

PROJET RESEAU

Analyseur Trame

LU3IN033

Serraf Dan
Mohellebi Melissa
2020-2021

Sommaire

1 Quelles sont les installations requises et comment lancer notre analyseur ?

2 Architecture du projet

3 Descriptions de nos choix

A. Choix de la technologie

B. Méthode de développement

C. Contributions personnelles

4 Démonstrations



QUELLES SONT LES INSTALLATIONS REQUISES ET COMMENT LANCER NOTRE ANALYSEUR ?

L'exécution de notre analyseur nécessite d'avoir python3.

Pour exécuter le projet :

1. Ouvrir un terminal
2. Ce placé dans le sous répertoire Analyse
3. Exécuté dans le terminale la commande suivante

```
python3 main.py chemin_fichier_analyser chemin_fichier_stocker
```

chemin_fichier_analyser : le chemin du fichier que vous souhaitez analyser

chemin_fichier_stocker : le chemin du fichier ou vous souhaitez stocker le résultat

Option : -d -détaille (a la fin de la commande)

ARCHITECTURE DU PROJET

Le projet est composée de plusieurs sous répertoires:

1 Analyseur

Le dossier analyseur contient les fichiers python pour l'exécution de l'analyseur.

2 Descriptions

Le dossier est composer d'un fichier howto.txt qui explique comment installer et lancer notre programme.

Ainsi qu'un fichier readme.txt qui décrit la structure de notre code.

3 Tests

Le dossier tests contient des trames que l'on a utiliser pour tester notre analyseur.

4 Résultat

Le dossier résultat contient les résultats des différents tests suite a l'analyse de notre analyseur.

ANALYSEUR

Main.py : Fichier principal qui permet d'exécuter le programme en fonction des paramètres saisies par l'utilisateur

ValideFichier : Fichier qui permet de nettoyer et de valider le fichier texte qui contient les trames

Reseau.py : Objet qui regroupe différentes variables utiles pour l'analyse de la trame

Ip.py : Classe qui contient l'objet Ip et les différentes fonctions pour extraire les champs du protocole Ip

Tcp.py : Classe qui contient l'objet Tcp et les différentes fonctions pour extraire les champs du protocole Tcp

CHOIX DE LA TECHNOLOGIE

Python

- Langage que l'ont maitrise le mieux
- Faciliter de développement
- Gestion des erreurs
- Traitement des fichiers simple
- Conversions simple en différents type (hexadécimal , char , int)
- Multiple bibliothèques pour différents emplois (Gestion serveur, graphique ...)

MÉTHODE DE DÉVELOPPEMENT

- Miser sur la faciliter de développement, tous développeurs peut faire évoluer ce projet en suivant le protocole de développement .

Pour ajouter un protocole il faut :

- Créer une classe NomProtocole
- L'objet possédera autant d'attribut que le protocole possèdera de champs.
- Chaque champs sera renseigner à l'aide d'une fonction calculer_champs
- Diviser pour mieux régner
- Convention de nommage pout un meilleur développement à travers l'équipe du projet. (Underscore case)

CONTRIBUTIONS PERSONNELLES

Serraf Dan

- Mise en place de la structure générale
- Nettoyage et validation du fichier
- Formatage en mode Wireshark
- Gestion des paramètres d'entrée saisie par l'utilisateur
- Gestion des erreurs

Mohellebi Melissa

- Application du protocole de développement
- Création des différents fichiers de tests
- Formatage en mode description
- Suggestions d'amélioration

CONTRIBUTIONS PERSONNELLES

Serraf Dan

- Mise en place de la structure générale
- Nettoyage et validation du fichier
- Formatage en mode Wireshark
- Gestion des paramètres d'entrée saisie par l'utilisateur
- Gestion des erreurs

Mohellebi Melissa

- Application du protocole de développement
- Création des différents fichiers de tests
- Formatage en mode description
- Suggestions d'amélioration
- Structure générale du projet

DEMONSTRATION

- Test 1 : Test frame complète (Ethernet, Ip, Tcp, http)
 - Test 2 : Trame trop petite(1)
 - Test 3 : Position offset courant invalide (3)
 - Test 4 : Commentaire entre ligne et fin ligne (6)
 - Test 5 : Trame sur une ligne (7)
 - Test 6 : Plusieurs frames mode wireshark (12)
 - Test 7 : Plusieurs frames mode description (11)
 - Test 8 : Analyse en direct avec wireshark
- 