture cible. Parmi ses différents plugins de filtre, le filtre GROK reste le plus utilisé.

4.2.2 Le filtre GROK

Le filtre GROK est le filtre le plus utilisé pour analyser des données. C'est un excellent moyen d'analyse de données non structurées en les rendant structurées et interrogeables. Il est généralement utilisé pour analyser les journaux Syslog (protocole définissant un service de journaux et d'événement d'un système informatique), Apache et autres journaux de serveurs web, les journaux MySQL, et en général tout format de journal.

Le filtre GROK fonctionne en combinant des modèles de texte dans quelque chose qui correspond aux journaux. Il remplace une « SYNTAX » par exemple « 10.20.3.4» (qui est un adresse IPv4) par une «SEMANTIC» par exemple «IP».

4.2.3 Configuration de Logstash

La configuration principale de Logstash (annexe I & II) est définie dans un fichier YAML (Yet Another Markup Langage) dénommé logstash.yml qui est chargé au démarrage de ce dernier. Ce fichier contient la configuration des pipelines utilisées notamment la taille d'un pipeline, le délai, le nombre de processeurs. Il contient aussi l'ensemble des modules chargés, la configuration de la mémoire de la queue etc... Pour rappel la queue est là où sont stockées les données Logstash avant d'être traitées par micro-lots.

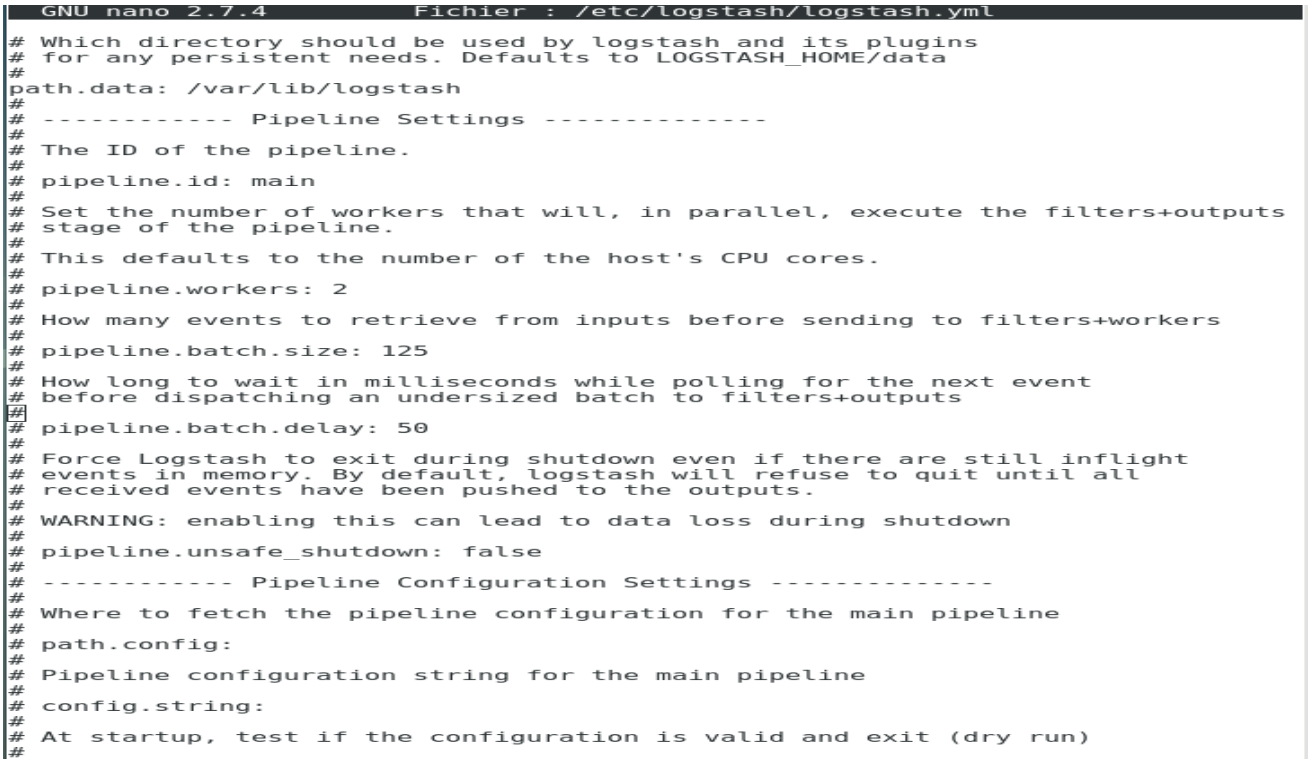

Figure 10: fichier de configuration Logstash

A côté de la configuration du serveur Logstash, il y a aussi la configuration de ses pipelines. Les pipelines Logstash disposent des fichiers de configuration qui spécifient la source des données à traiter (bloc input), le filtre utilisé (bloc Filter) mais également la destination où seront envoyées les données traitées (bloc output). A partir de la version six de Logstash, chaque pipeline peut avoir un ou plusieurs fichiers de configuration dont le chemin sera défini dans un fichier pipelines.yml (figure 11) du dossier de configuration de Logstash. Ce fichier indique à un pipeline par son identifiant le ou les fichiers de configuration qu'il doit utiliser. Le nombre de pipeline utilisable est illimité, il devient donc préférable d'utiliser un pipeline par fichier de configuration si les données sont différentes, car pour un pipeline donné Logstash lis toutes les entrées de données pour les mettre sur la sortie. Pour des versions de logstash antérieures à la version six, le chemin du fichier de configuration du pipeline est directement défini dans le fichier logstash.yml dans la partie configuration pipeline.

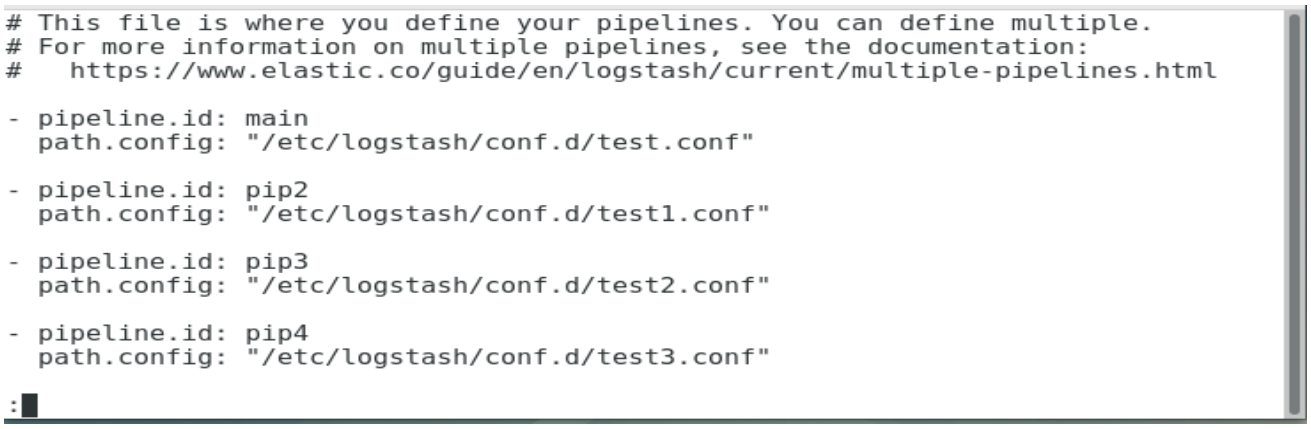

Figure 11 : Fichier pipelines.yml

De mème qu'elasticsearch, logstash est aussi développé en java donc requiert une machine virtuelle java pour fonctionner. La quantité de mémoire qui sera allouée à la machine virtuelle est définie dans le fichier jvm.options dans le dossier de configuration de logstash où on spécifie la quantité minimale et maximale de mémoire allouée à la machine virtuelle java. Pour le bon fonctionnement de logstash, cette quantité maximale de mémoire ne doit pas dépasser la moitié de celle du serveur où sera installé Logstash.

```
## JVM configuration

# Xms represents the initial size of total heap space
# Xmx represents the maximum size of total heap space

-Xms2g
-Xmx2g

################################################################
## Expert settings
################################################################
##
## All settings below this section are considered
## expert settings. Don't tamper with them unless
## you understand what you are doing
##
################################################################
:
```

Figure 12: fichier de configuration de la machine virtuelle java pour Logstash

4.3 Indexation de données des bases de données XNAT et REDCap dans Elasticsearch à l'aide de logstash

Les données des deux bases de données sont organisées sur plusieurs tables d'une base MySQL (REDCap) et PostgreSQL (XNAT). Une table correspond à un ensemble de données organisées sous forme d'un tableau où les colonnes correspondent à des catégories d'informations et les lignes correspondent à des enregistrements. Dans XNAT le nombre de table qui stockent les données correspond à quatre cents quarante et sept tables tandis que dans REDCap il correspond à seulement quatre tables exceptées les tables stockant les données utilisateurs à savoir les données d'authentifications.

La récupération des données de chaque table sera assurée par un plugin d'entrée de logstash appelé JBDC qui est une API de la plateforme Java et un pilote permettant de connecter le plugin aux bases de données, chaque base de données nécessite un pilote spécifique pour pouvoir accepter l'importation de ses données. Par définition un pilote est un programme informatique permettant à un autre programme d'interagir avec un périphérique. Une fois que les pilotes sont installés, on assigne à chaque table des bases de données un fichier de configuration (pipeline). Le chemin de chaque fichier de configuration est indiqué à un pipeline dans le fichier pipelines.yml dans le dossier de configuration de logstash.

4.3.1 Récupération et indexation des tables de XNAT

Chaque table dans la base de données XNAT est relié à un fichier de configuration (pipeline) de logstash (quatre cent quarante et sept fichiers de configuration). Chaque fichier est chargé de récupérer les données de la table et de l'indexer dans elasticsearch. Le nom de l'index sera le même que le nom de la table sur XNAT.

```
input {
  jdbc {
    jdbc_driver_library => "/usr/share/logstash/vendor/jdbc-postgres/postgresql-42.2.5.jre7.jar"
    jdbc_driver_class => "org.postgresql.Driver"
    jdbc_connection_string => "jdbc:postgresql://10.164.4.26:5432/xnat"
    jdbc_user => "postgres"
    jdbc_password => "xnat"
    statement => "SELECT * FROM xnat_experimentdata"  }
}
output {
elasticsearch {
    hosts => ["http://10.164.4.26:9200"]
    index => "xnat_experimentdata"
    user => "elastic"
   password => "adsl458093"
  }
}
:
```

Figure 13: fichier de configuration pour indexer une table XNAT

Dans le bloc « input » (figure 13) on déclare les paramètres nécessaires à l'accès aux données de la base de données PostgreSQL (XNAT). La ligne jdbc_driver_library indique à Logstash l'emplacement du pilote compatible avec PostgreSQL. La ligne jdbc_connection_string permet d'accéder au serveur de base de données en indiquant son adresse IPv4, le numéro de port utilisé et le nom de la base de données. « Statement » spécifie la requête lancée dans la base de données pour récupérer les données de la table spécifiées. Après récupération des données grâce au bloc « input », celles-ci sont passées au bloc output. Le bloc « output » correspond à elasticsearch puisque les données seront transmises à elasticsearch. Dans notre cas l'utilisation d'un filtre n'est pas nécessaire car elasticsearch supporte le filtre par défaut de logstash. La ligne « hosts » du bloc de sortie indique l'adresse IPv4 du serveur elasticsearch et le numéro de port utilisé. Tandis que la ligne « index » indique le nom qu'on va donner à ces données sur elasticsearch qui sont stockées sous formes de documents. Pour une identification plus simple des données, le nom de l'index dans elasticsearch correspond au nom de la table dans la base de données XNAT.

4.3.2 Récupération et indexation des tables de REDCap

Le procédé d'importation des données est le même que celui de XNAT. L'importation des tables repose sur des fichiers de configuration (pipelines). Chaque fichier de configuration est chargé d'importer les données de la table vers la base de données elasticsearch. Au total quatre fichiers de configuration donc quatre index dans elasticsearch dont chaque fichier se configure presque de la manière ci-dessous en fonction du nom de la table (figure 14).

Figure 14: fichier de configuration pour indexer une table REDCap

De même que XNAT on déclare dans un bloc « input » la ligne de connexion à la base de données MySQL (REDCap) mais aussi le fuseau horaire (time zone) qui est important pour le fonctionnement de l'importation de données MySQL. A la ligne jdbc_driver_library on indique le chemin vers le pilote compatible avec MySQL qui doit être téléchargé au préalable. La ligne « statement » permet de spécifier quelles données on veut importer (ici toute la table). Ensuite dans un bloc « output », on spécifie l'adresse et le port d'écoute de la base de données elasticsearch ainsi que le nom de l'index qui va servir d'ailleurs à la visualisation des données. Un paramètre « document_type » pourrait être spécifier pour indiquer à elasticsearch sur quel type de document il est censé travaillé.

4.3.3 La gestion des doublons de données

Après l'indexation des données dans la base de données elasticsearch, on a rencontré un gros problème. Des doublons de données apparaissent dans la base. Ce problème est très grave car cela sature d'une manière très rapide la base données. Afin de régler ce problème, il est important de connaître sa provenance.

En effet, lors de l'indexation, elasticsearch attribue à chaque document un identifiant de manière aléatoire servant lors de la recherche de données. Il peut arriver que l'importation de données dans un pipeline soit interrompue, une partie des données est déjà intégrée et dispose d'un identifiant. Mais lorsque le processus recommence les mêmes données seront injectées à nouveau et disposent d'un nouvel identifiant ce qui crée des doublons de données qui prennent de l'espace de stockage au fur et à mesure.

Pour éviter la présence des doublons de données, il faut attribuer manuellement un identifiant pour chaque document afin que des données portant le même identifiant ne soient pas répétées. Ce fameux identifiant est précisé sur le fichier de configuration de chaque pipeline dans le bloc de sortie. Son argument porte le nom de « document_id ».

4.4 Introduction à Kibana

Elasticsearch, puissant qu'il soit, a besoin d'une interface puissante pour visualiser les données. C'est par là que vienne l'idée de Kibana. Kibana est un logiciel open source sous licence libre apache version 2 développé en JavaScript. Kibana fournit une interface graphique très puissante accessible via un navigateur web pour visualiser les données qui sont indexées dans la base de données

elasticsearch. Il offre diverses fonctionnalités de mise en forme de données notamment des diagrammes, des tableaux, des graphes... il offre la possibilité de mettre son propre tableau de bords mais aussi de travailler sur les données à temps réel.
Kibana dispose de champs de filtre qui permettent de manipuler les données en toute aisance. Kibana donne aussi la possibilité de rajouter des fonctionnalité grâce à sa liste de plugins mais aussi de développé de nouveaux plugins. Il est très facile à configurer et ne demande pas beaucoup de connaissances en informatique.

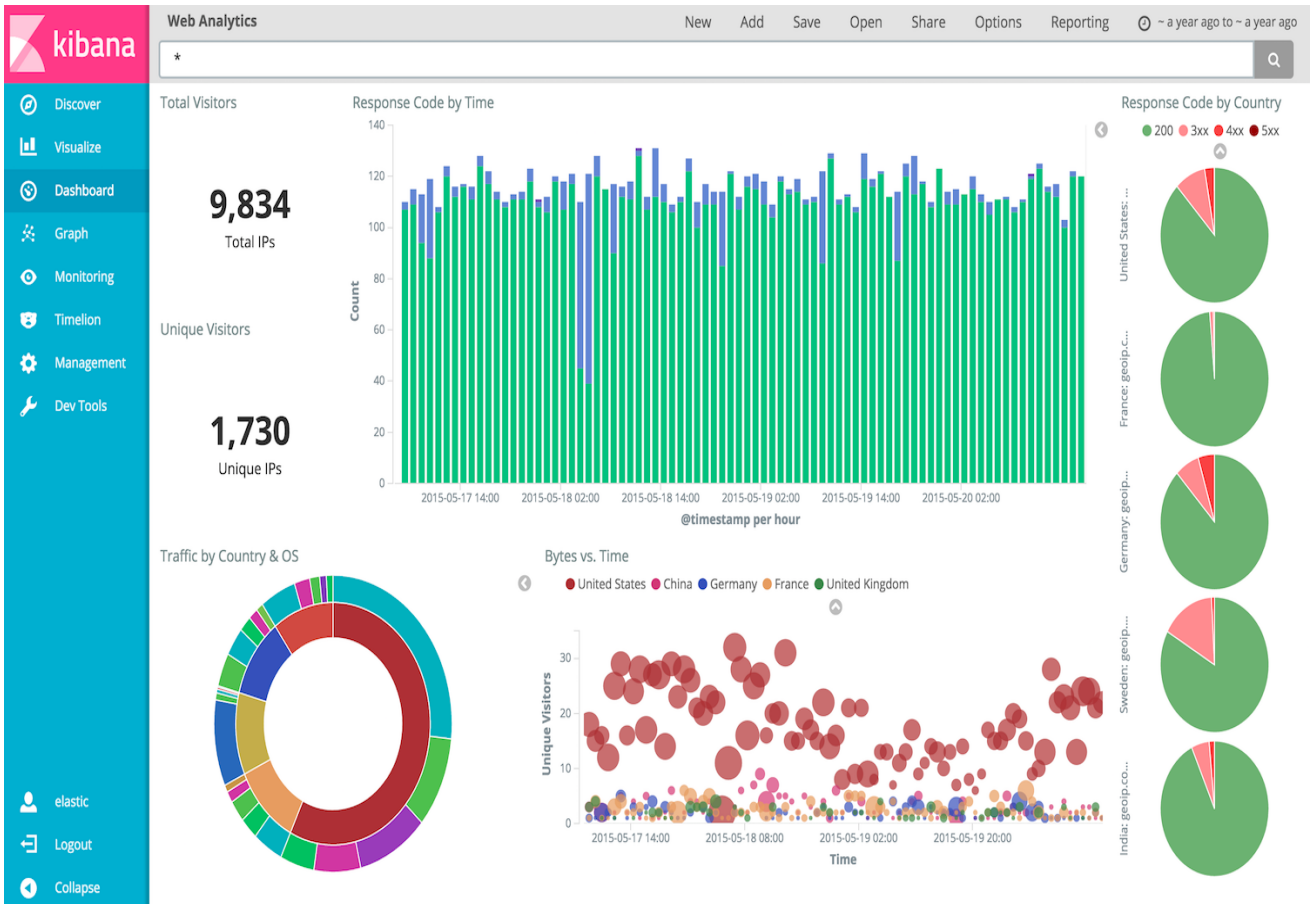

Figure 15: Interface utilisateur Kibana

4.4.1 configuration de Kibana

La configuration de Kibana s'avère très simple et basique. Toute sa configuration repose sur un seul fichier YML (Yet Another Markup Language). Sur ce fichier, est définis le port d'écoute de l'interface Kibana, l'adresse IPv4 sur laquelle l'interface est installée mais également la ligne qui permet de connecter Kibana à Elasticsearch grâce à l'adresse ipv4 et le port du serveur Elasticsearch.
La ligne serveur.port dans le fichier kibana.yml indique le port d'écoute de Kibana. Par défaut c'est 5601 tandis que la ligne server.host indique l'adresse d'écoute du serveur. Pour connecter Kibana à Elasticsearch il est important d'indiquer L'URL de Elasticsearch sur la ligne elasticsearch.url.

```
# Kibana is served by a back end server. This setting specifies the port to use.
server.port: 5601

# Specifies the address to which the Kibana server will bind. IP addresses and h
ost names are both valid values.
# The default is 'localhost', which usually means remote machines will not be ab
le to connect.
# To allow connections from remote users, set this parameter to a non-loopback a
ddress.
server.host: "10.164.4.26"

# Enables you to specify a path to mount Kibana at if you are running behind a p
roxy. This only affects
# the URLs generated by Kibana, your proxy is expected to remove the basePath va
lue before forwarding requests
# to Kibana. This setting cannot end in a slash.
#server.basePath: ""

# The maximum payload size in bytes for incoming server requests.
#server.maxPayloadBytes: 1048576

# The Kibana server's name.  This is used for display purposes.
#server.name: "your-hostname"

# The URL of the Elasticsearch instance to use for all your queries.
elasticsearch.url: "http://10.164.4.26:9200"
:
```

Figure 16 : fichier de configuration de Kibana

4.4.2 Visualisation des index Elasticsearch sur Kibana

Après que l'ensemble des donnés des deux bases de données soit indexé, il devient quand mème nécessaire de les visualiser et éventuellement de les mettre en formes sous formes de tableaux, de graphes etc. Pour pourvoir visualiser les index créés sur elasticsearch, il faudrait au préalable créer l'index sur Kibana. Elasticsearch réserve à Kibana un index nommé «.Kibana » qui contient toutes les informations concernant les index créés sur Kibana notamment la date de création de l'index, sa description…Cet index est créé automatiquement par elasticsearch.

La création d'index se fait via l'interface management de Kibana sur l'onglet « create index pattern ». Une fois que l'index est créé, Kibana recharge toutes les données de cet index qui étaient stockées dans elasticsearch. A ce stade, les données sont visibles sur l'onglet « Discover ». Chaque index contient un entête qui renseigne l'identifiant de l'index, son nom, le type de document qu'il stocke mais également la date de création de l'index.

Kibana offre la possibilité de faire des filtres sur les index en définissant le champ qu'on veut affichée avec différents arguments possibles. Un outil de développement (dev tools) est intégré à Kibana pour la manipulation des données en mode console. Cet outil de développement utilise le langage REST (Representationnal State Transfer).

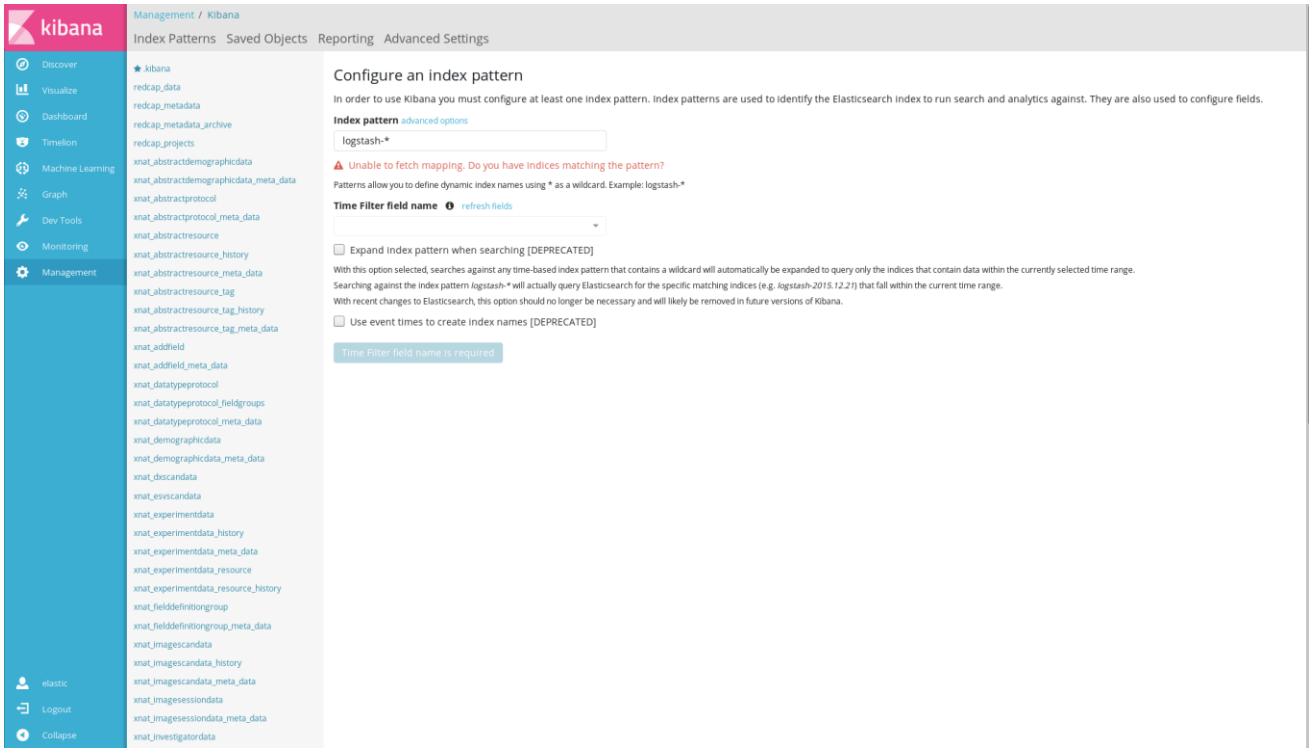

Figure 17 : création d'index Kibana

4.5 Sécurité avec x-pack

La mise en commun des données des deux bases de données REDCap et XNAT pose quelques contraintes de sécurité notamment l'accès par des utilisateurs non autorisés mais aussi le chiffrement de l'importation des données. Pour remédier à ce problème, l'extension x-pack (voir annexe III pour l'installation) de la suite ELK (Elasticsearch Logstash Kibana) garantit une authentification par mot de passe qui permet de restreindre l'accès à elasticsearch mais aussi de sécuriser la communication entre elasticsearch et Kibana et entre elasticsearch et logstash.

L'extension x-pack regroupe des nouvelles fonctionnalités dans un seul package avec la possibilité d'activer ou désactiver la fonctionnalité souhaitée. Sa mise en place est assez simple. Elle propose des fonctionnalités de sécurité, d'alerte et de « machine Learning ». Elle est sous licence d'une année, Il faut devoir la renouveler tous les ans.

Dans le répertoire des extensions elasticsearch et celui de Kibana, on installe juste le package x-pack. Après cela dans le fichier de configuration de Kibana, on renseigne le mot de passe (changeme par défaut) et l'utilisateur (Elastic par défaut) de elasticsearch pour permettre à Kibana de s'authentifier à elasticsearch au démarrage. Ces identifiants seront aussi renseignés dans le bloc « output » des fichiers de configuration des pipelines de logstash. Une fois l'authentification établie, on peut gérer facilement les utilisateurs depuis l'interface Kibana en leur attribuant des droits, ou bien en créer d'autres.

Parmi les droits par défaut que l'on peut attribuer à un utilisateur, il y a le droit de super-utilisateur qui dispose de tous les droits possibles à savoir modifier un index, le supprimer ou créer d'autres utilisateurs. Il y a aussi le droit de simple utilisateur (watcher user) qui ne peut donc modifier un index ni le supprimer.

On peut aussi restreindre la visibilité des données pour certains utilisateurs en les mettant dans un groupe qui n'a pas le droit de visualiser tous les index par exemple et tant d'autres fonctionnalités notamment la « machine learning » qui ne va pas nous intéresser sur ce projet.

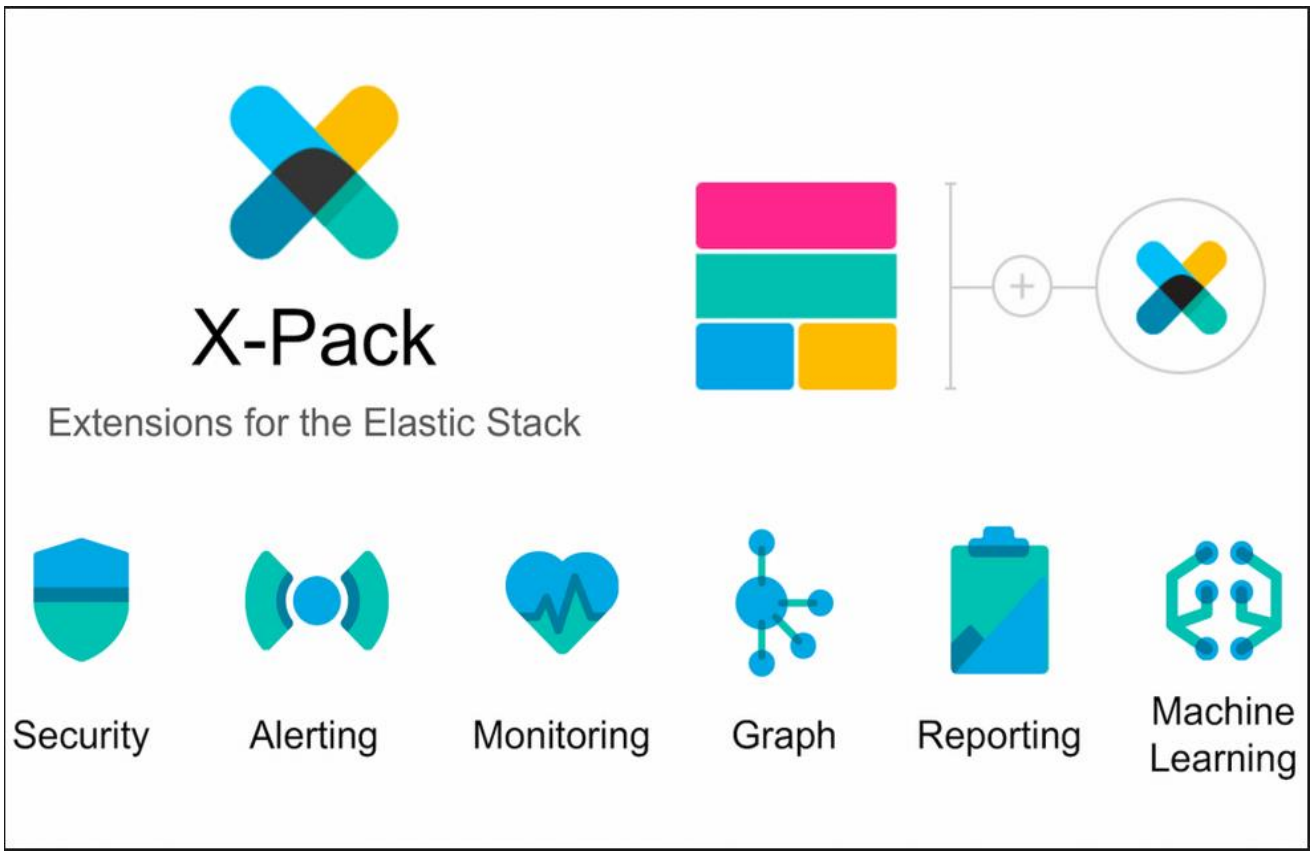

Figure 18: l'extension X-Pack

Bien que l'extension de sécurité x-pack permet de sécuriser totalement la suite ELK, il reste quand même insuffisant dans le cadre de ce projet car on décide de mettre en commun toutes les données donc une question de confidentialité et de protection de données personnelles se pose. Pour remédier à ce problème on décide alors d'anonymiser les données dans les différentes bases de données.

Partie II : Anonymisation des données

4.6 Introduction à l'anonymisation des données

L'institut neuroscience de la Timone dispose des bases de données servant pour le stockage des données issues des expérimentations au cours d'une ou plusieurs recherches en neuroscience. Ces bases de données contiennent une grosse quantité de données médicales de sujets. L'élaboration de la solution de mise en commun des données les rendent donc accessible par un ensemble d'utilisateurs. Par ailleurs ses données sont exposées publiquement et accessibles par tous. La loi sur la protection de données personnelles exige une anonymisation de ces données personnelles. Par définition l'anonymisation contrairement à la pseudonymisation est une opération irréversible. IL consiste à supprimer toutes informations identifiant les données d'un sujet et qui

permettent de remonter à ce dernier. Pour aboutir à l'anonymisation des données dans les bases de données, on a utilisé un algorithme développé de cryptage de données qui génère un identifiant unique nommé GUID (Globally unique identifier) de 256 bits à base du nom du sujet, son prénom, son genre et sa date de naissance. L'algorithme utilisé est le « Secure Hash Algorithm 256 » (SHA256) qui est un algorithme irréversible.

4.7 L 'algorithme SHA-256

L'algorithme SHA-256 (Secure Hash Algorithme) est une puissante fonction de hachage de l'algorithme SHA-2. Cet algorithme a été créé par la NSA (National Security Agency) pour répondre aux problèmes de sécurité posé par le SHA-1. L'algorithme accepte en entrée un message de taille 2^64 bits maximum et produit un code de 256 bits appelé Hash. L'algorithme est basé sur une fonction mathématique non linéaire. En effet il est impossible de remonter aux données avec seulement le Hash généré. Le Hash généré est unique.

L'algorithme SHA-256 (figure 19) se déroule en deux étapes : La phase de prétraitement, le message est complété par bourrage de façon à pouvoir découper le message en blocs de 512 bits c'est à dire le fait de faire en sorte que la taille du message passé en entrée soit compatible avec l'algorithme, s'en suit la phase de calcul du Hash par itération de la fonction de compression sur la suite des blocs obtenus après découpage du message en bloc de 512 bits. Le produit du hachage des informations personnelles va être utilisé pour notre GUID (Globally Unique Identifier).

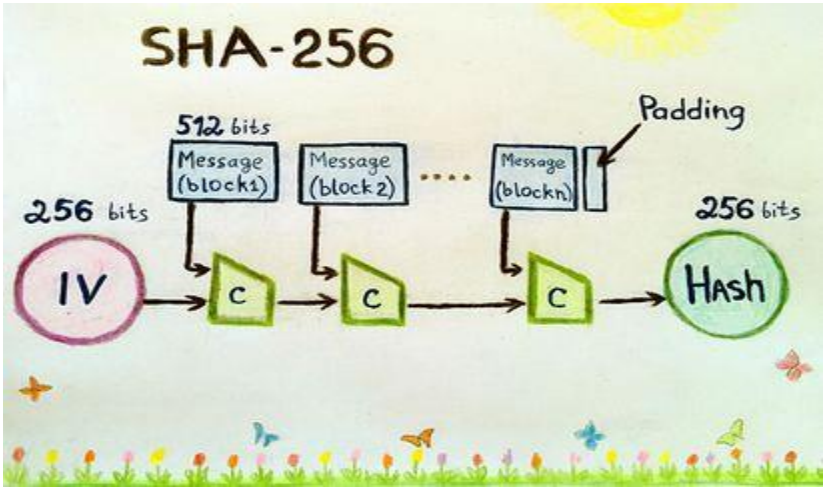

Figure 19: SHA-256 (Secure Hash Algorithme)

4.8 Génération du GUID (Globally Unique Identifier)

Le GUID est un identifiant unique généré à partir des informations personnelles du sujet notamment son nom, son prénom, son genre et sa date de naissance. Le GUID généré lors d'une première participation du sujet à un projet doit être le même lors de la prochaine participation du sujet. La génération du GUID va être faite par un script (annexe IV) rattaché à un formulaire qui recueille les informations personnelles du sujet dès son arrivée au laboratoire. Le formulaire (figure 20) comporte des champs nom, prénom, courriel, date de naissance et genre. Chaque champ du formulaire participant à la génération du GUID doit être confirmé car une fois que l'identifiant est généré il est impossible de revenir en arrière. Chaque champ est contrôlé à ce qu'il soit identique au champ de confirmation afin d'éviter que l'utilisateur se trompe sur la valeur saisie.

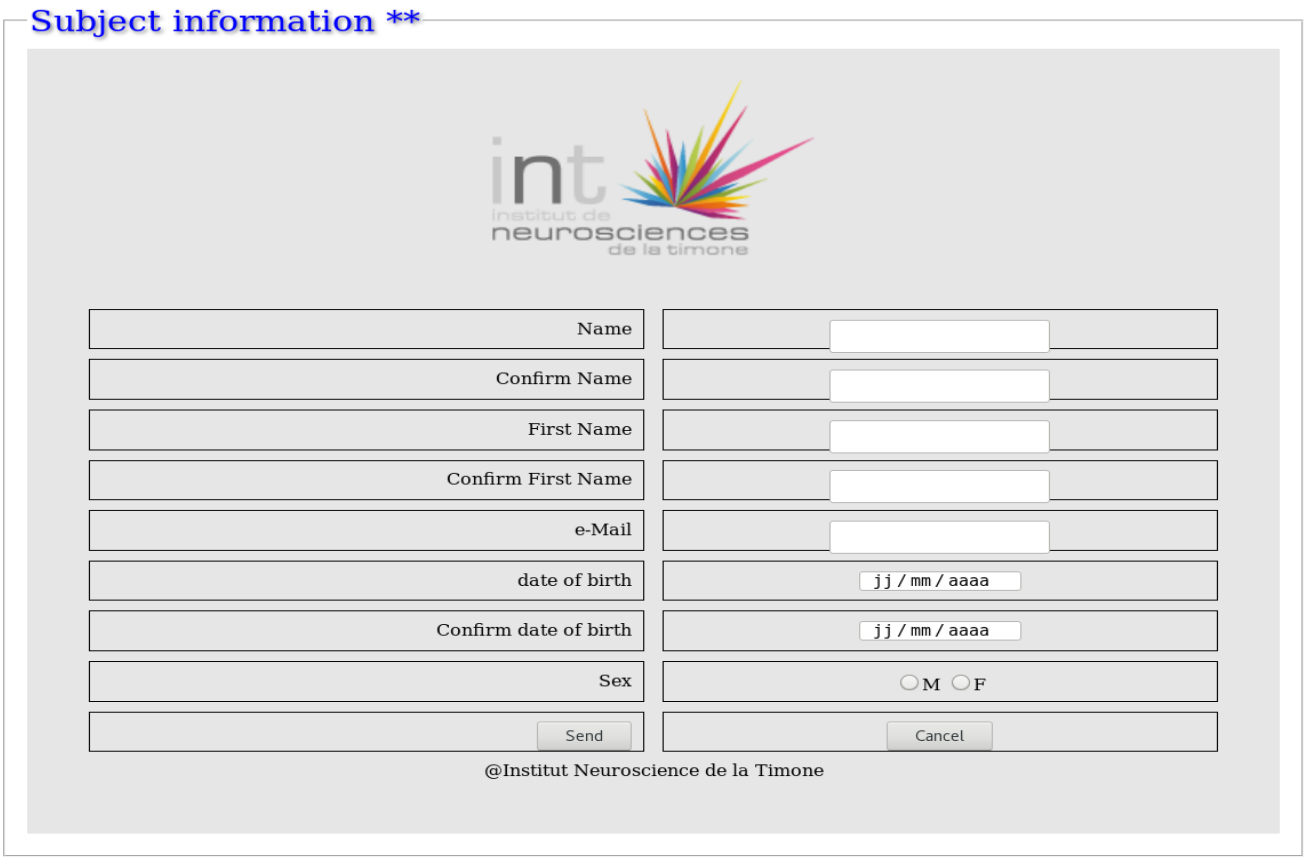

Figure 20: formulaire de génération de GUID

Un script va donc servir des récupérer les informations saisies sur le formulaire au moment de la soumission. À l'aide de la commande UNIX « sha256sum », le script génère un code de 256 bits qui est propre à chaque sujet car les informations saisies sont propres à chaque sujet.

Après la génération du GUID, il va être stocké dans une base de données très sécurisé avec les informations personnelles du sujet et son émail qui pourraient servir lors d'un test longitudinal notamment où le besoin de recontacter le sujet se pose. Cette base de données sera gardée hors du réseau pour des risques de sécurité. Contrairement à la base de données pour le stockage du GUID et les informations personnelles, les informations personnelles du sujet n'apparaîtront pas dans les bases de données XNAT (données d'imagerie) et REDCap (données cliniques). Le seul identifiant possible du sujet est son GUID (Globally Unique Identifier). Ce GUID va alors nous permettre de faire des requêtes « cross-index » puisque les données des bases de données XNAT et REDCap sont stockées sous forme d'index dans la base de données elasticsearch.

5 Conclusion

Durant mes dix semaines de stages à l'institut de neurosciences de la Timone j'ai eu à travailler sur des besoins très importants dans la recherche médicale voire même cruciaux pour l'aboutissement des travaux.

En effet L'ensemble des chercheurs exprime le besoin de corrélation des données d'un patient qui pourraient être stockées dans des bases de données différentes étant donné qu'il peut s'agir de domaines de recherche différents. Cette corrélation pourrait être faite manuellement mais lorsque le nombre de patient devient important, il sera très compliqué voire même impossible d'utiliser le même procédé. Après deux mois et demi de travail, j'ai eu à leur fournir une solution optimale et flexible qui leur permettra d'automatiser cette corrélation de données en mettant en commun les données des différentes bases de données.

La mise en commun de données implique l'accès à toutes les données du patient par toutes les chercheures et tous les chercheurs ce qui viole la loi sur la protection des données médicales. Pour contourner ce problème, on a décidé d'anonymiser les données du patient. J'ai élaboré un algorithme capable d'anonymiser les données en partant des données personnelles du patient pour générer un identifiant unique appelé GUID (Globally Unique Identifier). Cet identifiant va permettre aux chercheurs de travailler sur les données sans pour autant avoir affaires aux données personnelles. Ces technologies vont permettre aux chercheurs de l'institut neurosciences de la Timone de gagner plus de temps pour des résultats plus importants.

Ce stage m'a été d'un apport inestimable tant qu'il m'a permis de développer ma capacité d'autonomie, de force de proposition mais aussi de comprendre le fonctionnement du milieu professionnel afin de se préparer pour mon projet d'avenir. Le sujet qui m 'a été proposé m'a donné l'opportunité de mettre en pratique la plupart des compétences que j'ai acquises lors de mes deux années de formation. Il m'a permis aussi de confirmer le choix de mon projet d'étude en qualité d'ingénieur en administration système et réseaux.

6 Remerciements

Dix semaines de stage qui m'ont permis de comprendre le fonctionnement du milieu professionnel dans sa plus grande diversité mais aussi de mettre en pratique la plus grande partie des compétences que j'ai acquises lors de mes deux années de formations en réseaux et en télécommunications.

Je dois donc de sincères remerciements à l'ensemble de l'équipe de l'institut neurosciences de la Timone qui m'a été d'une aide exceptionnelle pour l'accomplissement de mes missions pendant toute la durée de mon stage en me mettant dans des conditions très favorables. Je tiens aussi à remercier particulièrement monsieur David MEUNIER et monsieur BACHAR Dipankar qui m'ont encadré et formé pendant mes heures à l'INT. Ils m'ont aussi permis de développer mes capacités d'autonomie qui me seront sans doute d'un grand intérêt pour la réussite de mon projet professionnel en qualité d'administrateur système et réseaux.

Tout le mérite revient au corps professorat de l'institut universitaire de technologie d'Aix Marseille université qui m'ont beaucoup aidé lors de ma formation. J'étais un étudiant qui n'avait aucune base en informatique et réseaux mais ils m'ont encadré, formé avec leur programme très solide qui doit inspirer d'autres établissements.

7 Glossaire

INT, Institut Neurosciences de la Timone

RGPD, Règlement Général sur la Protection des Données

NIT, Neuroinformatics and Information Technology

REDCap, Research Electronic Data Capture

CNRS, Centre national de la recherche scientifique

CRISE, Cellule réseau et Informatique

FERDER, fonds européens pour le développement régional

GUID, Globally Unique Identifier

SQL, Structured Query Langage

ELK, Elasticsearch Logstash Kibana

REST, Représentationnal State Transfer

SHA, Secure Hash Algorithm

Cognitif, processus par lesquels un être humain acquiert des connaissances sur son environnement.

Clinique, toute recherche menée sur l'homme

Index, un espace de nom logique semblable à une base de données

Plugins, Extension

8 Table des figures

9 Bibliographie

http://www.int.univ-amu.fr/institut [en ligne] site de l'insitut neurosciences de la Timone, consulté le 09/04/2019

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5764586/ [en ligne] article de « journal of the medical library association JMLA», publié le 02 janvier 2018, consulté 09/04/2019

https://www.project-redcap.org/ [en ligne] site officiel de REDCap, consulté dernièrement le 12/05/2019)

https://www.xnat.org/ [en ligne] site officiel de XNAT, consulté dernièrement le 12/04/2019

https://trait.health-ri.nl/trait-tools/xnat [en ligne], site consulté le 12/04/2019

https://hal.inria.fr/hal-00782339/ [en ligne], article du service neurospin de inria saclay , mise à jour 07/03/2019, consulté le 13/04/2019

https://xnat.readthedocs.io/en/latest/static/tutorial.html [en ligne], tutoriel sur l'api xnat, consulté le 20/04/2019

https://wiki.xnat.org/docs16/3-administrator-documentation/customizing-xnat/defining-project-metadata-modification [en ligne], xnat api, consulté le 13/05/2019

https://wiki.xnat.org/docs16/3-administrator-documentation/customizing-xnat/defining-project-metadata-modification [en ligne], tutoriel d'installation de la suite elk, consulté le 20/04/2019

[ERROR][logstash.pipeline] A plugin had an unrecoverable error. Will restart this plugin · Issue #6279 · elastic/logstash · GitHub (forum logstash, consulté le 22/04/2019)

https://www.microsoft.com/en-us/download/details.aspx?id=55539 [en ligne], lien de téléchargement du connecteur MySQL, consulté le 28/04/2019

https://jdbc.postgresql.org/download.html [en ligne], téléchargement du connecteur PostgreSQL, consulté le 28/04/2019

https://stackoverflow.com/questions/17426521/list-all-indexes-on-elasticsearch-server [en ligne], API REST, consulté le 02/05/2019)

https://www.elastic.co/fr/blog/logstash-jdbc-input-plugin [en ligne], blog ELK consulté 05/05/2019

https://www.elastic.co/guide/en/elasticsearch/reference/6.1/_list_all_indices.html [en ligne], commande REST, consulté le 07/05/2019)

https://github.com/Cyb3rWard0g/HELK/issues/70 (la sécurité x-pack, consulté le 20/05/2019)

https://www.elastic.co/fr/blog/logstash-jdbc-input-plugin (consulté le 22/05/2019)

https://discuss.elastic.co/t/xml-et-logstash-index-non-rempli/159023 (consulté 25/05/2019)

https://www.elastic.co/guide/en/kibana/current/connect-to-elasticsearch.html (consulté le 27/04/2019)

10 Annexes

ANNEXE I : L'INSTALLATION DETAILLEE DE LA SUITE ELK

ANNEXE II : CONFIGURATION DE LA SUITE ELK

ANNEXE III : INSTALLATION DU PACK DE SECURITE X-PACK

ANNEXE IV : SCRIPT DE GENERATION DU GUID

ANNEXE I : Installation détaillée de la suite ELK

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~pré-requis~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
le paquet apt-transport-https
le paquet curl
machine virtuelle java version 8
elasticseach version 5
kibana version 5
logstash version 6
connecteur jdbc mysql
connecteur jdbc postgresql
licence xpack

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~Etapes de l'installation~~~~~~~~~~~~~~~~~~~~~~~~~~

1. installation apt-transport-https

*****il permet l'utlisation des lignes deb .. dans /etc/apt/sources.list pour que les gestionnaires de paquets qui utilisent le bibliothèque libapt-pkg puissent accéder aux méta-données et paquets par https***

#apt-get install apt-transport-https

2. installation de curl

c'est une interface en ligne de commande, destinée à récupérer le contenu d'une ressource accessible par un réseau

#apt-get install curl

3. Machine virtuelle java8

la version utilisée dépend de la version de elasticsearch et logstash

#apt-get install openjdk-8-jdk

4. installation elasticsearch et kibana

###ajout du dépot
sudo wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
echo "deb https://artifacts.elastic.co/packages/5.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-5.x.list

###charger le dépot et install

apt update
apt install elasticsearch kibana

5. Installation de logstash version 6

##ajout du dépot

sudo wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
echo "deb https://artifacts.elastic.co/packages/6.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-6.x.list

##charger le dépot et install

apt update
apt install logstash

ANNEXES II : Configuration de la suite ELK

1) configuration elasticsearch

##Editer le fichier /etc/elasticsearch/elasticsearch.yml

1. decommenter la ligne network.host et remplacer localhost par l'IP de la machine
2. decommenter la ligne http.port

##Editer le fichier /etc/elasticsearch/jvm.options

**ce fichier indique la quantité de mémoire que va utilisée la machine virtuelle java

1. decommenter les lignes Xms1g et Xmx1g
2. remplacer 1g par 2g sur les deux lignes (dépend des ressources de la machine)

###demarrer elasticsearch

systemctl start elasticsearch

**demarrage automatique

systemctl enable alsticsearch

##verifier le serveur

curl -XGET ip_serveur:9200

**sur un navigateur ip_serveur:9200

2) configuration de kibana

##Editer le fichier /etc/kibana/kibana.yml

1. decommenter la ligne serveur.host et remplacer localhost par l'ip du serveur
2. decommenter la ligne serveur.port
3. decommenter la ligne elasticsearch.url et mettre "http://ip-serveur:9200"

##demarrer kibana

systemctl start kibana

**demarrage automatique

systemctl enable kibana

##verifier kibana

se connecter sur http://ip_serveur:5601

3) configuration de logstash

##Editer le fichier /etc/logstash/jvm.options

**ce fichier indique la quantité de mémoire que va utilisée la machine virtuelle java

1. decommenter les lignes Xms1g et Xmx1g
2. remplacer 1g par 2g sur les deux lignes (depend des ressources de la machine)

##installation du plugin jdbc

ce plugin va permettre d'établir la connexion aux bases de données

1. se mettre sur le repertoire /usr/share/logstash/bin/

cd /usr/share/logstash/bin/

2. install plugin

./logstash-plugin install logstash-input-jdbc

####Téléchargement des connecteurs

#creer le repertoire d"accueil du connecteur

mkdir -p /usr/share/logstash/vendor/jdbc-mysql pour mysql

mkdir -p /usr/share/logstash/vendor/jdbc-postgres pour postgresql

#telecharger connecteurs

1. telecharger le connecteur mysql sur et le placer dans le repertoire pour mysql

https://docs.microsoft.com/en-us/sql/connect/jdbc/download-microsoft-jdbc-driver-for-sql-server

2. telecharger le connecteur postgresql et le placer dans le repertoire pour postgres

http://pape-mademba.dieye.etu.perso.luminy.univ-amu.fr/pub/stage/

#changement de droits

1. changer le proprietaire du repertoire /usr/share/logstash/vendor/jdbc-mysql

chown logstash:logstash /usr/share/logstash/vendor/jdbc-mysql -Rvf

2.changer le propriétaire du repertoire /usr/share/logstash/vendor/jdbc-postgres

chown logstash:logstash /usr/share/logstash/vendor/jdbc-mysql -Rvf

###configuration des pipelines

#Editer le fichier pipelines.yml

1. attribuer un pipeline à un fichier de configuration ou plusieurs

2.exemple : cette ligne attribue le pipeline d'id pip au fichier de configuration test.conf

```
- pipeline.id: pip
  path.config: "/etc/logstash/conf.d/test.conf"
```

**attribuer un pipeline pour chaque fichier de configuration, ça evite d'ecraser les données

###veuillez trouver des fichiers de configuration à adapter à votre configuration permettant d'indexer une base MySQL (conf.d) et une base postgreSQL (conf1.d) dans elasticsearch sur ce lien:

http://pape-mademba.dieye.etu.perso.luminy.univ-amu.fr/pub/stage/fichier_config_logstash.zip

ANNEXE III : Installation et configuration de x-pack

ce plugin permet d'activer l'authentification sur elk

1) x-pack sur elasticsearch

1. se mettre sur le repertoire /usr/share/elasticsearch/bin

cd /usr/share/elasticsearch/bin

2. installer x-pack

./elasticsearch-plugin install x-pack

user par défaut : elastic
mot de passe par défaut : changeme

3. changer le mot de passe par defaut

curl -XPUT -u elastic:changeme
'ip_serveur:9200/_xpack/security/user/elastic/_password?pretty' -H 'Content-Type:
application/json' -d'
{
  "password": "Nouveau_mot_de_passe"
}
'

**pour se connecter avec l'authentification
curl -XGET -u elastic:NewElasticPassword ip_serveur:9200

**NB: pensez à mettre le mot de passe de elasticsearch dans le fichier de configuration logstash bloc output

2) x-pack sur kibana

1. se mettre sur le repertoire /usr/share/kibana/bin

cd /usr/share/kibana/bin

2. installer x-pack

./kibana-plugin install x-pack

user par défaut : kibana
mot de passe par défaut : changeme

3. changer le mot de passe par defaut

```
curl -XPUT -u kibana:changeme
'ip_serveur:9200/_xpack/security/user/kibana/_password?pretty' -H 'Content-Type:
application/json' -d'
{
  "password": "Nouveau_mot_de_passe"
}
```

ANNEXE IV : Script de génération du GUID

```
#!/bin/bash
#DIEYE PAPE MADEMBA
#script remplis les infos perso dans une base de donnée info_sujet



#récupéraation du contenu des champs du formu
nom=`echo "$QUERY_STRING" | sed -n 's/^.*nomcc=\([^&]*\).*$/\1/p' | sed "s/%20/ /g"`
nom1=`echo "$QUERY_STRING" | sed -n 's/^.*nomc=\([^&]*\).*$/\1/p' | sed "s/%20/ /g"`
prenom=`echo "$QUERY_STRING" | sed -n 's/^.*prenom=\([^&]*\).*$/\1/p' | sed "s/%20/ /g"`
prenom1=`echo "$QUERY_STRING" | sed -n 's/^.*prenom1=\([^&]*\).*$/\1/p' | sed "s/%20/ /g"`
age=`echo "$QUERY_STRING" | sed -n 's/^.*age=\([^&]*\).*$/\1/p' | sed "s/%20/ /g"`
age1=`echo "$QUERY_STRING" | sed -n 's/^.*age1=\([^&]*\).*$/\1/p' | sed "s/%20/ /g"`
sexe=`echo "$QUERY_STRING" | sed -n 's/^.*sexe=\([^&]*\).*$/\1/p' | sed "s/%20/ /g"`
email=`echo "$QUERY_STRING" | sed -n 's/^.*email=\([^&]*\).*$/\1/p' | sed "s/%20/ /g"`

echo "Content-type: text/html"
echo ""
echo "<html >
<head><title>sauvegarde infos personnelles du sujet</title>

</head>"
echo "<body>"

#stocke info dans fichier
echo "$nom $prenom $age $sexe" > /usr/lib/cgi-bin/fichier_priv_.txt

#hachage du fichier pour generer guid
guid=`sha256sum /usr/lib/cgi-bin/fichier_priv_.txt | cut -d ' ' -f1`
echo >  /usr/lib/cgi-bin/fichier_priv_.txt
#controle si les infos existent dans la base
lesguid=(`psql -d info_sujet -U postgres -c "SELECT guid FROM sujet"| cut -d '(' -f1 `)

echo $bool
#boucle controlant si le sujet existe
     for((i=3;i<=${#lesguid[@]}-1;i++)) do

#s'il existe je mets bool à vrai
          if [ ${lesguid[$i]} = $guid ]; then
```

```
           bool="vrai"

          else bool="faux"

          fi

#echo $guid
      done

   if [ $bool == 'faux' ]; then
#ecris sur la base de données
psql -d info_sujet -U postgres -c "INSERT INTO sujet(nom, prenom, age, sexe, email, guid)  VALUES
('$nom', '$prenom', $age, '$sexe', '$email', '$guid')" >/dev/null
#controle si l'ecriture sur la base a réussi
     if [ $?==0 ];then
       echo "informations remplies avec succés!        "
       else "Une erreur s'est produit veuillez réessayer"
      fi
#vider le fichier utilisé

    else echo "Le sujet exitse deja"
    fi


echo " <a href="http://10.164.7.69">Revenir au formulaire</a>"

echo "</body>"
echo "</html> "
```

NB : vous trouverez tous les documents avec des explications claires du projet sur mon site internet :

http://pape-mademba.dieye.etu.perso.luminy.univ-amu.fr/pub/stage/