LOG6302A — Analyse d'applications et Cyber-sécurité

Laboratoire #6

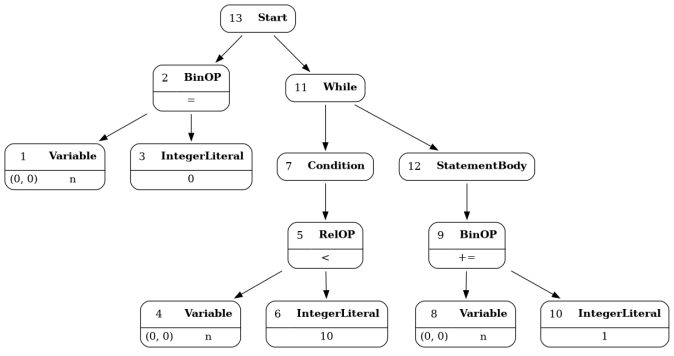
Donné le : Apr 04 2024, 09 :30 AM
Échéance : Apr 18 2024, 09 :30 AM

- Il s'agit d'un travail en équipe de deux.
- Chaque groupe doit rendre sur Moodle une archive contenant leur code et un rapport (PDF) avant la date limite.
- Le rapport doit rendre compte de ce que vous avez fait et les problèmes rencontrés. Vous pouvez discuter de tout autre élément que vous jugeriez pertinent.
- Chaque jour de retard entraîne une pénalité de 50%.
- Si vous avez des questions, vous pouvez demander des clarifications sur Discord (#lab-question)

Détection de clone et similarité

Vecteur de métriques

Pour résoudre les problèmes posés dans ce laboratoire, nous allons extraire des métriques des AST. Pour ce faire, vous trouverez dans la classe python AST une méthode "*vectorize*". Cette dernière retourne un vecteur qui correspond aux occurrences de chaque type de nœuds : un exemple concret de la transformation appliquée est disponible ci-dessous.



(a) AST

Prenons le dictionnaire suivant : $\{Start, BinOP, Variable, IntegerLiteral, While, Condition, RelOP, StatementBody, String\}$

Le résultat de la transformation serait alors le vecteur : $[1, 2, 3, 3, 1, 1, 1, 1, 0]$

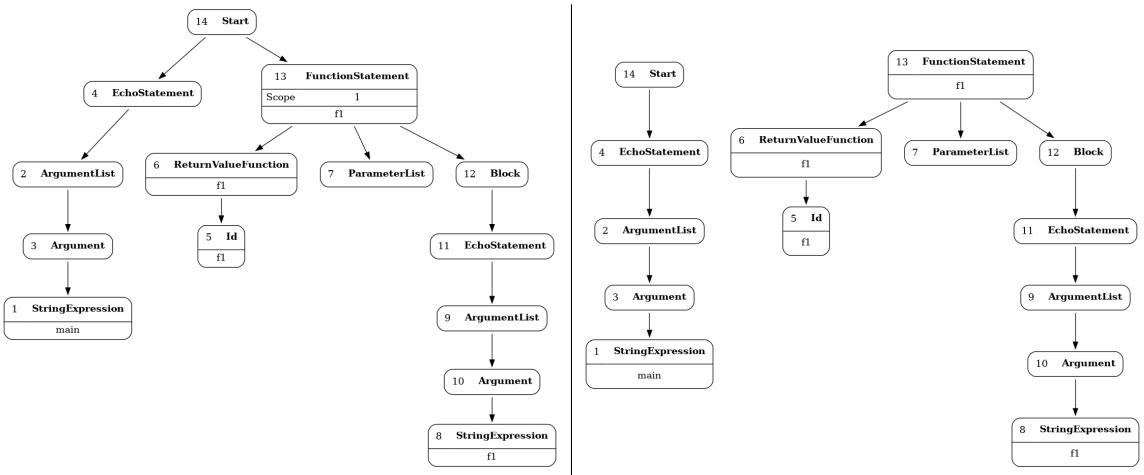
(b) Vecteur

```python
from code_analysis import ASTReader
reader = ASTReader()
ast    = reader.read_ast("file.ast.json")
vector = ast.vectorize()
# Will return a numpy vector of 126 values
```

(c) Python code

Fragmentation

Le vecteur de métriques extrait précédemment s'applique à l'AST complet, donc à l'ensemble du fichier. Pour extraire les métriques à un niveau plus fin, il va falloir fragmenter l'AST. Par exemple, si on souhaite extraire les métriques de chaque fonction pour faire des calculs de similarité dessus, il va falloir fragmenter les AST au niveau des fonctions (nœuds *MethodStatement et *FunctionStatement).



(a) AST (b) Fragmented AST

Dataset

Nous allons travailler sur un dataset contenant du code potentiellement malveillant, composé de "phishing kits" :
Les "phishing kits" sont des archives prêtes à être déployés dont l'utilisation ne nécessite qu'un minimum d'efforts. Leurs développeurs les vendent et fournissent des instructions pour les "attaquants" / "phishers". Ces kits sont conçus pour générer des copies de sites web représentant des marques célèbres ayant un large public, et vont ensuite exfiltrer les information des victimes (passwords, credit card, mails, etc).
Bien que le code sources des kits que nous allons analyser provient d'un dataset public accessible sur GitHub (PhishingKitTracker), nous allons travailler uniquement à partir des AST dans ce TP.

Attention Toutes informations trouvées (mail, URL, etc) ne doivent surtout pas être utilisées ou enregistrées.

1 Fichiers

1.1 Clones "paramétriques"

Trouvez dans le dataset le plus gros groupe de fichiers ayant des vecteurs identiques, en limitant la recherche aux AST de **plus de 100 nœuds**.

— Que remarquez-vous concernant ces fichiers ?
— Refaite l'opération en ignorant les vecteurs dupliqués au sein d'un même kit.

1.2 Fichier similaires

Trouvez dans le dataset le plus gros groupe de fichiers ayant des vecteurs similaires, en limitant la recherche aux AST de **plus de 100 nœuds**.
Ici, pour être considéré similaire, la distance de Manhattan entre les deux vecteurs ne doit pas dépasser 30% de la taille du vecteur.

```
# Voici un code d'exemple avec numpy
if numpy.abs(vector_j-vector_i).sum() <= 0.3 * numpy.sum(vector_i):
    # similaire
```

2 Fragments

2.1 Clones "paramétriques"

Trouvez dans le dataset le fragment le plus dupliqué, en limitant la recherche aux fragments de **plus de 10 nœuds**, et en ignorant les fragments dupliqués au sein d'un même kit.

— Quelle est le nom(s) de la fonction correspondant à ce fragment ?

2.2 Fragments similaires

Trouvez dans le dataset le plus gros groupe de fragment ayant des vecteurs similaires, en limitant la recherche aux AST de **plus de 10 nœuds**, et en ignorant les fragments dupliqués au sein d'un même kit.
Ici, pour être considéré similaire, la distance de Manhattan entre les deux vecteurs ne doit pas dépasser 10% de la taille du vecteur.

— Quelle est le nom(s) de la fonction correspondant à ces fragments ?

3 Kits "paramétriques"

Calculer pour chaque kit un vecteur le représentant, en sommant (par colonne) tout les fragments qui composent ce kit.
Chercher ensuite le plus gros groupe de vecteurs identiques.

— Quelles sont ces kits ?

— Il y a-t-il des différences entre ces kits au niveaux des types des nœuds de l'AST (type, image) ?

4 Rapport

Détailler dans le rapport les résultats de votre analyse.