Faculté
des Sciences
Aix*Marseille Université

ALTERFRAME

Rapport de stage de fin d'études

Sécurité, CI & développement Java

Alexandre Léonardi

M2FSIL-FSI

Année 2016/2017

Encadrants : Clément Fleury &
Idir Meziani

Enseignant : Emmanuel Godard

Table des matières

Introduction

Développement Java et développement d'une solution d'analyse statique de sécurité : ce sont les deux branches de mon stage. Il s'agit pour partie de prendre part aux contrats en Java d'Alter Frame, l'entreprise qui m'accueille pour la durée du stage, et d'autre part d'intervenir sur un projet en interne visant à mettre en place une analyse de sécurité systématique des projets Web au-travers de pratiques de CI [1]. La présentation d'Alter Frame sera donc naturellement la toute première partie de ce rapport.

Ce sujet a l'avantage d'être ouvert et diversifié. Il me permet d'une part de travailler sur du pur développement et d'autre part de mettre en pratique la composante sécurité de la formation FSI [2], tout en découvrant les concepts de CI qui m'étaient jusque là étrangers, ainsi que des technologies qui vont de pair telles que Docker. Présenter ces deux pans de mon travail à Alter Frame composera la suite du rapport.

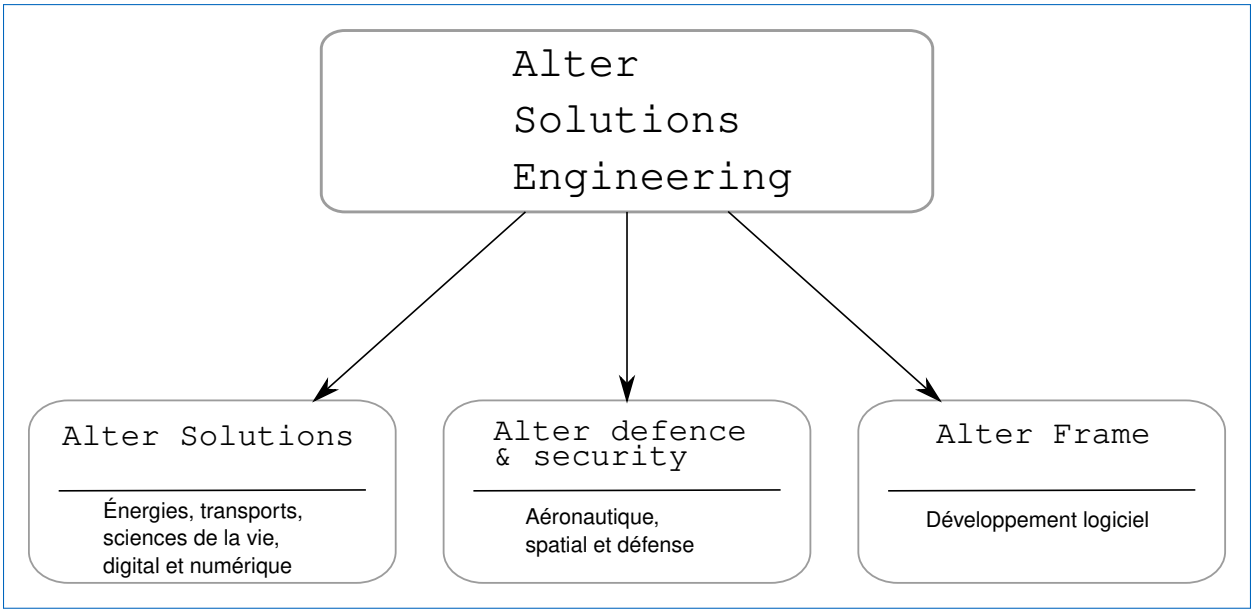
Celui-ci se cloturera en analysant et résumant les apprentissages que j'ai retiré de ce stage, et les perspectives d'avenir qu'il m'ouvre.

À noter que, par discrétion à leur égard, les noms des clients d'Alter Frame ne seront pas mentionnés et seront effacés des captures d'écran que vous trouverez dans ce document. Il en ira de même pour les différents projets et les noms de personnes physiques.

1. Continuous Integration ou intégration continue, cf. `https://fr.wikipedia.org/wiki/Int%C3%A9gration_continue`
2. Fiabilité et sécurité informatique, cf. `http://masterinfo.univ-mrs.fr/FSI.html`

Rapport de synthèse

Graphique 1 – Alter Solutions Engineering et ses filiales



1 Présentation d'Alter Solutions Engineering

Alter Solutions Engineering, et plus particulièrement sa filiale Alter Frame, est l'entreprise qui m'a accueilli pour la durée de mon stage de fin d'études, nous allons donc commencer par la présenter rapidement.

1.1 Les subdivisions d'Alter Solutions Engineering et leurs secteurs d'activité

Alter Solutions Engineering est une entreprise relativement jeune : elle a été créée en 2006 et, si elle n'entre plus maintenant dans la catégorie des PME en termes de nombre de collaborateurs, elle reste une structure de petite taille.

Le siège social de l'entreprise se trouve à Versailles et c'est là où travaille l'équipe de développement française dont je fais partie. En pratique, il s'agit de l'équipe de développement d'Alter Frame qui est une entité enfant d'Alter Solutions Engineering (cf. section 1.2).

Alter Solutions Engineering est une société de conseil en hautes technologies mais en pratique, elle est composée de trois filières qui ont chacune une spécialité bien distinctes (cf. graphique 1 et graphique 2).

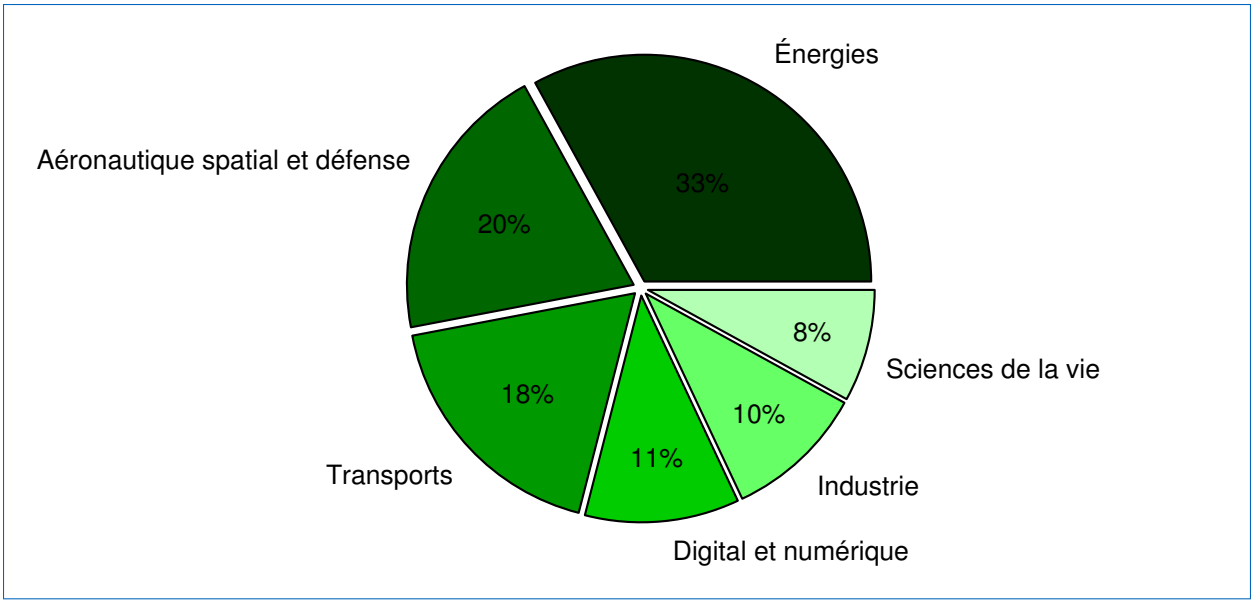
1.1.1 Alter Solutions

Cette filiale est spécialisée dans le conseil en ingénierie, notamment dans les domaines de l'énergie, des transports, des sciences de la vie, du digital et du numérique.

1.1.2 Alter defence & security

Alter defence est également orientée vers le conseil, mais cette fois plus particulièrement dans l'aéronautique, le spatial et la défense.

Alter Defence and Security est également la filiale d'Alter Solutions Engineering qui m'accueillera une fois mon stage terminé (cf. section **??**). L'objectif du groupe Alter au-travers de ce stage est de me donner le



Graphique 2 – Répartition de l'activité des différentes filiales d'Alter Solutions Engineering

bagage technique et l'entraînement nécessaires pour pouvoir à terme me déployer comme expert technique auprès de clients de l'entreprise, or les missions de cyber-sécurité sont le domaine d'activité de Defence & Security.

1.1.3 Alter Frame

Alter Frame enfin est la branche spécialisée dans l'édition de logiciels et celle que j'ai rejoint durant mon stage. C'est une ESN [3] dont l'activité est elle-même répartie en deux catégories :

- le conseil, c'est-à-dire le fait de fournir des spécialistes d'un domaine du numérique pour la durée d'un contrat à un client ;
- le développement de logiciels au forfait, c'est-à-dire le fait de prendre commande d'un logiciel à réaliser en interne et de le livrer à la fin du contrat.

1.2 Un peu plus de détails sur Alter Frame

Bien qu'Alter Frame ait des clients et des domaines d'intervention variés, en termes de technologies il y a trois pôles de compétences qui sont caractéristiques de l'entreprise et reviennent le plus régulièrement :

- Java ;
- .NET ;
- PHP.

Bien que mon stage se soit divisé en deux grands axes, mon travail a été dans tous les cas lié au pôle de développement Java et au responsable technique sous la direction de qui j'ai travaillé. En conséquence j'ai participé à plusieurs projets Java de manière anecdotique, en plus du projet principal que je détaillerai en partie 2.

Quelques projets notables d'Alter Frame, mais sur lesquels je n'ai pas eu l'occasion de travailler :

3. Entreprise de Services du Numérique, cf. `https://fr.wikipedia.org/wiki/Entreprise_de_services_du_num%C3%A9rique`

- interface de *monitoring* de plateformes pétrolières pour tablettes ;
- plateforme web permettant d'accéder facilement à des exécutables des différents projets d'Alter Frame, à destination des commerciaux des autres branches de la compagnie ;
- VOIR SI JE PEUX RAJOUTER QUELQUES TRUCS INTÉRESSANTS

2 Détail du sujet de stage

Le sujet de mon stage était ainsi formulé : « Intégration de tests de sécurité dans le processus d'intégration continue, et développement Java selon les besoins de l'entreprise. »Nous avions convenu, lors de l'entretien d'embauche, que mon travail serait également réparti entre ces deux aspects.

En pratique, cela a représenté plusieurs projets différents et une intervention en tant que consultant.

2.1 Sécurité & CI

La partie la plus précisément définie de mon stage : Alter Frame dispose d'un système d'intégration continu qui, à mon arrivée, incorporait de l'analyse qualité et la compilation du code à chaque *push* sur leur plateforme git.

Mon travail serait donc d'incorporer un aspect sécurité à la configuration déjà existante, de manière à obtenir le processus du graphique 3.

Éléments à implémenter :

- analyse dynamique, uniquement dans le cas d'application web, en automatisant des tests de sécurité avec ZAP ;
- analyse statique, si le temps le permettait, en réutilisant les analyses de code *via* Sonar déjà en place et les affinant d'un point de vue sécurité.

Ces tâches de départ dépendent de plusieurs autres qui faisaient, de fait, également partie de mon sujet de stage. Réaliser des analyses de sécurité contre les applis web d'Alter Frame impliquaient que celles-ci soient accessibles en ligne, au minimum sur un serveur privé pour les besoins du développement. Automatiser ce déploiement, et idéalement pouvoir l'étendre pour réaliser une livraison dans certaines conditions (telles qu'un *push* tagué), devrait donc être la première étape à réaliser, avant d'ensuite programmer les tests, ZAP ayant le bon goût d'être très finalement contrôlable par des APIs dans plusieurs langages [4] et en ligne de commande [5] [6].

Par ailleurs, si la partie analyse statique devait se réaliser, cela impliquait d'apprendre le fonctionnement des plugins sonars et comment étendre les règles créées par défaut par l'analyseur, pour ensuite à proprement parler isoler des problématiques de sécurité qui peuvent être adressées et les faire vérifier.

2.2 Développement Java

L'aspect Java, dans mon stage, était plus flou dans la mesure où il était sujet aux projets qui seraient en cours et en besoin de soutien au moment de mon arrivée dans l'entreprise. En pratique, cela a été majoritairement du développement sur un projet de gestions de tests sur des voitures, pour un constructeur automobile, ainsi que des interventions ponctuelles sur plusieurs autres projets destinés au même client.

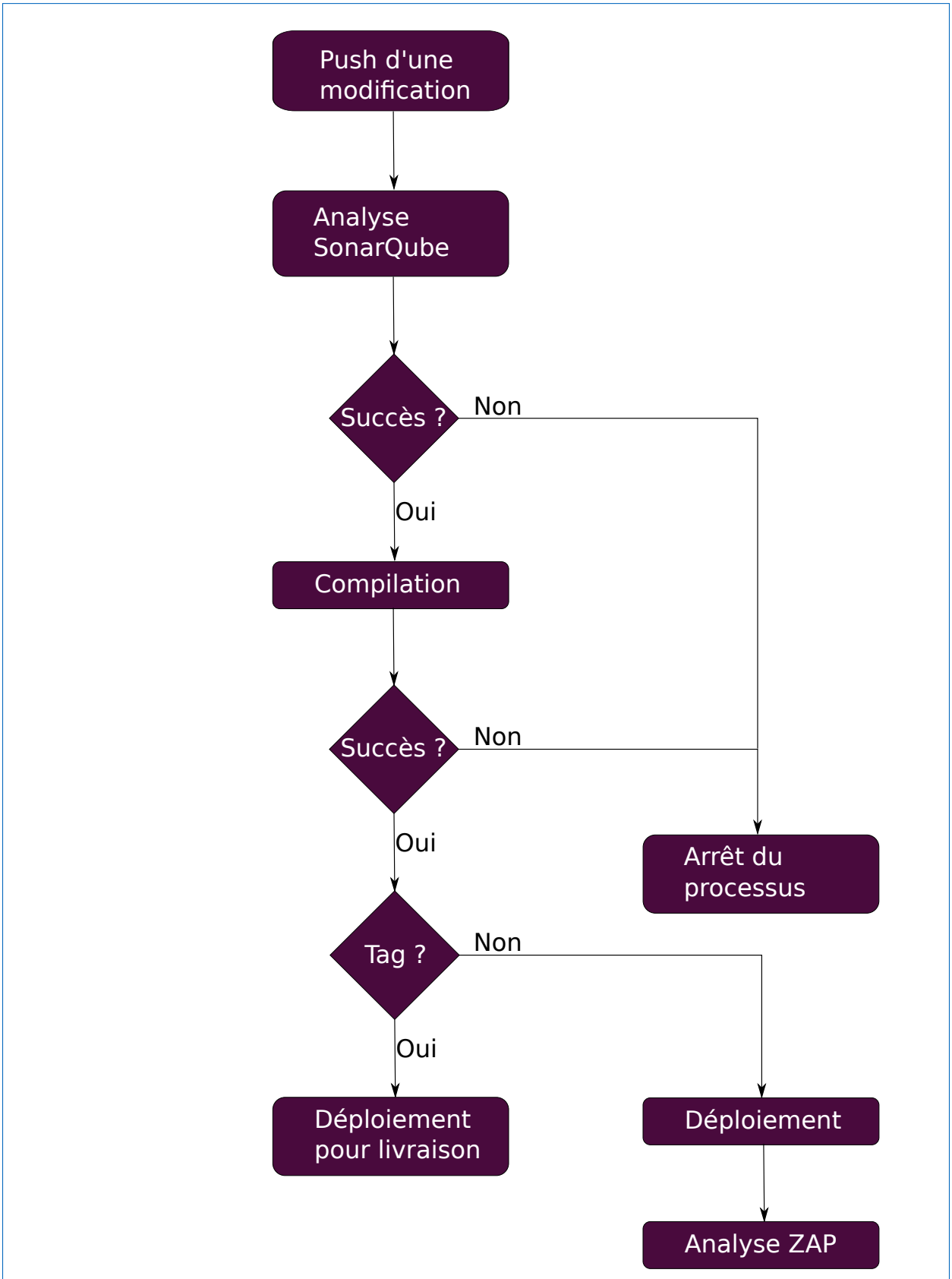
2.3 Interventions en fonction du besoin

De même que le développement Java, cette partie de mon travail était sujette à évolution en fonction du besoin. En fin de stage, elle a pris la forme d'un audit technique pour une compagnie d'assurance qui souhaitait améliorer son service d'Intranet, tant d'un point de vue sécurité que qualité de code ou performances d'exécution.

Durant cet audit j'ai eu à gérer une partie de l'aspect sécurité de notre intervention, et une partie de l'aspect performance, cela au cours d'une mission de 3 jours chez le client.

4. `https://github.com/zaproxy/zaproxy/wiki/ApiDetails`
5. Options de ZAP en CLI : `https://github.com/zaproxy/zap-core-help/wiki/HelpCmdline`
6. Un wrapper pour utiliser l'API Python à travers la ligne de commande, non-officiel : `https://github.com/Grunny/zap-cli`

Graphique 3 – Déroulement du processus d'intégration continue



3 Synthèse du travail effectué

justifier les solutions retenues, décrire l'environnement et la méthodologie, présenter les difficultés rencontrées

Rapport technique

4 Sécurité & intégration continue

La seconde partie de mon stage a été majoritairement consacrée à l'amélioration du processus de CI au sein d'Alter Frame. Il y a beaucoup à dire sur le sujet, tant en fait que je n'ai fait qu'effleurer la surface de ce qui est possible, d'une part parce qu'il s'agit d'un secteur de l'informatique encore jeune et d'autre part parce que je me suis naturellement concentré sur l'aspect "sécurité" qui est loin d'être le seul intérêt de l'intégration continue.

4.1 Le contexte : GitLab-CI

L'intégration continue dans les projets d'Alter Frame se fait à l'aide d'un service proposé par la plateforme d'hébergement de projets informatiques GitLab [7]. Le service en question, GitLab-CI [8], propose de mettre en place de l'intégration continue sur les projets hébergés sur GitLab.

Au moment de mon arrivée chez Alter Frame la partie CI des projets consistait majoritairement en la compilation des projets et une analyse de code à l'aide d'un plugin SonarQube [9], mise en place depuis environ 2 ans.

Le principe est que les actions décrites ci-dessus, compilation et analyse de code, sont effectuées à chaque push sur le serveur GitLab. Ce fonctionnement peut ensuite être affiné, pour ne se produire que lorsqu'un tag git est pushé ou sur certaines branches (branche master, tag de release, etc).

Il n'y avait néanmoins pas de composante cybersécurité dans le processus de CI d'Alter Frame et c'est donc ce sur quoi je suis intervenu en priorité. Néanmoins, mon travail ne s'est pas limité à cela et j'ai aussi pu intervenir sur d'autres aspects du CI et améliorer l'existant.

4.2 Les outils

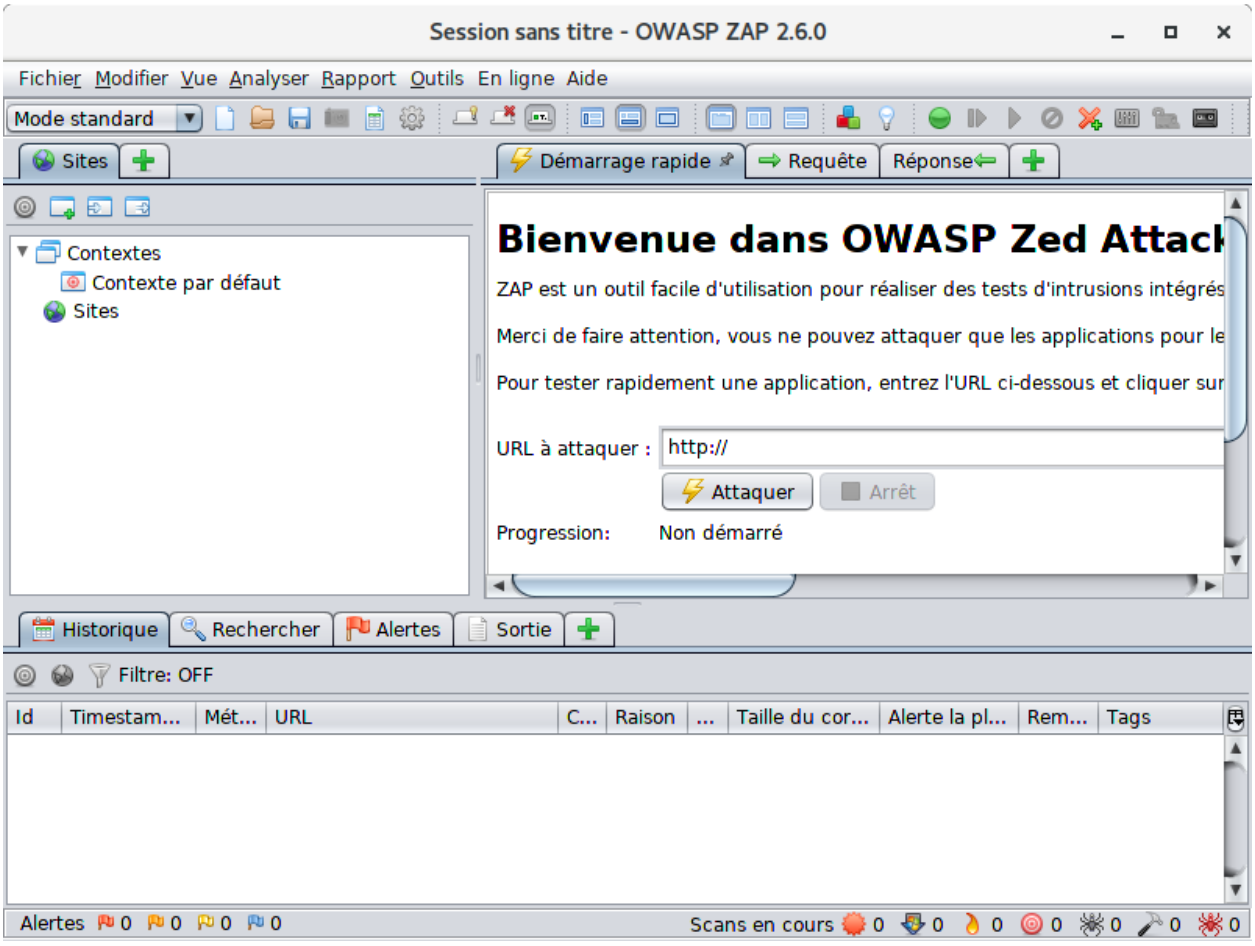
4.2.1 ZAP : Zed Attack Proxy

ZAP [10] (voir figure 4) est un projet open source développé par l'OWASP. Il s'agit un proxy qui peut intercepter et analyse le trafic qui traverse la machine hôte. ZAP est un outil de sécurité très intéressant et ce pour un grand nombre de raisons :

- activement développé [11] ;
- open source et cross-platform ;
- OWASP est une référence dans le monde de la sécurité ;
- une large communauté, et donc une grande quantité de ressources sur laquelle s'appuyer ;
- ZAP est contrôlable en ligne de commande/*via* des APIs en plusieurs langages.

```
1 \$ zap.sh -cmd -help
2   zap.sh [Options]
3
4 Options de base:
5   -newsession <path>        Cree une nouvelle session a la position specifiee
6
7   -session <path>           Ouvre la session specifiee apres le demarrage de
      ZAP
8
```

7. https://about.gitlab.com/
8. https://about.gitlab.com/features/gitlab-ci-cd/
9. https://www.sonarqube.org/
10. https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project
11. Plus de 60 commits en juin 2017, voir https://github.com/zaproxy/zaproxy



Graphique 4 – Fenêtre de démarrage de ZAP

```
  -host <host>              Ecrase l'hote du proxy designe dans le fichier de
    configuration

  -port <port>              Ecrase le port du proxy designe dans le fichier de
    configuration


Options pour accessoires:
  -script <script>          Script a demarrer en ligne de commande ou a charger
    dans l'interface graphique
  -addoninstall <addon>     Installer l'accessoire specifie depuis la foire aux
    modules ZAP
  -addoninstallall          Installer tous les accessoires disponibles depuis
    la foire aux modules de ZAP
  -addonuninstall <addon>   Desinstalle l'accessoire specifie
  -addonupdate              Mettre a jour tous les accessoires modifies depuis
    la foire aux modules de ZAP
  -addonlist                Afficher tous les accessoires installes
  -quickurl [target url]: L'URL a attaquer, p.ex. http://www.example.com
  -quickout [output filename]: Le fichier dans lequel ecrire les resultats XML
  -quickprogress: Afficher les barres de progression pendant le balayage
```

Listing 1 – Options de ZAP en ligne de commande

Je n'avais, avant mon stage, que briévement eu l'occasion d'utiliser ZAP, au-travers du sous-projet de tests d'intrusion avec M. Pachy. Pouvoir m'entrainer plus longuement avec représentait donc à la fois un intérêt personnel, car cela me permettait d'en apprendre plus sur les vulnérabilités web les plus répandues, et professionnel car c'est un outil dont l'usage pourrait être pertinent pour mes futurs emplois.

4.2.2 Docker

Docker [12] est une technologie de virtualisation basée sur des conteneurs, qui vient se place en opposition aux hyperviseurs et machines virtuelles [13]. En plus d'une charte graphique à base de faune marine des plus plaisantes [14], la technologie Docker présente plusieurs fonctionnalités qui la rendent intéressante dans le monde de l'industrie informatique :

- un conteneur est plus léger qu'une VM ;
- un conteneur s'exécute de la même façon sur n'importe quelmachine où Docker est installé ;
- un conteneur peut embarquer toute la configuration nécessaire au bon fonctionnement de l'application, et c'est là le point le plus important. L'étape de configuration de l'environnement n'a à être effectuée qu'une seule fois, à la création de l'image [15]. De plus le système de Docker Store [16], proche de celui d'un gestionnaire de paquets, permet au client d'avoir facilement la dernière version possible d'un logiciel, encore une fois en s'abstractisant des changements de configuration qui vont avec la mise-à-jour.

On assiste donc à une généralisation de l'utilisation de Docker depuis sa première version en 2013, avec de nombreux cas d'utilsation [17], mais aussi à une multiplication des outils en lien avec la technologie Docker comme des outils de gestion de groupes de containers [18].

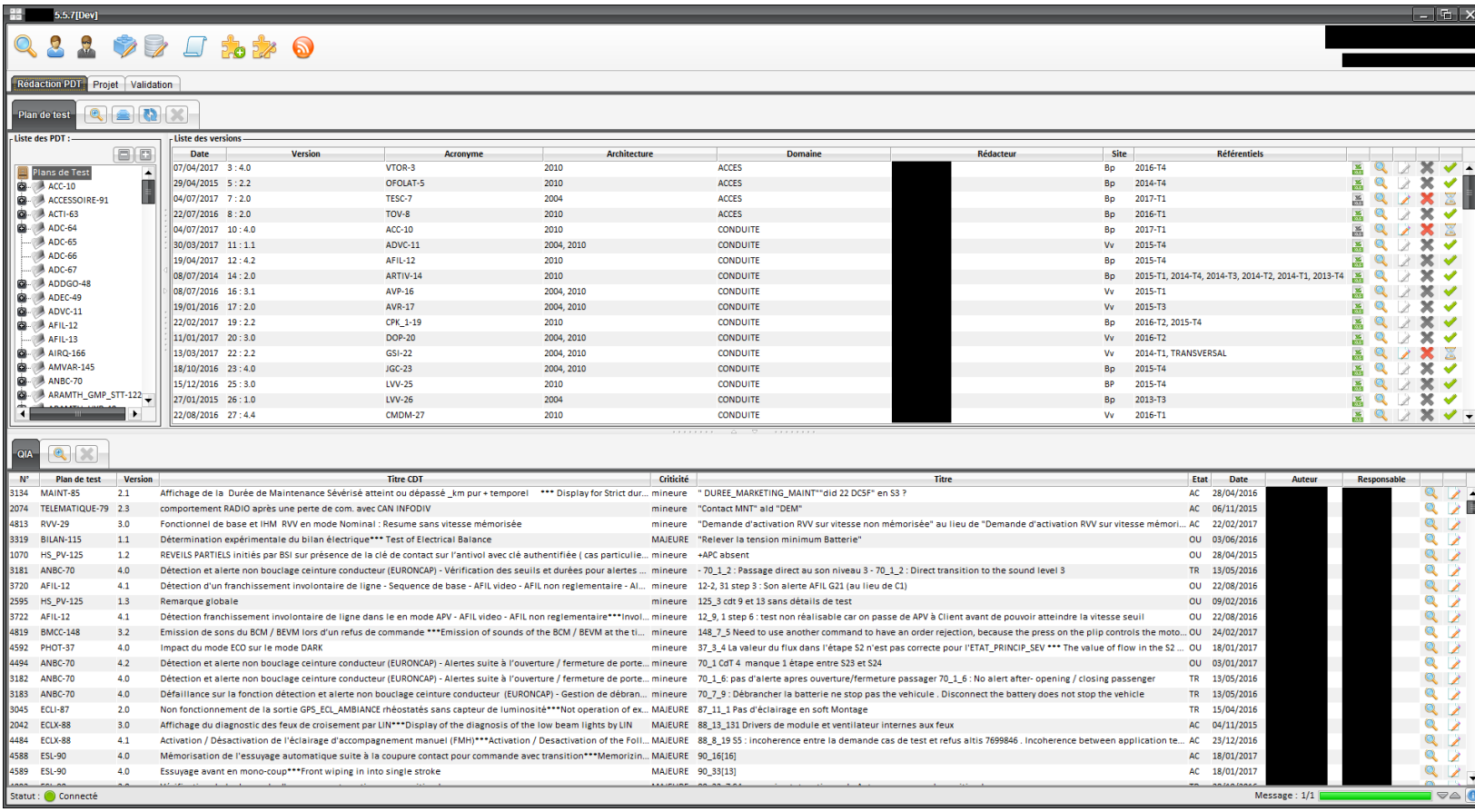
12. `https://www.docker.com/`
13. Ou VMs pour Virtual Machines
14. `https://www.docker.com/sites/default/files/group_5622_0.png`
15. On ne parle de conteneur qu'une fois l'image en cours d'exécution, cf. différence entre processus et programmme
16. `https://store.docker.com/`
17. `https://www.airpair.com/docker/posts/8-proven-real-world-ways-to-use-docker`
18. e.g. Kubernetes, Docker Swarm

4.2.3 YAML

YAML Ain't Markup Language [19], de son nom complet, est un "standard de serialisation de données".

4.3 Mon action sur le sujet

19. http://yaml.org/



Graphique 5 – Fenêtre de démarrage de l'outil

5 Développement Java : Outil de gestion de tests pour un constructeur automobile

La pemière partie de mon stage a été occupé par beaucoup de développement en Java. L'idée de cette part du stage était de participer aux contrats remplis par Alter Frame et avoir ainsi une idée précise d'à quoi ressemblait le travail dans l'entreprise. Je vais profiter de cette partie pour résumer les projets auxquels j'ai participé et dans quelle mesure, sans pour autant le commanditaire dans un soucis de discrétion.

5.1 Le contexte

Produire et mettre en vente une voiture requière que celle-ci soit passée par une batterie de tests intensifs. Ce premier projet était un logiciel permettant de gérer ces tests de bout en bout : ajouter un véhicule à la base de données, établir une liste d'organes à tester, une liste de tests pour chaque organe, puis ensuite enregistrer les résultats des tests qui ont été effectués (voir graphique 5). Cela représente le cœur de la logique métier impliquée dans le logiciel, mais naturellement il y avait autour de ce noyau de nombreuses fonctionnalités plus "classiques" telles que la gestion d'authentification des utilisateurs, les différents privilèges (administrateur, utilisateur simple), etc.

De manière intéressante, j'ai remarqué en travaillant sur le workflow des tests automobile implémenté dans cet outil que celui-ci est semblable au cycle en V qui fait partie intégrante de la gestion de projet en informatique. Les détails étaient, naturellement, très différents mais ce qui semblait être les grandes

lignes de la façon dont un projet de tests devait être géré était au contraire très proche de ce que nous connaissons dans le monde de l'informatique.

Une particularité de ce projet est qu'il s'agit d'un logiciel déjà existant qui a été récupéré par Alter Frame pour de la mise à jour et de la maintenance. L'ESN initialement en charge du projet avait perdu le contrat et mon travail a, de ce fait, été majoritairement constitué de correction de bugs et, dans une moindre mesure, d'ajout de fonctionnalités mineures (voir section 5.3).

5.2 Environnement technique

- *Java 7 :* Le projet étant vieux de plusieurs années déjà, il utilise la version de Java qui était en vigueur à l'époque de sa genèse à savoir Java 7.
- *Java Swing :* Pour les mêmes raisons que Java 7, le framework utilisé pour la partie graphique de l'application est Java Swing [20].
- *Apache ActiveMQ*[21] *:* L'outil fonctionne selon une architecture client-serveur classique et la communication entre le serveur et les différents clients se fait au-travers d'un système de queues et du système open source ActiveMQ.
- *MyBatis :*[22] La liaison base de données/application se fait grâce à cet ORM open source qui a la particularité de mapper des méthodes Java à des requêtes SQL [23].

5.3 Fonctionnalités ajoutées

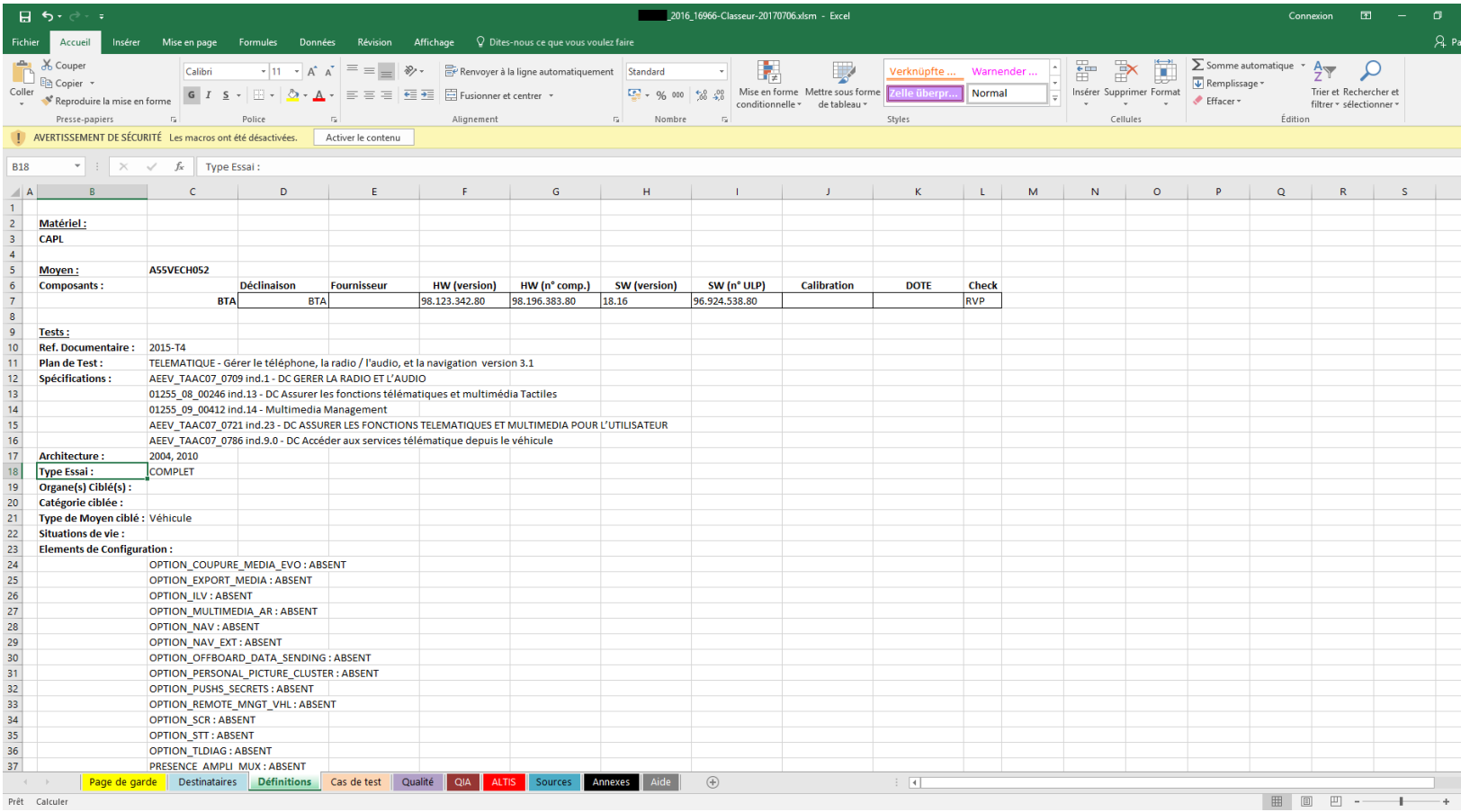
Comme mentionné plus haut, le gros du travail sur cet outil consistait à reprendre et débugger un code au départ écrit par une autre société. Il s'agissait donc tout à la fois de trouver des erreurs, améliorer ce qui pouvait l'être (notamment en termes de performances) et comprendre intrinséquement le fonctionnement de l'application. Tout cela représente un temps de développement considérable.

Les fonctionnalités ajoutées sont donc, comparativement, peu nombreuses. Ma première réalisation originale sur ce projet a été de modifier la génération de tableau Excel (voir figure 6) pour prendre en compte de nouvelles spécifications.

La génération en elle-même existait déjà à ce moment à l'aide de Microsoft Visual Basic, Scripting Edition [24]. Ce qui était demandé par le client était d'ajouter une nouvelle catégorie d'informations à ce classeur, à savoir les *Situations de vie*.

INSÉRER UNE EXPLICATION ET DES EXTRAITS DE CODE

20. https://en.wikipedia.org/wiki/Swing_(Java)
21. http://activemq.apache.org/
22. http://www.mybatis.org/mybatis-3/
23. Comparaison de MyBatis à JDBC et Hiberte : https://blog.zenika.com/2012/03/28/presentation-de-mybatis/
24. VBScript pour les intimes, cf. https://support.microsoft.com/en-us/help/198703/how-to-automate-excel-from-a-client-side-vbscript



Matériel :										
CAPL										
Moyen :	**A5SVECH052**									
Composants :		**Déclinaison**	**Fournisseur**	**HW (version)**	**HW (n° comp.)**	**SW (version)**	**SW (n° ULP)**	**Calibration**	**DOTE**	**Check**
	BTA	BTA		98.123.342.80	98.196.383.80	18.16	96.924.538.80			RVP
Tests :										
Ref. Documentaire :	2015-T4									
Plan de Test :	TELEMATIQUE - Gérer le téléphone, la radio / l'audio, et la navigation version 3.1									
Spécifications :	AEEV_TAAC07_0709 ind.1 - DC GERER LA RADIO ET L'AUDIO									
	01255_08_00246 ind.13 - DC Assurer les fonctions télématiques et multimédia Tactiles									
	01255_09_00412 ind.14 - Multimedia Management									
	AEEV_TAAC07_0721 ind.23 - DC ASSURER LES FONCTIONS TELEMATIQUES ET MULTIMEDIA POUR L'UTILISATEUR									
	AEEV_TAAC07_0786 ind.9.0 - DC Accéder aux services télématique depuis le véhicule									
Architecture :	2004, 2010									
Type Essai :	COMPLET									
Organe(s) Ciblé(s) :										
Catégorie ciblée :										
Type de Moyen ciblé :	Véhicule									
Situations de vie :										
Elements de Configuration :										
	OPTION_COUPURE_MEDIA_EVO : ABSENT									
	OPTION_EXPORT_MEDIA : ABSENT									
	OPTION_ILV : ABSENT									
	OPTION_MULTIMEDIA_AR : ABSENT									
	OPTION_NAV : ABSENT									
	OPTION_NAV_EXT : ABSENT									
	OPTION_OFFBOARD_DATA_SENDING : ABSENT									
	OPTION_PERSONAL_PICTURE_CLUSTER : ABSENT									
	OPTION_PUSHS_SECRETS : ABSENT									
	OPTION_REMOTE_MNGT_VHL : ABSENT									
	OPTION_SCR : ABSENT									
	OPTION_STT : ABSENT									
	OPTION_TLDIAG : ABSENT									
	PRESENCE_AMPLI_MUX : ABSENT									

Graphique 6 – Classeur Excel généré par l'outil

6 Audit de sécurité et de performances

J'ai eu l'occasion, sur la fin de mon stage, de réaliser un audit pour une société travaillant dans le monde de l'assurance. Ce genre de projets sort du cadre habituel de ceux entrepris par Alter Frame. C'est un contrat historiquement entretenu par Alter Frame depuis plusieurs années, et une opportunité intéressante de s'éloigner du développement et de faire une mission de conseil.

L'audit comportait trois axes importants : la qualité, la sécurité et la performance.

6.1 Méthodologie

Mon intervention a naturellement concerné ces trois aspects, et ce de trois façons différentes.

6.1.1 Sécurité

Il s'agit naturellement de la partie où mon intervention a été la plus notable (NOTE : est-ce que ça fait pas pompeux ?!!!). Les analyses de séurité se sont faites sur site, à partir d'une machine configurée par le client : l'intervention concerne une application web disponible uniquement en intranet chez le client et leurs procédures et protocoles de sécurité rendent très compliqué d'exporter l'appplication pour l'auditer à partir des locaux d'Alter Frame.

Le test a consisté à mener des analyses avec OWASP ZAP et à analyser et rejouer les résultats, trier les faux positifs des vraies failles de sécurité et faire un état de l'évolution de l'application depuis le précédent audit qui date d'un an.

Nous avons utilisé les fonctionnalités classiques de ZAP pour obtenir un résultat le plus exhaustif possible compte tenu du peu de temps alloué à l'audit de sécurité : d'abord *spider* l'application cible pour que le proxy en découvre la majorité, avant de lancer un scan actif qui, compte tenu de la taille de l'application auditée, a pris plusieurs heures de temps d'exécution.

Le test a également incorporé une partie audit de code, supportée par l'utilisation de SonarQube et qui se recoupera avec la partie qualité. Sonar peut, en effet, être configuré pour retourner des alertes de sécurité en ce qui concerne le code analysé. Comme pour ZAP il requière une intervention humaine pour écarter les faux positifs et n'offre aucune garantie d'exhaustivité, mais cela représentait un point d'entrée efficace pour chercher des traces de risques dans le code audité.

6.1.2 Performance

La partie audit de performance a elle aussi compris deux sous parties, l'une dynamique basée sur l'utilisation de JMeter [25] et de nmon [26], l'autre statique (à savoir, une analyse de code). Je ne suis personellement intervenu que sur la partie dynamique qui s'est effectuée sur site pour les mêmes raisons que l'analyse de sécurité.

JMeter est un outil développé par Apache qui permet de faire des tests de charge d'applications Web. Les fonctionnalités de l'outil sont impressionantes de diversité et de profondeur et comme encore une fois le temps alloué à l'auit était court, je n'ai pu qu'en effleurer la surface : mon travail a consisté à enregistrer une série de cas d'utilisation typiques en plaçant JMeter en proxy de navigation, puis à les configurer pour qu'ils soient reproductibles automatiquement (par exemple en générant des utilisateurs de manière procédurale) et à les rejouer en boucle.

JMeter produit des résultats très complets formatés en HTML qui affichent plusieurs métriques sous formes de graphiques ou de tableaux récapitulatifs, et nous avons effectué plusieurs jeux de test en augmentant à

25. `http://jmeter.apache.org/`
26. `http://nmon.sourceforge.net/pmwiki.php`

chaque fois la charge à laquelle le serveur était soumis pour observser son comportement jusq'au point où il "tombe".

Nmon pour sa part est un outil de monitoring

6.1.3

6.2 Résultats obtenus

Conclusion

Penser à mettre les crédits pour le template