



SAE BUT 3 FA S5 : Développement d'une application

Rapport de SAE

Réalisé par :

BA Demba | MENDES Fredy | Paul Baudinot | Dorian Gaspart

Année universitaire 2023 - 2024=

Introduction :

Notre client, monsieur Fabrice Huguin, nous a contactés pour réaliser un projet dans le cadre de la SAE du semestre 5 de notre BUT Informatique. Le projet consiste à créer une interface web hébergée sur un serveur Linux permettant d'accéder à 3 modules. Le document ci-présent représente le cahier des charges.

1. Serveur Linux

1.1. Module ping

L'une des demandes du client est que l'interface web soit hébergée sur un serveur Linux et qu'il puisse être installé par le client. On remarque donc qu'un enjeu de portabilité se présente. Pour y répondre, nous avons choisi d'implémenter des conteneurs docker. L'idée sera de conteneuriser notre interface web avec docker pour qu'elle puisse être installée facilement par le client sur sa distribution.

Pour plus de détails voir le rapport d'installation.

2. Les Modules Utilisateurs

1.1. Module ping

1.1.1. Le fonctionnement global

Le premier module demandé par le client est une page permettant à un utilisateur du site de pouvoir envoyer des paquets à une adresse ip (ping) et de visualiser les résultats. Cette page permet de faire des ping d'un nombre de paquets ou des ping de manière continue. Il est également possible d'arrêter les pings à tout moment et de recommencer.

La page comporte des zones de saisie permettant d'inscrire l'adresse IP, le nombre de paquets à envoyer et de préciser si le ping devra s'effectuer de manière continue ou non.

Elle contient également une zone affichant les informations de chaque ping ainsi qu'une autre zone récapitulant les ping effectués jusqu'ici (le taux de réussite entre les paquets transmis et ceux reçus ainsi qu'une approximation du temps moyen de réponse).

Elle contient également une zone affichant les informations de chaque ping ainsi qu'une autre zone récapitulant les ping effectués jusqu'ici (le taux de réussite entre les paquets transmis et ceux reçus ainsi qu'une approximation du temps moyen de réponse).

1.1.2. Le détail de l'implémentation

La page principale du module de ping est *home/sae_but3_fa/src/ping.php*. Elle est la page où l'utilisateur renseigne les détails du ping. C'est une page php car elle appelle ensuite la

page `home/sae_but3_fa/src/ping_action.php` qui lance la commande ping et envoie chaque ligne d'output (chaque ping) au client.

Le script `ping_action.php` lance la commande dans un processus et récupère les output avec des pipe. PHP a été choisi pour réaliser cette partie car il est plus sécuritaire de faire l'exécution des commandes du côté serveur et PHP possède la fonction `exec()`.

Enfin, la page `ping.php` est également dépendante de `home/sae_but3_fa/src/script/scriptPing.js`. Ce script JavaScript gère la mise à jour des tableaux et espaces qui contiennent les détails des ping (les statistiques et leur détails). Nous avons choisi JS car cette technologie nous permet d'actualiser la page html sans devoir rafraîchir la page à chaque nouveau message du serveur vers le client.

1.2. Module IPv6

1.2.1. Le fonctionnement global

Le deuxième module permet à l'utilisateur de simplifier l'écriture d'une adresse IPv6 (qui possède plus de 128 bits) ou de l'écrire complètement. Elle est également capable de donner l'écriture binaire des 2 bits significatifs de l'adresse IP et de déterminer le type de l'écriture binaire de l'adresse.

1.2.2. Le détail de l'implémentation

La simplification de l'écriture d'une adresse IPv6 ne relevant que du traitement de texte, tout le module a été codé en JavaScript.

1.3. Module IPv4

1.3.1. Le fonctionnement global

Ce dernier module permet à l'utilisateur (via des formulaires) d'obtenir la division en sous-réseaux logiques à partir d'une adresse IP donnée et d'un masque écrit avec la notation CIDR lorsque l'utilisateur rentre les informations suivantes :

- Ip du réseau en question
- Mask du réseau
- Sous réseaux
 - Nom sous-réseaux
 - Nombre d'IP
 - Le nombre de sous réseaux voulus (afin de créer le nombre de zone de texte nécessaire, sous forme de volet roulant de nombre).

1.3.2. Le détail de l'implémentation

La partition de sous-réseaux ne nécessitant pas de faire des requêtes au serveur, tout le module a été codé en JavaScript.