Note intermédiaire rapport de stage

Mathieu Thomassin

2024-05-28

L'Insee a renforcé récemment ses capacités opérationnelles en terme de cybersécurité via notamment la création de son SOC (Security Operation Center) en septembre 2023. Devant la quantité de données et leur diversité et face aux évolutions des techniques et tactiques des attaquants, les méthodes de détection de cyberattaques peuvent avoir des limites. L'application d'algorithmes de Machine Learning peut aider les analystes SOC à repérer des attaques. Se déroulant au sein de l'équipe SOC construite depuis peu, le stage va permettre d'appliquer des techniques de Machine Learning et de deep learning sur des jeux de données réelles, afin de participer à la détection d'incidents de sécurité.

Table of contents

1 Introduction

- **Présentation du stagiaire et du maître de stage** :
  - stagiaire: Mathieu THOMASSIN
  - maître de stage: Michael ORSUCCI
  - Titre du stage: Machine Learning appliqué à la cybersécurité

2 Description de la mission

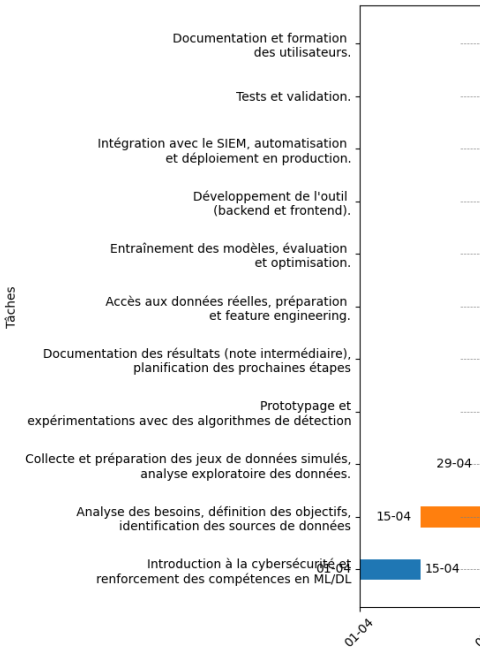
2.1 Environnement de travail:

- **Présentation de l'organisation** : L'Insee a renforcé récemment ses capacités opérationnelles en terme de cybersécurité via notamment la création de son SOC (Security Operation Center) en septembre 2023. Cette équipe est répartie entre les sites de la DR de Nantes et la DR de Metz et a pour objectif de renforcer la sécurité du SI.

- **Contexte général du stage** : Devant la quantité de données et leur diversité et face aux évolutions des techniques et tactiques des attaquants, les méthodes de détection de cyberattaques peuvent avoir des limites. L'application d'algorithmes de Machine Learning peut aider les analystes SOC à repérer des attaques. Le stage va permettre d'appliquer des techniques de Machine Learning et de deep learning sur des jeux de données réelles, afin de participer à la détection d'incidents de sécurité.
- **Objectifs du Stage** :
  - Objectifs principaux: l s'agit d'appliquer des techniques de détection de requêtes malveillantes arrivant dans le SI de l'Insee. Les requêtes arrivant au sein du SI peuvent être centralisées par un SIEM (Security information and event management).
  - Résultats attendus:
    * Offrir un service s'ajoutant au SIEM permettant d'identifier des requêtes malveillantes.
    * Pouvoir comparer différents *meilleurs* modèles (au sens d'une recherche dans les hyper paramètres) entre eux
    * Favoriser les bonnes pratiques du MLOps: reproductibilité, contrôle de version, automatisation, surveillance, collaboration
    * Étendre la méthodologie à des jeux de données publics différents
    * Explorer les algorithmes de détection d'anomalie
- **Enjeux** :
  - Importance de la mission pour l'entreprise: à titre d'exemple, l'Insee détient l'application Elire. Une attaque réussie sur cette application perturberait le bon déroulement de la vie démocratique.
  - Impact potentiel des résultats: La détection de requêtes malveillantes pourrait permettre de déjouer une attaque en cours

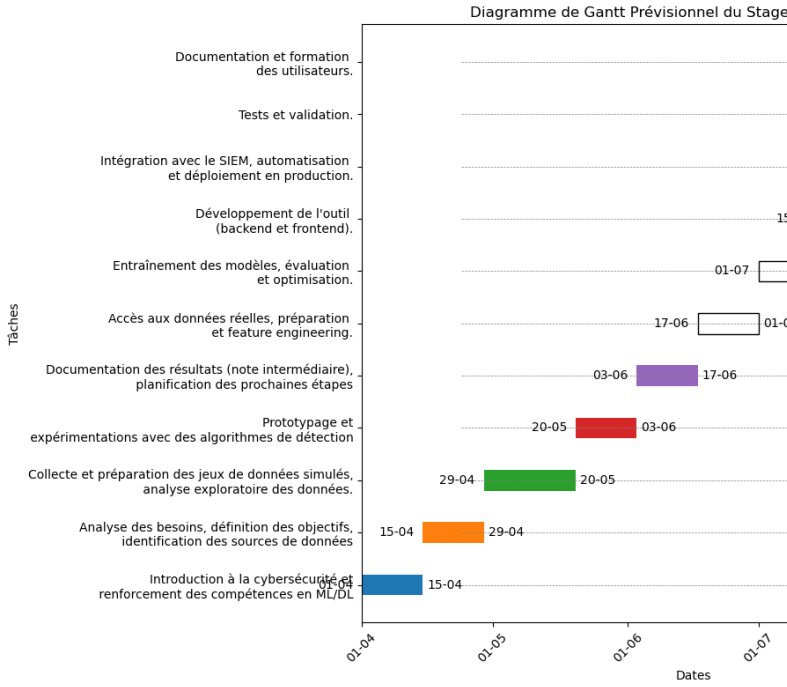
2.1.0.1 3. Timing Prévisionnel

- **Calendrier des Étapes** :



  - Détail du planning prévisionnel des différentes phases du projet
  - Échéances importantes et jalons:
    * note intermédiaire (1er juin)
    * être prêt pour tester pendant les élections européennes (8 et 9 juin)
    * surveillance pendant les jeux olympiques (26 juillet au 11 août)
    * rapport final (fin août)

- **Avancement Actuel** :



- Comparaison avec le calendrier initial:
- Retards éventuels et raisons:
  - Les premiers tests et expérimentations ont eu lieu, cependant, j'ai préféré commencer à comprendre plus en profondeur les design patterns de deep learning ainsi que le fonctionnement de MLFlow avant de pousser davantage la comparaison des modèles et des datasets. MLFLow va permettre une bonne méthode de comparaison et de sélection des meilleurs modèles, dans la perspective d'une intégration avec une mise en production.

  Je n'utilise donc pour le moment qu'un seul dataset constitué de bonnes et mauvaises requêtes labellisées, un seul modèle (sur plusieurs déjà entraînés).

  Je n'ai pas encore terminé l'intégration sur MLFlow.

2.1.0.2 4. Méthodologies Utilisées et Prévisionnelles

- **Collecte de Données** :
  - Sources des données:
    - dataset public pour l'entraînement des modèles:
      - http://www.secrepo.com/#
      - https://www.netresec.com/index.ashx?page=PcapFiles
      - https://www.stratosphereips.org/datasets-overview
      - description des datasets
      - datasets-for-network-security-e25238704c7f

            - dépôt github contenant des liens vers plusieurs datasets
            - Good et Bad queries dataset
        * données issues du SIEM, soit le POC en cours (fin mai) soit potentiellement le SIEM qui sera mis en place
    - Méthodes de collecte:
        * téléchargement à partir des liens
        * téléchargement au format Json à partir d'une recherche dans Splunk. Il serait souhaitable de réussir à utiliser l'API du SIEM afin d'automatiser cette tâche et de reproduire les résultats.

- **Préparation et Nettoyage des Données** :
    - Techniques de nettoyage: **Lecture des fichiers de requêtes**, étiquetage en fonction de leur nature (bonne ou mauvaise) et concaténation dans un DataFrame.

      Étiquetage des données : - Les données sont lues à partir de fichiers texte et chaque ligne est étiquetée comme étant soit une mauvaise requête (1) soit une bonne requête (0)

      Séparation des ensembles d'entraînement et de test :
        * Les données étiquetées sont divisées en deux ensembles : un ensemble d'entraînement (80%) et un ensemble de test (20%).

      Tokenisation personnalisée :
        * Une fonction de tokenisation personnalisée utilise des expressions régulières pour extraire des tokens pertinents des textes, y compris les mots, les chiffres, les URL et certains caractères spéciaux.

      Vectorisation TF-IDF :
        * Les données textuelles sont transformées en matrices de caractéristiques en utilisant la vectorisation TF-IDF, ce qui permet de préparer les données pour un modèle de machine learning.
    - Outils utilisés:
        * **Pandas** : Manipulation de données.
        * **Scikit-learn (sklearn)** :
            - `train_test_split` : Séparation des données en ensembles d'entraînement et de test.
            - `TfidfVectorizer` : Vectorisation des données textuelles en utilisant TF-IDF.
        * **re** : Utilisation d'expressions régulières pour la tokenisation personnalisée.

- **Analyse et Modélisation** :
    - Méthodes d'analyse (exploratoire, descriptive): **Extraction des caractéristiques** :

        * Une fonction `extract_features` est définie pour extraire les parties importantes des requêtes, en particulier les noms de scripts ou de fichiers ciblés.
        * Les données sont nettoyées pour éliminer les caractères non désirés et les retours à la ligne, puis les scripts/fichiers sont extraits et ajoutés dans une nouvelle colonne `Script` du DataFrame. **Statistiques descriptives** :
        * Les statistiques descriptives sur les scripts/fichiers ciblés sont générées en utilisant `value_counts()` pour obtenir la fréquence des différents scripts/fichiers ciblés.
        * Les 20 scripts/fichiers les plus fréquents sont affichés.
    - Modèles de machine learning ou autres techniques analytiques envisagées

- **Évaluation des Modèles** :
    - Métriques de performance prévues
    - Stratégies de validation

2.1.0.3 5. État des Travaux

- **Travaux Réalisés Jusqu'à Présent** :
    - Progrès réalisés
    - Résultats obtenus jusqu'ici
- **Difficultés Rencontrées** :
    - Problèmes techniques ou méthodologiques
    - Solutions envisagées ou mises en place
- **Travaux Restants** :
    - Étapes à venir
    - Objectifs pour les prochaines semaines

2.1.0.4 6. Bibliographie

Le SOC étant une nouvelle unité, il n'y a pas eu de travaux sur le sujet en amont. J'ai axé ma recherche bibliographique autour des techniques de machine learning pour la cybersécurité, et autour des techniques de Deep Learning.

- **Références Utilisées** :
    - Articles scientifiques:
        * Isolation Forest, Liu, Ting, Zhou
        * R. Fontugne, P. Borgnat, P. Abry, K. Fukuda. "MAWILab: Combining diverse anomaly detectors for automated anomaly labeling and performance benchmarking". ACM CoNEXT 2010. Philadelphia, PA. December 2010.

  - Livres et manuels:
    * An introduction to statistical learning with applications in Python
    * Deep Learning Patterns and Practices, Andrew Ferlitsch
    * Deep Learning from Scratch - Building with Python from first principles, Seth Weidman
    * Deep Learning_ A Visual Approach – Andrew Glassner – Illustrated, 2021
    * Machine learning with Python cookbook, Kyle Gallatin & Chris Albon
    * Deep Learning, Ian Goodfellow
    * Machine Learning for cybersecurity, Emmanuel Tsukerman, (Chapter 6 Automatic Intrusion Detection)
    * Machine Learning for computer and Cyber Security, Principles, Algorithms, and Practices, Brij B. Gupta
  - Documentation technique

- **Ressources Additionnelles** :
  - Sites web
    * Scikit-Learn
  - Outils et bibliothèques spécifiques
    * Splunk

- Articles:

2.1.0.5 7. Conclusion

- **Synthèse de l'Avancement** :
  - Résumé des principales réalisations
  - Confirmation du bon déroulement du stage

- **Perspectives** :
  - Étapes finales du stage
  - Objectifs pour la phase suivante

2.1.1 Format et Soumission

- **Format du Document** :
  - Nommer le fichier suivant le format : « NOM_Prénom_mi_parcours »

- **Modes de Soumission** :
  - Soumission sur Moodle
  - Envoi par mail à l'enseignant référent

En suivant ce plan, vous devriez être en mesure de rédiger une note d'étape complète et structurée, respectant les exigences de votre programme de stage. Assurez-vous de respecter la longueur recommandée (2 à 3 pages) en étant concis et précis dans vos descriptions.

2.1.2 2. État de l'Art

- **Revue de Littérature** : Synthèse des travaux précédents, articles scientifiques, et technologies utilisées en lien avec la problématique du stage.

Le SOC étant une nouvelle unité, il n'y a pas eu de travaux sur le sujet en amont. J'ai axé ma recherche bibliographique autour des techniques de machine learning pour la cybersécurité, et autour des techniques de Deep Learning.

- Livres:
  - An introduction to statistical learning with applications in Python
  - Deep Learning Patterns and Practices, Andrew Ferlitsch
  - Deep Learning from Scratch - Building with Python from first principles, Seth Weidman
  - Deep Learning_ A Visual Approach – Andrew Glassner – Illustrated, 2021
  - Machine learning with Python cookbook, Kyle Gallatin & Chris Albon
  - Deep Learning, Ian Goodfellow
  - Machine Learning for cybersecurity, Emmanuel Tsukerman, (Chapter 6 Automatic Intrusion Detection)
  - Machine Learning for computer and Cyber Security, Principles, Algorithms, and Practices, Brij B. Gupta
- Articles:
  - Isolation Forest, Liu, Ting, Zhou
  - R. Fontugne, P. Borgnat, P. Abry, K. Fukuda. "MAWILab: Combining diverse anomaly detectors for automated anomaly labeling and performance benchmarking". ACM CoNEXT 2010. Philadelphia, PA. December 2010.
  - **Technologies et Outils** : Présentation des technologies, outils, et bibliothèques couramment utilisés en data science et spécifiques au projet.

Pour ce projet, j'utilise Python, et des notebook Jupyter, au sein d'environnement hébergés sur la plateforme Onyxia (qui met à dispositio un GPU) ou son équivalent en interne à l'Insee (qui n'en dispose pas). J'utilise les packages classiques de machine learning dont scikit-learn, et les outils associés: TfidfVectorizer, tensorflow, LogisticRegressionLSTM, XGBoost CNN, KFold GridSearchCV. J'ai également commencé à utiliser l'outil MLFlow, qui est une plateforme permettant de gére le cycle de vie d'un projet de machine learning de bout en bout, notamment les étapes d'expérimentation et d'amélioration continue.

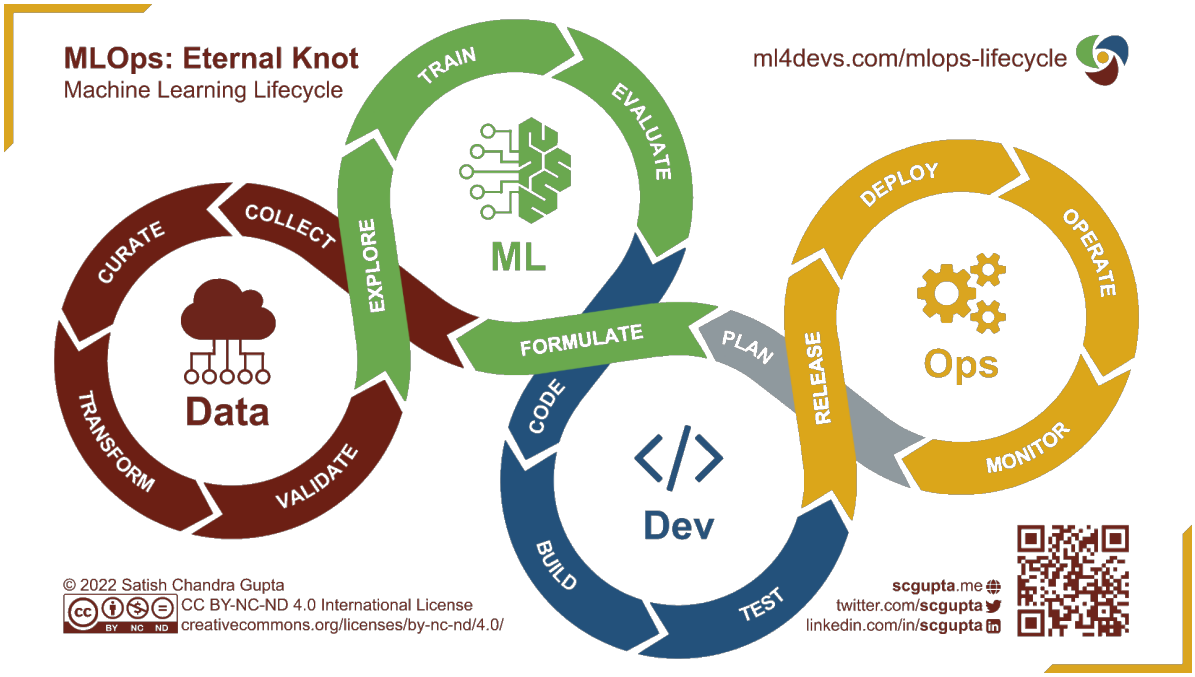


Figure 1: L'approche MLOps

2.1.3 3. Méthodologie

- **Collecte de Données** : Sources des données, méthodes de collecte, et description des jeux de données utilisés.

Les données se classent en deux catégories: les données sur lesquelles les modèles sont entraînés, et les données issues de Splunk. Pour le moment, j'ai testé différents algorithmes sur un unique jeu de données que je nomme ["Good-Bad queries"]https://github.com/faizann24/Fwaf-Machine-Learning-driven-Web-Application-Firewall.git

- **Préparation des Données** : Techniques de nettoyage, transformation, et enrichissement des données.
- **Exploration des Données** : Analyse exploratoire, visualisations, et insights préliminaires obtenus des données.
- **Modélisation** : Description des modèles de machine learning ou autres techniques analytiques utilisés. Justification des choix de modèles.
- **Évaluation des Modèles** : Métriques et méthodes utilisées pour évaluer la performance des modèles.

2.1.4 4. Développement et Implémentation

- **Architecture du Système** : Schéma et description de l'architecture du système ou de la solution développée.
- **Pipeline de Données** : Description du pipeline de traitement des données, de l'ingestion à l'analyse.
- **Déploiement** : Processus de déploiement des modèles ou solutions, environnements utilisés (local, cloud, etc.), et outils de déploiement.

2.1.5 5. Résultats et Discussions

- **Résultats Obtenus** : Présentation des résultats des analyses et des performances des modèles.
- **Interprétation des Résultats** : Analyse et interprétation des résultats. Comparaison avec les attentes initiales et les benchmarks.
- **Limites et Contraintes** : Discussion sur les limites rencontrées, les défis techniques, et les contraintes liées aux données ou aux outils.

2.1.6 6. Conclusion et Perspectives

- **Synthèse du Travail** : Résumé des principales réalisations et des contributions du stage.
- **Perspectives d'Avenir** : Suggestions pour des travaux futurs, améliorations possibles, et nouvelles pistes de recherche ou de développement.

2.1.7 7. Références

- **Bibliographie** : Liste des ouvrages, articles, et ressources consultés pour la réalisation du projet.
- **Outils et Logiciels** : Références des outils et logiciels utilisés.

2.1.8 8. Annexes

- **Code Source** : Extraits de code pertinents ou lien vers un dépôt de code.
- **Documents Supplémentaires** : Diagrammes, schémas, documents techniques, etc.

Ce plan vous fournira une structure claire et organisée pour votre note intermédiaire, couvrant tous les aspects essentiels de votre stage en data science. Chaque section peut être adaptée en fonction des spécificités de votre projet et des exigences de votre programme de master.