**NOTE DE STAGE**

Ecrit par

**David Roufé**

**david.roufe@gmail.com**

27 juin 2022

SOMMAIRE

* Mise en contexte
  + Présentation de l’entreprise
  + Présentation du projet
  + Présentation de l’équipe
* Expression des besoins
  + Description de l’existant
  + Liste des besoins
  + Contraintes à respecter
  + Ressources utilisées
* Analyse
  + Cas d’utilisation
  + Spécification des IHM
  + Modèle de données
* Réalisation
  + Choix techniques
  + Conception du logiciel
  + Planning de réalisation
  + Exemples de codage
  + Organisation et réalisation des tests
* Conclusion

Mise en contexte

* Bsecure



Bsecure est un cabinet indépendant d’expertise en cybersécurité des applications et des infrastructures crée en 2009.

Bsecure possède une équipe de devSECops qui consiste à réaliser des audits de sécurité dans les entreprises clients, et les accompagne en remédiant aux problèmes de sécurité découverts lors de l’audit de sécurité.

Des pentesters et des logiciels de pentest, pour trouver des vulnérabilités, en avertir les clients et trouver des solutions à ces failles.

L’entreprise a été créé en 2009 à Paris et est partenaire avec cybermalveillance.gouv.fr

* Présentation du projet

L’entreprise utilise un logiciel de pentest afin de rechercher les failles systèmes et logiciels de ses clients(qui sont des entreprises).

Cet outil, InsightappSec lance un scan à travers une application, il suffit d’enregistrer la target (en écrivant le nom de domaine du site) pour commencer le scan.

L’objectif pour moi sera donc de coder un logiciel en utilisant la version API de InsightAppSec, d’utiliser une expression régulière (crontab) afin d’être au courant chaque minute du statut du serveur du site client, dès que celui-ci ne répondra plus, je devrai avertir à la minute les responsables.

* Equipe du projet

Nous sommes 4 personnes intervenant dans le projet :

-Moi

-Yehouda : camarade de classe présent dans le même stage que moi.

-Ouriel : Chef de projet

-Luc : Ingénieur informatique

Expression des besoins

* Description de l’existant

Pour l’instant, il n’y a pas de script permettant de réaliser mon projet, je devrai donc commencer de zéro.

* Liste des besoins

Le script doit devra :

-Être codé en JavaScript codé serveur (NodeJS)

-Être asynchrone pour permettre le multithreading de requêtes et optimiser les temps d’exécution

- Être codé dans une fonction AWS pour être serverless en utilisant le FAAS (Function As A Service)

-Être optimisé, synthétique pour réduire les pertes lors de l’exécution

-Être automatisé en utilisant une expression régulière (crontab) qui permettra au script de se relancer chaque minute de façon autonome.

-Utiliser un module permettant d’envoyer un mail aux ingénieurs avertissant d’un site tombé en panne à cause d’un scan.

-Utiliser des callbacks et des Promises.

-Informer du statut du scan, et la cause de la mise hors service du site.

* Contraintes à respecter

Comme les employés dans l’entreprise travaillent tous en distanciel, j’ai travaillé de chez moi.

Cela a résulté en quelques contraintes :

-Contrainte d’accompagnement car je ne disposais pas d’aide physique pour m’aider à résoudre des problèmes dans mon code, je devais faire plus de recherche et lire plus de documentations.

-Contrainte de concentration car j’estime que travailler en équipe en présentiel est plus efficace et formateur, alors que chez soit on est plus distrait, je devais donc rester complétement concentré pour mener à bien ma mission.

* Définition des ressources matérielles et logicielles

J’ai disposé de ces ressources matérielles et logicielles lors de mon stage :

* Mon ordinateur portable personnel fonctionnant sous Windows 11.
* Accès à un compte AWS permettant de réaliser une fonction lambda.
* Une clé d’API permettant l’échange de données.
* L’éditeur de code Sublime Text pour réaliser mon script.

Analyse

Comme mon projet n’étant pas de créer un site, il n’est donc pas possible de réaliser toute la partie Analyse de la note de synthèse.

Mon professeur réfèrent m’a donc conseillé d’aménager ma synthèse de façon que chaque demande du maître de stage soit associée à une réalisation.

Comme, je n’ai eu qu’une seule demande, mais qui reste conséquente, je vais vous la présenter dans les plus précis détails.

Une analyse du projet a été nécessaire avant la réalisation du script.

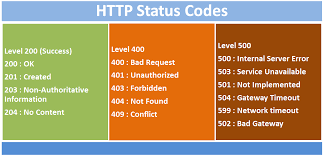
Et pour cela je devais répondre à un problème s’interposant devant moi.

En effet, l’utilisation de méthodes de pentest et de scans d’un site internet entraine souvent une panne du site scanné, pourquoi ?

Parce qu’il existe des méthodes de scans intenses qui consistent à utiliser n’importe quel procédé pour vérifier s’il existe des vulnérabilités ou des failles dans le site.

Ceci peut entrainer une mise hors service du site scanné avec l’erreur fatale du site : NETWORK\_UNAVAILABLE et un code d’erreur HTTP 404.

Du a une surcharge de requête HTTP et ICMP (Ddos)



L’équipe se chargeant du pentest peut donc faire tomber le site de ses clients

Causant une perte de gains pour l’entreprise client , une mauvaise image pour le client et pour l’entreprise de pentest.

Mon entreprise a donc tout intérêt à être averti à la minute lorsqu’un scan fait tomber un site, c’est là que j’interviens !

Dans les grandes lignes, je devrais communiquer en API avec le service web de scan, et je devrais automatiser l’exécution de mon script toutes les minutes afin de vérifier si un scan a provoqué la chute d’un site, mon script devra alors avertir mes supérieurs et en plus spécifier la raison de la mise hors ligne du site.

Nous décortiquerons la manière de réalisation du script dans la partie suivante.

Réalisation

* Choix techniques

Langage utilisé : JavaScript coté serveur (NodeJs)

Editeur de code : Sublime Text

Plateforme API : Postman

Modules utilisés : HTTPS et HTTP pour faire des requêtes GET, POST… afin de récupérer le format et le convertir en Json.

La plateforme utilisée pour coder mon script était AWS.

AWS est un service spécialisé dans le cloud computing et offrant des ressources informatiques en fonction de la demande.



* Conception du script

Afin de réaliser mon script, j’ai d’abord organisé ma méthode de travail.

En effet j’ai dû d’abord apprendre JavaScript en profondeur, ensuite les appliquer dans mon projet.

L’apprentissage de l’asynchrone, présent dans JavaScript était quelque peu déroutant pour moi.

En effet, étant habitué à des langages comme Python, PHP ou Java qui sont des langages synchrones, la compréhension de l’asynchrone avec JavaScript n’a pas facilité la tâche.

J’ai donc alterné séance d’apprentissage et séance de pratique pour réaliser mon projet.

Plusieurs scripts ont été codés lors du projet :

* statusCode.js

Premier script, au début du stage, permettant de vérifier l’état d’un site en envoyant une requête et en vérifiant le statut de réponse, si celui-ci est dans les 200, il répond, si celui-ci renvoyait un code dans les 400, alors il ne répondait pas.

Ce premier script m’a aussi permis de m’initier au JavaScript.

* domains.js

Script réalisé en utilisant une API servant de passerelle entre mon script et le logiciel InsightAppSec, me permettant de récupérer les informations comme les noms de domaines des sites qui ont été scannés par l’entreprise.

* running.js

Script permettant de récupérer tous les sites qui sont en train d’être scannés par l’entreprise, récupérer aussi les scans qui ont échoués et la raison pour laquelle le scan n’a rien donné

* failedScans.js

Script qui retrace tous les scans qui ont échoué et la raison pour laquelle ils ont échoué.

Par exemple un scan intense peut entrainer la mise hors service d’un site entrainant un statut 404 avec un code NETWORK\_UNAVAILABLE.

Une image contenant texte

Description générée automatiquement

Dans ce script , on voit tous les sites qui sont tombés.

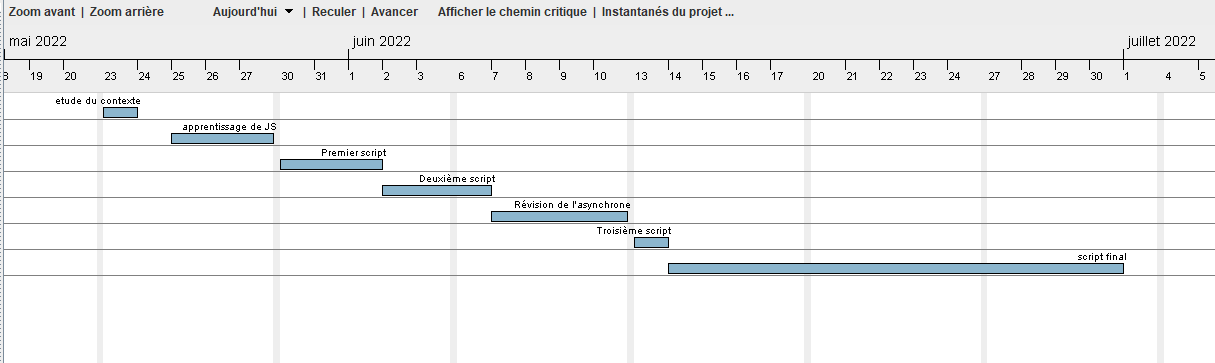
NETWORK\_UNAVAILABLE : le site a crashé du à l’intensivité du scan

CANCELED : Le scan a été annulé

ENGINE\_UNAVAILABLE : La plateforme a des difficultés à identifier le moteur d’analyse

BAD\_AUTH : Mauvaise authentification

* Planning de réalisation



* Codage de l’application.

Le script comporte une centaine de lignes, je vais donc montrer étape par étape mon code

Une image contenant texte

Description générée automatiquement

Les 12 premières lignes permettent d’importer le module https, module qui offre de nombreuses fonctionnalités, notamment d’effectuer des requêtes http depuis un service web et de récupérer la réponse.

Ici on spécifie le site, le port, le chemin, la méthode et les en-têtes, nécessaires pour utiliser correctement l’Api (clé API nécessaire).

Une image contenant texte

Description générée automatiquement

Dans cette fonction getFailedScans qui prend en paramètre les options (vu dans la première image).

La fonction permet dans les grandes lignes de récupérer le résultat de la requête et de stocker dans un dictionnaire, les applications (objet contenant le nom de domaine) ayant un scan avec un statut de « Failed » (raté) et la raison de l’échec du scan.

On utilisera un objet Date au milieu du code qui permettra de vérifier si la date du scan qui a crash est inférieur à une minute car notre but c’est d’avertir à la minute si l’un des scans réalisés par l’entreprise a fait tomber un site client.

Une image contenant texte

Description générée automatiquement

Une fonction asynchrone (reconnaissable par le mot clé async) permet d’utiliser la fonctionnalité await qui va attendre le résultat de la fonction getDomainNames avant de passer à la ligne suivante, cela permet donc un traitement synchrone pour cette ligne seulement.

La fonction getDomainNames utilise une Promesse qui va traduire les applications en nom de domaine.

Une Promesse est un objet permettant de réaliser un traitement asynchrone.

Cet objet est donc très utile surtout lors de requêtes ou des interactions avec un serveur web car le programme pourra effectuer plusieurs actions en même temps sans attendre la fin d’un bout de code afin de passer à la ligne suivante.

C’est le multithreading.

Par souci de confidentialité, je ne pourrais pas vous montrer le résultat de mon programme.

* Organisation et réalisation des tests.

Les tests ont été réalisé sur la plateforme AWS Lambda, qui est un service sans serveur et permet d’exécuter du code pour tout type d’application.  
La plateforme, le langage et le code permettent une optimisation totale et une grande efficacité, JavaScript coté serveur (NodeJS) est un langage très scalable, dynamique et rapide.



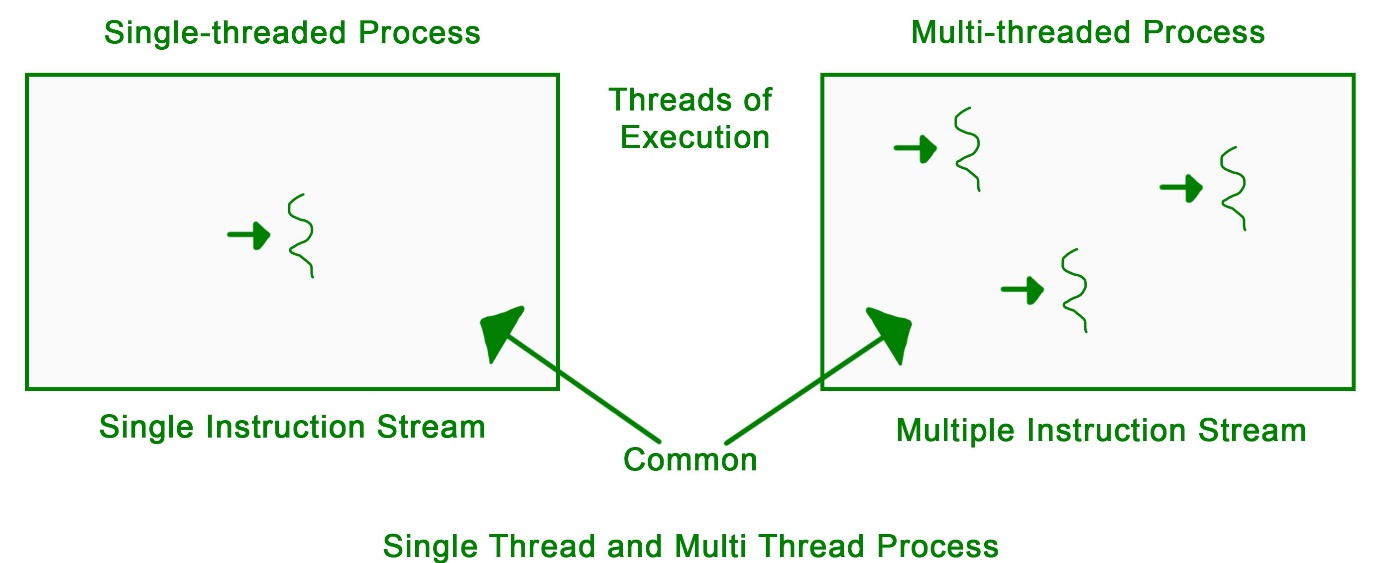
Le modèle de cloud computing utilisé pour les besoins de mon entreprise étaient le FAAS (function as A Service) qui permet aux utilisateurs de développer des applications et de déployer des fonctionnalités sans gérer de serveur, ce qui améliore l'efficacité du processus. Le concept est également connu sous l'appellation [serverless](https://www.lemagit.fr/definition/Serverless-computing).

En outre, ce système est intéressant pour les entreprises car il utilise le strict nécessaire des ressources informatiques en fonction de la demande, il y a ainsi le moins de pertes possible.

De plus, l’utilisation de NodeJs permet du multithreading c’est la capacité à effectuer plusieurs tâches en même temps, en parallèle et bien sûr.

Single thread c’est le fait que de ne s’occuper que d’une tâche à la fois.

L’utilisation de la version 8 du moteur de compilateur JavaScript de Google, compile en code natif et fonctionne très rapidement.



Les tâches courantes des applications web, comme la lecture ou l’écriture dans la base de données, la lecture et l’écriture dans les connexions réseau et la lecture ou l’écriture dans le système de fichiers, peuvent être exécutées rapidement à l’aide de Node.js.

Node.js aide les entreprises à créer des applications réseau rapides et évolutives, capables de gérer des connexions simultanées à haut débit. Node.js ne ralentit rien ; il suffit aux développeurs d’écrire le code et les applications Node.js se chargent de faire le bon choix.

CONCLUSION

Durant ce stage, de nombreux problèmes se sont interposés, comme NodeJS, que je n’avais jamais vu auparavant.

De plus, la syntaxe était tout autant déroutante, différente de tous les langages que j’avais rencontré.

Mais la persévérance, les cours et la documentation m’ont permis de m’habituer à des concepts nouveaux comme les Callbacks, les Promise et l’asynchrone en général.

Le service que j’ai réalisé est aujourd’hui utilisé par mon entreprise et permet de les alerter à la minute qu’un site client tombe.

Cependant, mon script pourra toujours être optimisé, amélioré, et je serai ravi d’effectuer mon stage de deuxième année dans la même entreprise.

Etant passionné par la cybersécurité, ce stage a amorcé mon expérience dans l’informatique et plus particulièrement dans la cybersécurité et ses enjeux, la sécurité et la confiance étant selon moi les deux critères fondamentaux pour construire ce type d’entreprise.