colorlinks

Petit manuel de RoCaWeb

Team RoCaWeb

May 13, 2015

Contents

Introduction

0.1 Introduction

Dans ce petit guide, nous allons expliquer le fonctionnement du logiciel. Nous informons dès à présent le lecteur que RoCaWeb est toujours en constant développement, de ce fait, il fera face à des bugs que nous lui demandons de nous faire parvenir. Nous souhaitons aussi qu'après l'avoir testé, qu'il nous fasse parvenir les pros et les cons et des idées d'amélioration ainsi que des algorithmes à étudier.

RoCaWeb est un projet en cours financée par la DGA et conduit par Kereval et Télécom Bretagne.

Il a pour but la conception :

- d'un ou de plusieurs *reverse proxies* ;
- d'un module d'apprentissage ;
- et d'une interface graphique.

Le reverse proxy est mis entre le client et un serveur et vérifie que l'utilisation que le client fait des services hébergés sur le serveur (ou du serveur lui-même) répond aux **contrats** régissant les services. Á la suite de cette vérification, le *reverse proxy* autorise les requêtes ou lève des alertes. Dans le projet RoCaWeb, le fonctionnement à terme du *reverse* proxy devra assurer aussi bien un mode détection d'intrusions que la prévention. Dans ce dernier cas, il aura la charge de couper la connexion. L'interface graphique permet une gestion de l'apprentissage des contrats, leur manipulation et la gestion des profils. Nous allons maintenant donner quelques Définitions.

0.2 Définitions

Le **contrat** spécifie le fonctionnement normal d'un service. Il est difficile de définir de façon exhaustive le **fonctionnement normal** d'un site web. Cette difficulté s'accroit selon plusieurs facteurs dont par exemple le changement de frameworks applicatifs. Cependant, *l'apprentissage automatique* permet de déterminer, sur un ensemble de données représentative, un contrat. Les connaissances métiers servent aussi à compléter ce contrat.

Dans cette version de RoCaWeb, le contrat est défini sous forme :

- d'expressions régulières;

- de types prédéfinis.
- ou un mixage de ces deux types.

En partant de la capture d'un trafic, que nous supposons **sain**,Il faut noter que cette hypothèse joue un rôle important dans la suite. Car nos algorithmes ne traitent pas ce "bruit". Dans le cas où le trafic n'est pas sain, nous avons prévu des prétraitements pour éliminer les attaques connues, les doublons, etc. avant d'apprendre

nous avons conçus des algorithmes pour apprendre des expressions régulières et aussi assigner des types prédéfinis à ces données. Nous avons aussi implémenté la validation croisée et le *clustering* afin d'améliorer la phase d'apprentissage.

Ainsi, l'**apprentissage** est l'utilisation d'algorithmes de *machine learning* ou autres domaines pour déterminer des invariants permettant de définir un **comportement normal.** Le lecteur trouvera une description détaillé de ces algorithmes dans l'article publie dans la conférence C&ESAR 2014.

Le profil est l'ensemble des contrats exprimé sous les formes définis et appris pour un site web.

L'utilisateur est toute personne amenée à utiliser ce programme. Il peut s'agir d'un expert en sécurité ayant des compétences sur les algorithmes ou non. Ainsi, il aura plus ou moins besoin de se familiariser avec les algorithmes. Mais dans cette phase, il lui faudrait acquérir les notions sur l'alignement de séquences et la validation croisée.

0.3 Installation du logiciel

La version actuelle est distribuée sous forme d'une archive .tar.gz qu'il faudra décompresser. Á sa racine vous trouverez trois fichiers exécutables :

1. rocaweb.sh pour les systèmes UNIX
2. rocaweb.bat pour Microsoft Windows.
3. rocaweb.jar multiplateformes.

Les deux premiers fichiers sont des raccourcis pour ne pas avoir à taper :Dans le cas où vous vous trouver à la racine.

Vous pouvez aussi double cliquer sur rocaweb.jar pour lancer le programme.

Les vues

Au premier démarrage du programme, vous allez voir apparaître une fenêtre vous demandant d'accepter la licence. L'interface graphique que nous avons fournis se base sur celle d'OWASP ZAP. Nous avons gardés quelques similarités. Une fois accepté, le programme affiche la vue apprentissage par défaut ou la dernière vue sélectionnée avant la fermeture.

0.3.1 La vue apprentissage

Icône de la vue apprentissage.

C'est la vue principale de RoCaWeb. Elle est illustrée par la figure **??**.
Elle permet :

- la visualisation des données d'apprentissage;
- la configuration des prétraitements à appliquer sur ces données;
- le choix et le paramétrage des algorithmes d'apprentissage ;
- le choix du formatage des règles.

Nous allons détailler tous ces point plus en avant. Pour naviguer vers cette vue, il vous suffit de cliquer sur le bouton dans la barre des outils**??**.

0.3.2 La vue gestions des sites

Icône de la vue site.

Cette vue permet à terme de gérer la phase d'obtention des données d'apprentissage et des sites web. Cette vue est illustrée par la figure **??**.

Elle est donnée à titre illustrative et nous avons prévu d'adapter les *crawlers* de OWASP ZAP pour pouvoir :

- à partir d'une capture extraire les URL d'intérêt pour RoCaWeb
- crawler les sites afin :
  - d'obtenir leur squelette;
  - de compléter les données d'apprentissage en instrumentant le site.
- plusieurs autres traitements sont prévues à ce niveau.

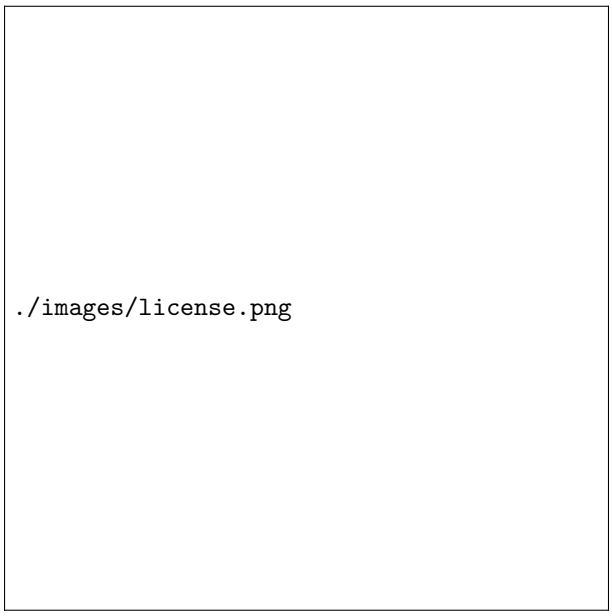

Figure 1: La licence de OWASP ZAP.

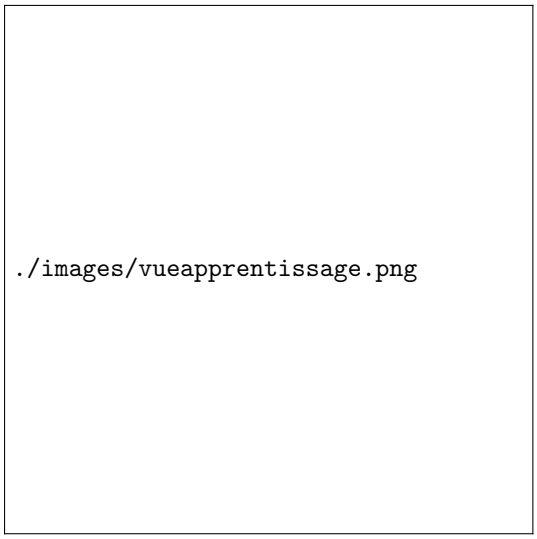

Figure 2: La vue apprentissage.

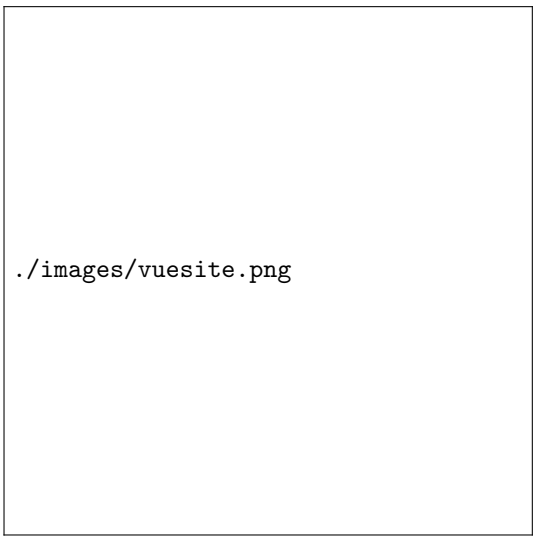

Figure 3: Vue permettant la gestion des sites Web notamment le crawling.

0.3.3 La vue profil

./images/expand_full.png Icône de la vue profil.

La vue profil permet la visualisation des profils et leur modification. Une fois l'apprentissage terminé, l'utilisateur peut décider ou non de sauvegarder le résultat. Lorsque le site est nouveau, un profil vierge est crée contenant quatre répertoires :

- regex : pour les expressions régulières (Principalement les résultats d'alignement suivi de génération d'expressions régulières) ;
- type : pour les résultats de l'algorithme de typage
- vc : pour la validation croisée
- proxies pour la configuration des reverses proxies qui surveillent ce site. Nous avons prévu à ce niveau un affichage en temps réel des performances du profil sur chaque proxy.

L'apprentissage et la génération des règles

0.4 Création d'une base d'apprentissage

Nous allons maintenant décrire en détails un processus d'apprentissage. L'utilisateur est supposé avoir un fichier PDML obtenu par ses soins...

RoCaWeb dispose d'un parseur PDML permettant l'extraction des données.Le Packet Details Markup Language est un dialecte XML qui permet de décrire des capture de trafic. Il est supporté par Wireshark. Pour lancer le parseur, l'utilisateur clique sur le bouton PDML (entouré de rouge sur l'image **??**).

Le programme tshark peut être utilisé pour enregistrer les requêtes au format PDML.

```
tshark -V -T pdml -i eth0  -R "http.request && ip.dst == [IPDESTINATION] && ip.src == [IPSOURCE] \
 && not http.request.uri matches \"(js|gif|png)\$\"" > fichier_output.pdml
```

Pour ce exemple, nous allons utiliser un site interne à la société Kereval qui s'appelle *tutos*. Nous allons fournir un fichier capturé durant la semaine 35 de l'année 2014. En cliquant sur le bouton PDML nous avons la fenêtre suivant**??** :

Ensuite, nous choisissons le fichier contenant la capture. Il est à noter que nous avons tester cette version sur des capture ne contenant qu'un seul site. A la fin du parsing, le nom "tutos" apparait dans l'arborescence du panneau "Data". Il est à noter que l'utilisateur peut fournir d'autres données sans passer par le parseur. Pour cela, il suffit de créer un dossier à la racine du répertoire "learninndata".

Nous attirons, votre attention sur le fait qu'il faudra respecter un certain convention pour que les règles soit format correctement.

- un répertoire par site à la racine de learninndata
- puis pour chaque site, l'arborescence doit correspondre à :

  ```
  $learningdata/site/subdirs/.../methodeHTTP/paraName
  ```

  Lorsque la méthode de formatage ne rencontre pas de méthode HTTP dans le chemin absolu, elle formate les règles en les faisant précédées du mot "FILE". Nous allons montrer ce cas un peu plus loin.

Voici un exemple de règle obtenu après apprentissage et formatée selon le format RoCaWeb:

```
SecRule "URL:'/tutos/php/skills/skill_eval_ins.php',GET:'SkillExperienceSelection_148716'" "0\+" "id:'0',sec:'wl',mode:'ID
```

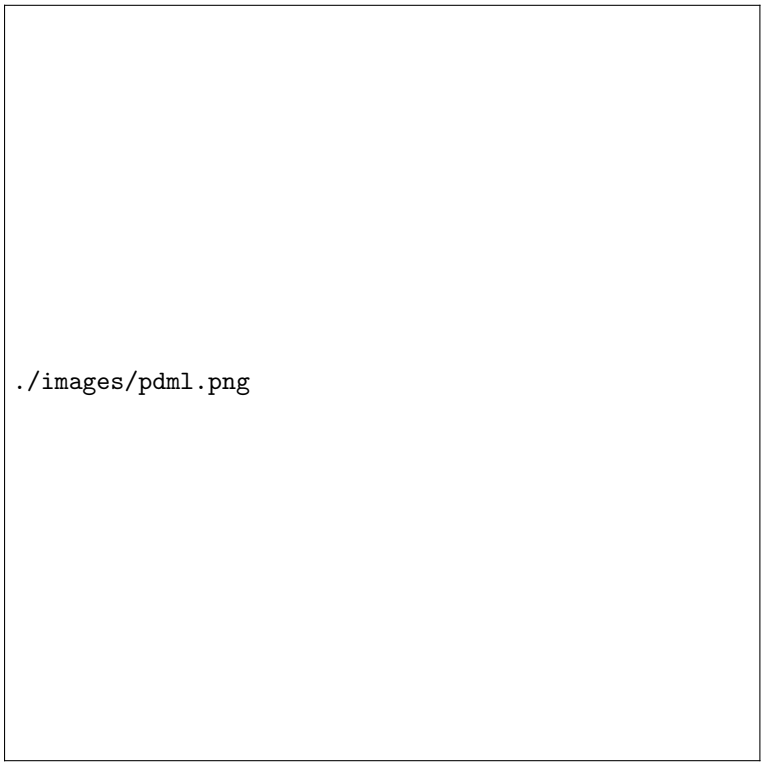

Figure 4: Cliquer sur le bouton PDML pour charger un fichier PDML.

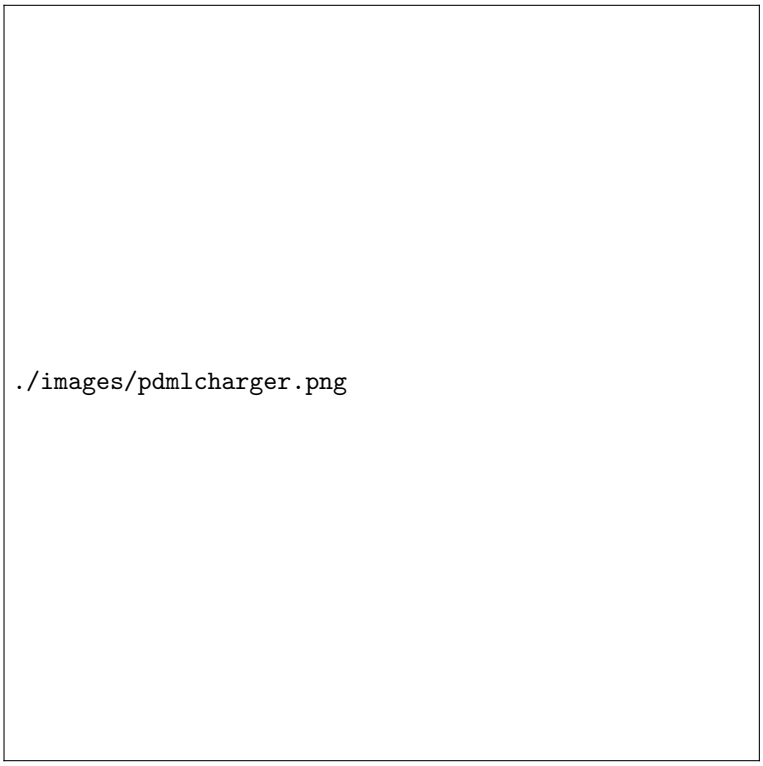

Figure 5: Fenêtre pour charger un PDML

0.5 Exemples de cas d'utilisation

Une fois les données fournies, l'apprentissage peut commencer. Il se déroule de la façon suivante :

1. choix des données d'apprentissage
2. choix des prétraitements
3. choix des algorithmes
4. choix du formatage des règles.

Ce qui est illustré par la figure **??**.

0.5.1 Premier cas d'utilisation

Par exemple, nous allons lancer un apprentissage sur tutos avec :

1. un nettoyage des doublons ;
2. par la méthode AMAA (Another Multiple Alignment Algorithm) ;
3. Avec comme sous méthode NeedlemanWunsch et un paramétrage par défaut de cet algorithme.
4. et le format est celui de RoCaWeb.
5. puis n'oublier pas de cliquer sur le bouton "Ajouter" pour que cette configuration d'apprentissage soit ajoutée à la liste des tâches à exécuter.

L'utilisateur peut répéter le processus pour différent répertoire, sous-répertoires et fichiers.

Lorsque l'utilisateur clique sur "Ajouter", les algoritmes choisis sont affichés dans le panneau "Running" en bas de l'écran et le bouton "Run" est activé. Il peut dès à présent lancer cette tâche ou ajouter d'autres algorithmes. Dans ce premier exemple, nous avons le résultat illustrée sur la figure **??**

0.5.2 Second cas d'utilisation

Dans ce second exemple, nous allons ajouter plusieurs algorithmes sur "tutos" et "training" et aussi le fichier "cat" dans training. Il est à noter que l'utilisateur peut visualiser le contenu des fichier d'apprentissage avant de les utiliser. Dans cette version, même s'il peut les éditer, les modifications ne sont pas prises en charge dans l'apprentissage.

C'est à vous maintenant de paramétrer les algorithmes et d'évaluer les résultats.

Une question que le lecteur se posera est de savoir quel est le meilleur paramétrage ? Á ce stade, nous continuons l'évaluation de nos algorithmes et nous ne pouvons pas lui fournir de valeurs. Les valeurs par défaut ne sont pas des valeurs trouvées de façon empiriques. Mais elles permettent de générer des règles. Cependant, AMAA et le typage génèrent des règles. Et lorsqu'ils sont combinées à la validation croisée permettent de créer un profil. Nous avons fourni une version de Littleproxy avec un parseur de nos règles.

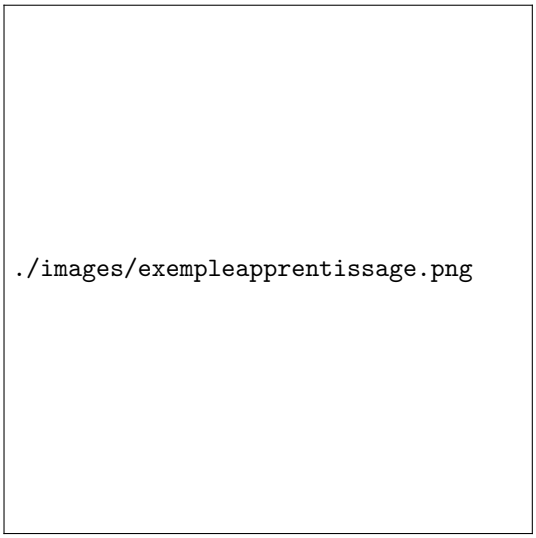

Figure 6: Le rouge indique les phases de l'apprentissage. Il signifie aussi que si vous oubliez une étape RoCaWeb ne sera pas content...

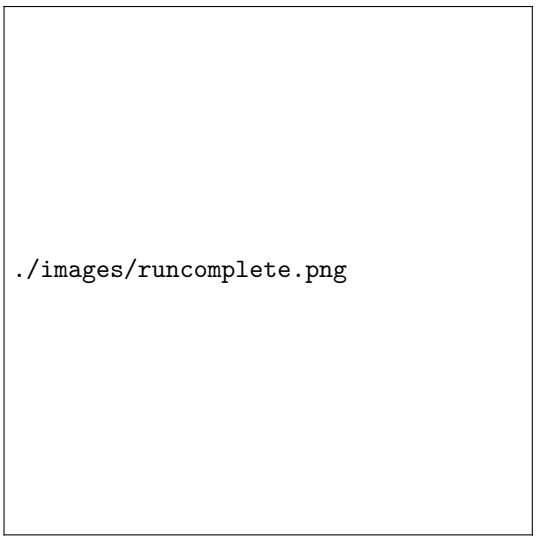

Figure 7: Illustration de la fin de l'apprentissage. Vous pouvez répéter le processus autant que vous le souhaitez.

Figure 8: Second exemple plus riches sur l'apprentissage.

Une autre remarque est le format des règles. Deux sont supportés en ce moment et les autres sont des illustrations des fonctionnalités à intégrer dans une version prochaine.

La création d'un profil

Une fois l'apprentissage terminé, l'utilisateur peut sauvegarder le résultat dans un profil, en cliquant sur le bouton "Enregistrer". Dans le cas où, un profil n'existe pas, le programme en crée un et ouvre le dossier pour la sauvegarde. Ceci est illustrer par la figure **??**.

Il est à noter que l'utilisateur peut aussi, éditer les règles et, cette fois-ci, les modifications seront pris en compte et sauvegarder dans le profil.

Vous pouvez aussi choisir un autre chemin.

Maintenant que vous avez créé vos profils, nous allons passer à la vue permettant de les manipuler.

0.5.3 La visualisation et la gestion des profiles

Cette vue permet de :

- visualiser les profils
- de créer de nouveaux;
- ou d'importer des fichiers définissant des règles.

Pour visualiser un profil, l'utilisateur clique sur l'arborescence afin de choisir le type de contrat. Celui est affiche comme sur la figure **??**. Cette vue peut être personnalisée par un menu déroulant permettant de choisir plusieurs cas dont :

- une syntaxe pour mettre certaine partie de la règle en surbrillance.
- un retour à la ligne

Le menu déroulant apparait après un clic droit dans le panneau qui affiche les règle. Ce menu peut aussi servir en (en/de)coder les règles, à faire des recherches, etc.

Le bouton "Visualiser" n'est pas activé pour le moment. A terme il permet une visualisation sous forme de graphe des règles.

Comme nous l'avons indiqué, l'utilisateur peut créer un profil vierge et aussi importer des fichier déjà existant. Pour ce faire, il clique dans la vue profil sur les boutons indiqués sur la figure **??**.

0.6 Le reverse proxy

Le proxy Rocaweb peut être utilisé pour valider le trafic à partir de règles au format natif ou Modsecurity. L'exécutable du proxy est fourni à la racine

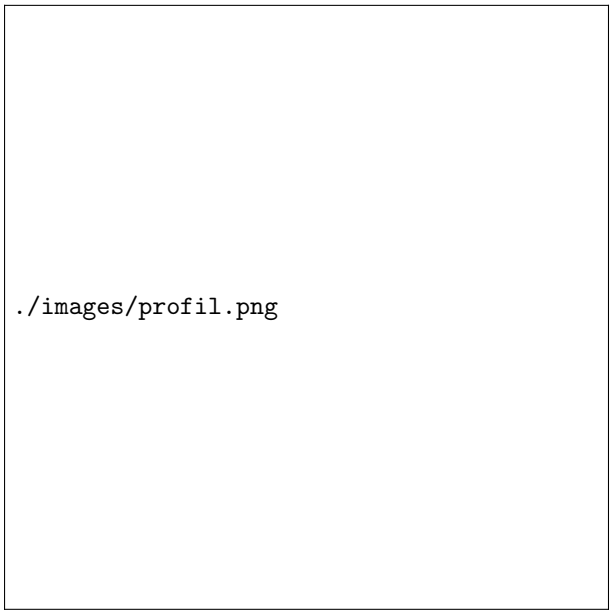

Figure 9: Création d'un profil vierge pour tutos.

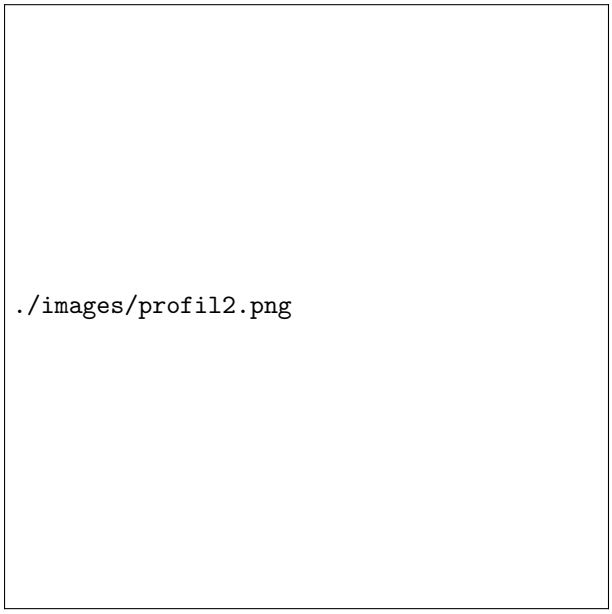

Figure 10: Sauvegarde du profil dans regex ou type ou vc.

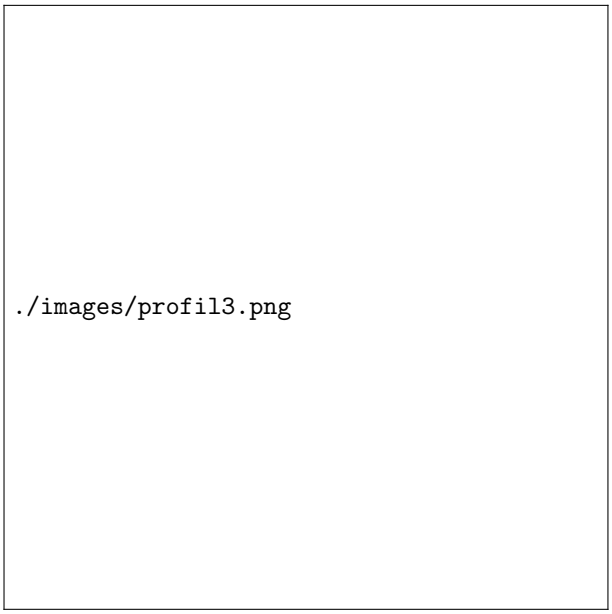

Figure 11: La visualisation des profils.

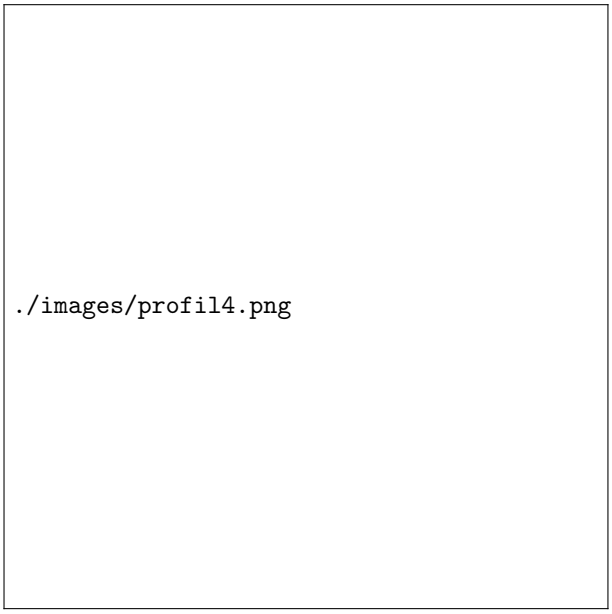

Figure 12: Profil est affiché.

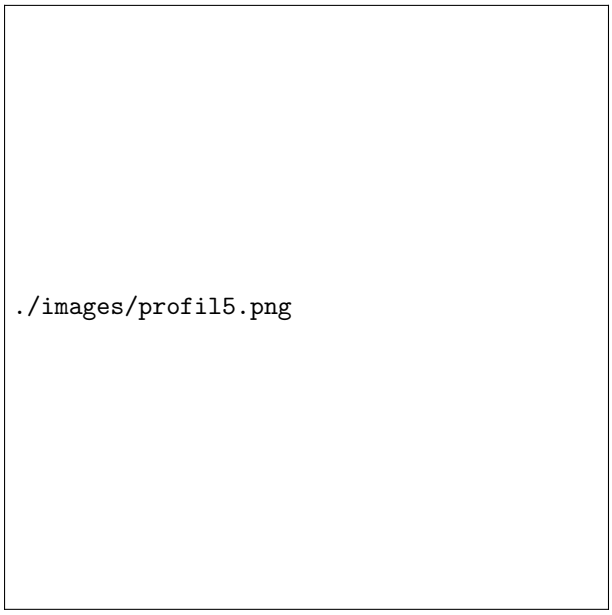

Figure 13: Coloration syntaxique de la vue et possibilité d'encoder et de retourner à la ligne.

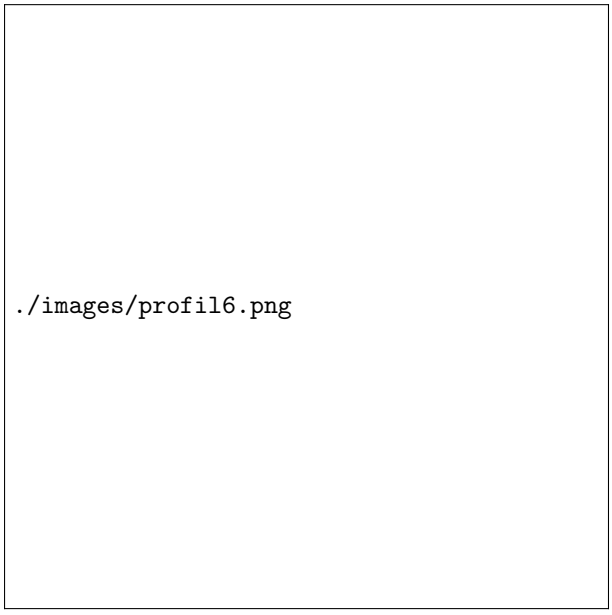

Figure 14: Création et importation de profils.

du répertoire rocaweb : rocaweb_proxy_V2.zip à décompresser dans un autre répertoire.

Il est fourni sous forme de ficher jar et nécessite Java 7 pour fonctionner.

```
--dumping              Dumpe le trafic en PDML modifie
                       (n cessite l'option --validation pour etre active)
--folder <arg>         Lit les fichiers contenus dans le dossier
                       precise et parse les regles au format Rocaweb
--help                 Affiche l'aide
--https                Le proxy se connecte en https au serveur cible et en http aux clients
--msfolder <arg>       Lit les fichiers contenus dans le dossier pr cise
                       et parse les regles au format Modsecurity.
                       Seules les r gles correspondants    des expressions
                       regulieres sont comprises.
--port <arg>           Specifie le port d' coute  du proxy (par d faut 8080)
--reverse <arg>        Site cible
--reverseport <arg>    Port de connexion au site cible
--validation           Active la validation des requetes
```

Cette partie clôt la description des fonctionnalités principale que nous avons implémenté dans RoCaWeb. Cependant, d'autres fonctionnalités sont présentes mais pas totalement aboutis. Notamment la configuration avancée des algorithmes. Nous y travaillons.

Conclusions

Nous allons maintenant aborder les cas où, le programme que vous aller utiliser rencontre des problèmes :

- le traitements des PDML peut lever des Exceptions
- lorsque vous ne choisissez par un fichier pour l'apprentissage ou un algorithmes
- des restrictions liés au systèmes d'exploitation. Vous n'avez pas besoin de droit **root** pour utiliser ce programme, mais nous avons mis en œuvre deux processus légers qui surveillent les dossiers "/resource/learningdata" et "profiles" pour qu'en cas de modifications les JTree associés soient automatiquement mis à jour. Or dans certain cas des exceptions ont été levées.
- l'adaptation automatique des vues à la taille des écrans est en cours d'amélioration. En effet, les panneaux et sous-panneaux sont redimensionnée mais pas toujours de façon agréable.
- nous n'avons pas encore intégré une validation des entrées utilisateur. Ce qui veut dire que vous êtes libres de tester des injections de toutes sortes dans les champs recevant les paramètres. Normalement, vous devez voir des exceptions. Normalement...

Nous concluant sur le fait que la recherche et le développement sont en cours et que ces problèmes et d'autres seront solutionnés dans les versions futures. Nous allons mettre en place des outils de mises à jours automatique et de réception des bugs et des souhaits.