

BTS Services Informatiques aux Organisations	
E4- Conception et maintenance de solutions informatiques (Coefficient 4)	
DESCRIPTION D'UNE SITUATION PROFESSIONNELLE N°2	
<p>Nom et prénom: David AMARO MONTEIRO Parcours : SISR Situation: Stage 2eme Année BTS SIO SISR</p>	
<p>Contexte de la situation professionnelle: Cette situation professionnelle a été réalisé dans le cadre d'un projet personnalisé encadré. Ce projet à pour but de créer un outil d'automatisation sur Linux pour permettre d'analyser un réseau local ou choisi par l'utilisateur...</p> <p>Objectif: Création d'un script BASH sous Linux pour analyser un réseau</p>	
<p>Intitulé de la situation professionnelle: Script BASH sous Linux Période de réalisation: 8/01/2023 -3/02/2023 Lieu: SC Micro, Landerneau 29800 Modalité: Seul</p>	
<p>Compétences validées : * Gérer le patrimoine informatique * Répondre aux incidents et aux demandes d'assistance et d'évolution * Mettre à disposition des utilisateurs un service informatique * Organiser son développement professionnelle</p>	
<p>Ressources fournies:</p> <ul style="list-style-type: none"> - OS Linux Ubuntu 20.04 - Schéma réseau - Exemple des résultats du script <p>Résultats attendus:</p> <ul style="list-style-type: none"> - Création du Script nmap - Script fonctionnel 	
<p>Productions associées:</p> <ul style="list-style-type: none"> - Internet - Captures d'écran 	
<p>Modalités d'accès aux productions :</p> <p>https://drive.google.com/drive/folders/1ptBswWChqkc4n-fGPw8USh5Ir6wlZKCN?usp=sharing</p>	

Descriptif de la réalisation professionnelle.

Introduction

Le but de cette mission est de concevoir un outil d'automatisation sous Linux (Ubuntu). Cet outil sera développé en script Linux (BASH), elle va contenir différentes instructions à exécuter dans la console, les unes à la suite des autres. Elle servira à analyser un réseau local ou choisi par l'utilisateur (ex : réseau par défaut 192.168.1.0/24 ou réseau choisi 192.168.32.0/24) afin de détecter certaines anomalies dans un réseau.

Condition de réalisation

Environnement de développement :

Pour effectuer cette mission, j'ai utilisé une interface graphique Linux Ubuntu 22.04.

(Joindre le schéma ainsi que l'exemple)

Organisation – Méthodes de travail :

Pour mener à bien cette mission, j'ai décidé de procéder en plusieurs étapes. Je faisais étapes par étapes et en plusieurs bouts de script les fonctions à réaliser et ensuite je mettais tous ces bouts de script en commun pour avoir un rendu final bien fonctionnel.

Réalisation

Afin de nous éviter de taper en ligne de commande la commande permettant de lancer un scan sur le réseau, ce script lancera automatiquement un scan suivant le paramètre choisi par l'utilisateur à la suite du script en ligne de commande (ex : AnalyseReseau.sh -m « -m » est le paramètre choisi en ligne de commande qui ajoutera les adresses mac au scan basique...).

(Voir le schéma ainsi que l'exemple DRIVE)

Tout d'abord, j'ai créé une boucle While pour créer les paramètres qui seront directement applicable en ligne de commande. J'ai aussi créé une condition afin de savoir si un paramètre est rentré en ligne de commande ou non, s'il n'y a pas de paramètre le script lance le scan par défaut avec l'adresse IP par défaut. (cf Drive, Capture Écran « ScriptWhileOPT»).

Ceci étant fait, j'ai fait des fonctions qui correspondent à chaque paramètre, par exemple si « -i » est rentré en ligne de commande, la fonction « interactive » se lance. (cf Drive Capture Écran, ScriptInteractive et resultatinteractive). Les différentes fonctions présente dans ce script sont « -h » pour la fonction Help, « -m » pour la fonction adresses MAC, « -i » pour la fonction interactif, « -v » pour la fonction verbose. (cf Drive, ExempleResultat)

Pour la fonction interactive, si l'utilisateur souhaite changer les paramètres trouver sur la carte réseau, il a fallu que je vérifie si l'adresse IP rentrée par l'utilisateur est une adresse IP valide, pour ce faire j'ai utilisé des Expressions régulières d'adresse IP.

Si une adresse IP est renseigné par l'utilisateur celle-ci doit être valide afin que l'utilisateur ne puisse pas rentrer une adresse IP qui n'existe pas. Si elle n'existe pas ou est non conforme, le script demande encore de rentrer une bonne adresse IP.

Pour la fonction qui ajoute les adresses MAC au scan de base, j'ai dû extraire les adresses IP avec la commande « `egrep -o` » ainsi qu'une expression régulière. J'ai aussi utilisé une expression régulière pour les adresses MAC et ensuite pour mettre les adresses IP à coter des adresses MAC j'ai dû utilisé la commande `awk` (cf Drive, Capture Ecran ScriptFonctionBASEetMAC)

Conclusion

Cette mission à été pour moi très enrichissante, car je n'avais jamais fais de script sur Linux jusqu'à maintenant. Cela m'a permis de découvrir un autre aspect de Linux et de développer mes compétences.