LogAnalyzer

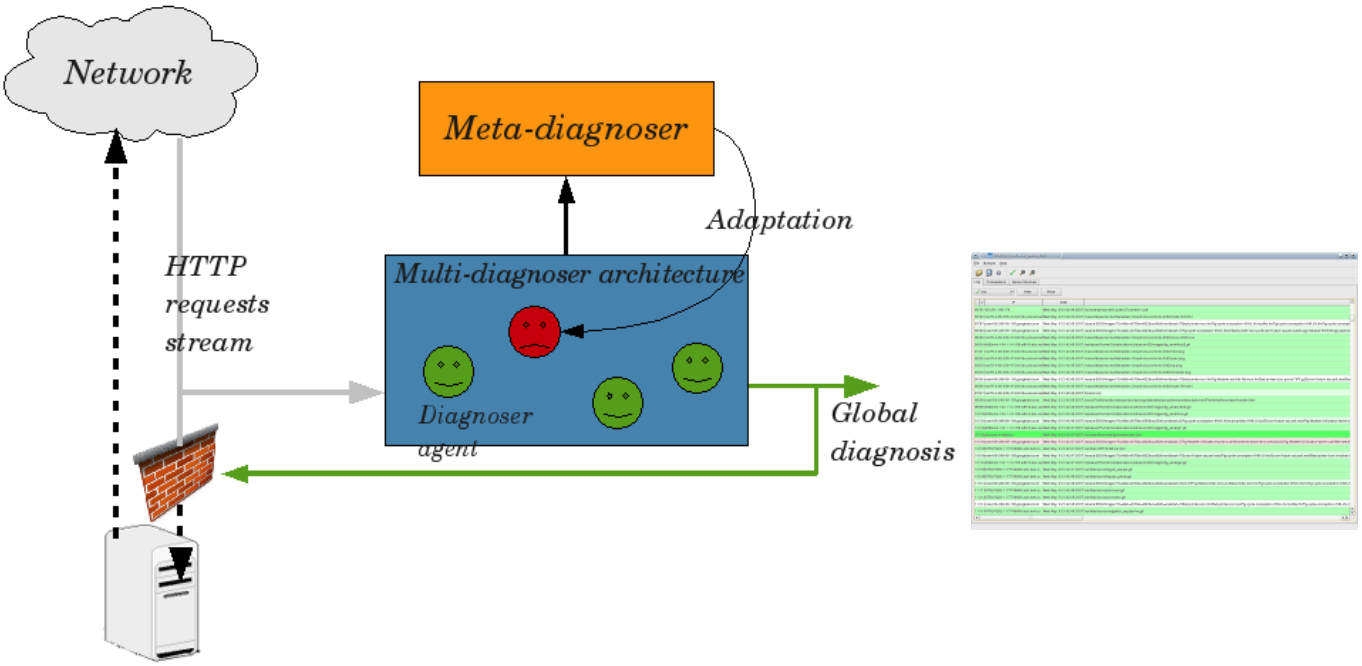
Thomas Guyet[1,2], René Quiniou[2] et Marie-Odile Cordier[3]

[1] AGROCAMPUS-OUEST
[2] INRIA/IRISA – Centre de Rennes (Équipe DREAM)
[3] Université de Rennes/IRISA (Équipe DREAM)

Contact : thomas.guyet@irisa.fr
Plus d'informations : http://www.irisa.fr/dream/LogAnalyzer/

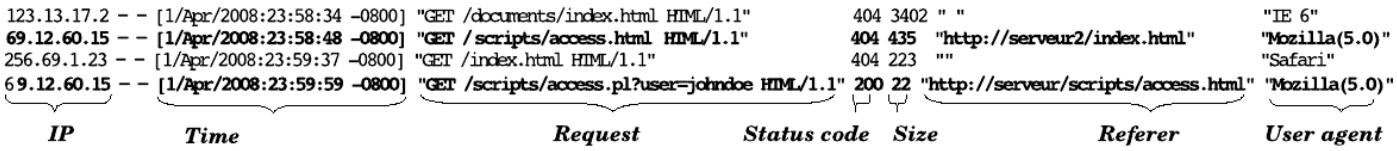
Diagnostic adaptatif d'un flux d'alarmes par méta-diagnostic distribué

Application à la détection d'intrusions dans un serveur Web



1 Détection d'intrusions à partir des logs Apache

1. Flux d'informations structurées obtenus à partir du **log Apache** :



2. Intrusions web

Intrusion web : exploitation d'une faille d'un serveur ou d'un outils accessibles par celui-ci, et déclenchable par des **requêtes adéquates**, pour un usage frauduleux.

Exemples :

- `/awstats/awstats.pl?configdir=echo;echo%20YYY;cd%20%2ftmp%3bwget...;echo%20YYY;echo`
- `/rpc/..%%35%63..%%35%63/winnt/system32/cmd.exe?/c+dir+c:\\+/OG`
- `/cgi-bin/mrtg.cgi?cfg=//../../../winnt/win.ini`

La **protection d'un serveur web** doit permettre de bloquer l'exécution de ces requêtes tout en permettant aux autres utilisateurs de continuer leur activités. Il est donc nécessaire de savoir les détecter de manière précise.

3. Méthodes de détection d'intrusion

Une méthode de détection d'intrusion consiste à utiliser comparer une requête avec un **modèle d'intrusion** ou un **modèle de requête normale**.

Les modèles s'appuient sur des **caractéristiques** des champs d'une requête qui sont **discriminantes pour certains types d'intrusions** :

- distribution de caractères de la requête,
- distribution de *tokens* de la requête,
- longueur de la requête,
- chaînes de markov ou réseaux bayésiens entre tokens de la requêtes,
- classes d'adresses IP (blacklist, whitelist),
- ...

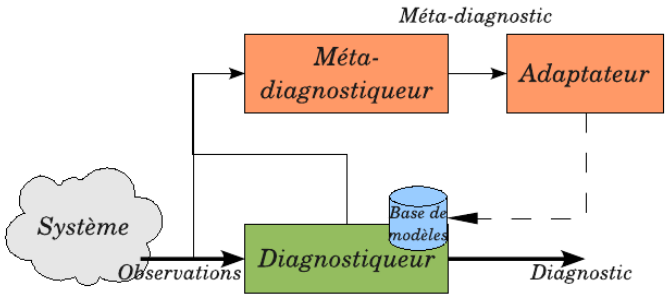
En pratique, les informations accessibles par un serveurs web et les intrusions **évoluent** dans le temps. Un bon **système de détection doit pourvoir s'adapter** (en des interventions humaines) pour détecter au mieux les intrusions.

2 Diagnostic adaptatif à l'aide d'un ensemble de diagnostiqueurs et un meta-diagnostiqueur

A Diagnostic adaptatif

Le but d'un **système adaptatif de diagnostic** est de surveiller et diagnostiquer un système tout en s'adaptant à son évolution. Ceci passe par l'adaptation des diagnostiqueurs qui précisent ou enrichissent leur propre modèle pour suivre au mieux le système au fil du temps. Le monitoring adaptatif par un méta-diagnotiqueur soulève trois problèmes majeurs :

1. **détection de changements** : la détection d'un besoin d'adaptation du modèle,
2. **diagnostic des diagnostiqueurs** : l'identification de la partie du modèle à adapter (,
3. la détermination (non-supervisée) du diagnostic qu'il faudrait obtenir en présence de la double indétermination sur l'état et le mode du système.

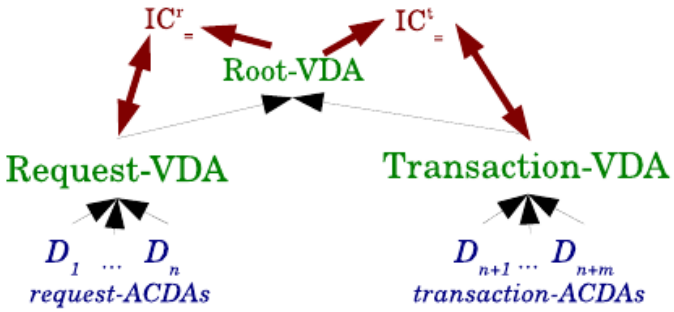


B Approche par un ensemble de diagnostiqueurs et méta-diagnostiqueur

- Le diagnostic est construit par un **ensemble de diagnostiqueurs**
  - Chaque diagnostiqueur utilise une caractéristique des données,
  - Le diagnostic global est obtenu par **fusion de diagnostic**
- Des connaissances *a priori* sur des relations attendues entre les résultats diagnostiqueurs (redondance d'information) permettent d'expliciter des **contraintes d'intégrité** de l'ensemble des diagnostiqueurs.
- Le **méta-diagnostiqueur** vérifie la satisfaction des contraintes d'intégrités
  - Sinon, il provoque une adaptation des diagnostiqueurs impliqués

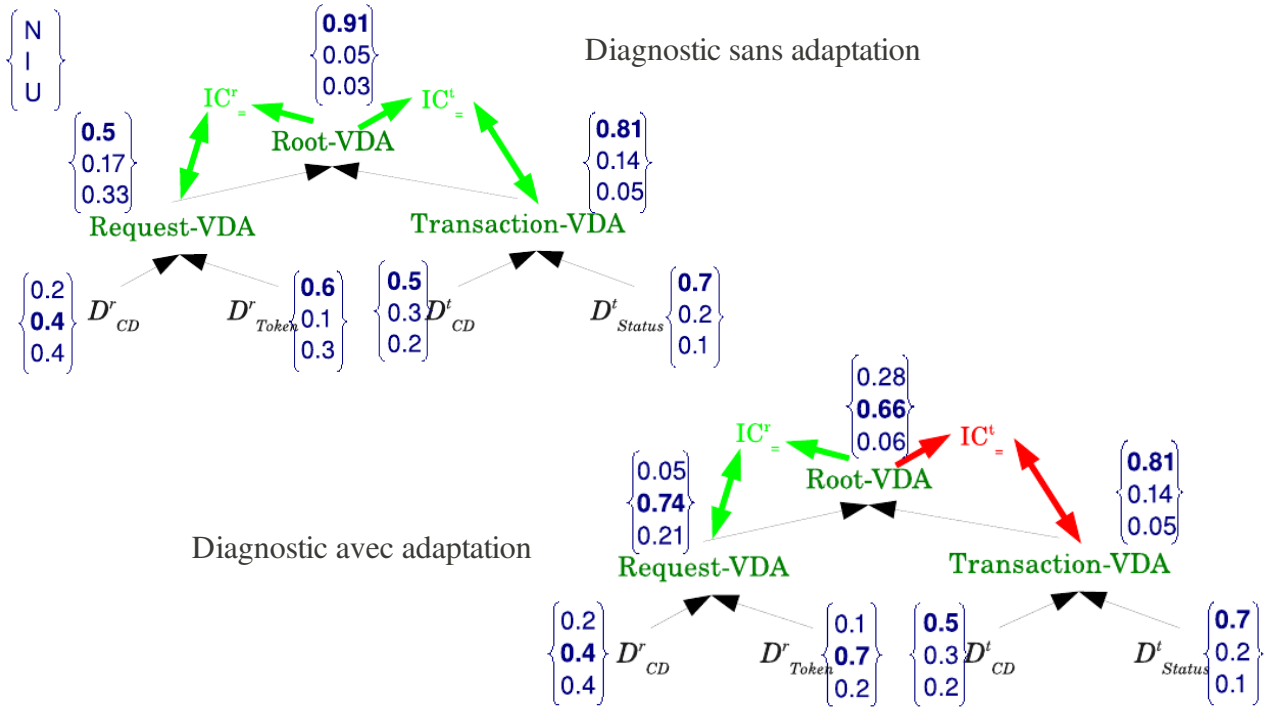
C Exemple d'ensemble de diagnostiqueurs pour la détection adaptative d'intrusion

1. Les diagnostiqueurs
   - Diagnostiqueurs utilisant des informations sur les requêtes



   - Diagnostiqueurs utilisant des informations sur les transactions (ensemble de requêtes effectuées par un même IP)
   - Contraintes d'intégrités : les diagnostiqueurs de requêtes et les diagnostiqueurs de transactions doivent donner le même diagnostic

2. Exemple



3 Les logiciels

LogAnalyzer et ***LogAnnotator*** sont des logiciels de visualisation, d'exploration et d'analyse de flux de requêtes à des serveurs HTTP (journaux d'accès à un serveur Apache). L'objectif de ces logiciels a été l'**évaluation d'une méthode adaptative de détection d'intrusions à partir d'un flux de données structurées**. *LogAnnotator* permet d'annoter manuellement des logs Apache afin de disposer de logs annotés. *LogAnalyzer* est une implémentation de la méthodes adaptative de détection d'intrusion à partir d'un flux de requêtes.

Les logiciels présentés ont été développé en C/C++ en utilisant les librairies *Qt4* et *Qwt* pour la conception des interfaces graphiques. Ils ont été développés dans le cadre du projet SéSur (ARC INRIA). Ce sont des logiciels expérimentaux, et à ce titre, non-dénués de bugs latents ...

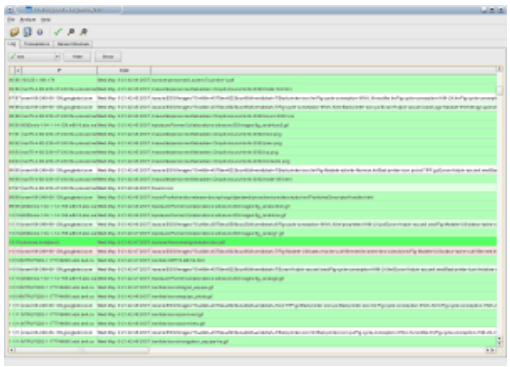
A Principe de conception modulaire

Conception modulaire :

- Une API implémentant l'approche de diagnostic adaptatif par un ensemble de diagnosers et un meta-diagnosers
- Une API de fonctionnalités sur les logs Apache : parsing statique ou en flux, méthodes de détection d'intrusions, ...
- Une instanciation de l'API de diagnostic au cas de la détection adaptative d'intrusion
- Des IHM

L'API de diagnostic est réutilisable pour d'autres applications. Pour l'utiliser dans sa propre application, il suffit de développer ses classes de diagnostiqueurs (hériter des classes de l'API) et ses classes correspondant à des contraintes d'intégrités.

B LogAnnotator



1. Présentation

L'outil LogAnnotator est un outil interactif pour l'aide à l'annotation d'intrusions dans des logs Apache.

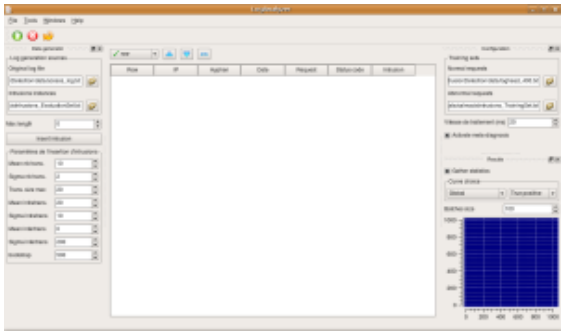
2. Fonctionnalités

- Parsing des logs apaches

(format Combined),

- Filtrage *a priori* de requêtes : permettre de réduire efficacement le nombre de requêtes à annoter
  - Utilisation d'expressions régulières pour définir des types de requêtes,
  - Exemples de filtrages : les robots, les images, les pages statiques, ...
- Trois vues sur les données pour faciliter la détection manuelle des intrusions
  - Vue séquentielle du log
  - Vue par transactions pour identifier des utilisateurs
  - Vue par documents pour identifier des documents potentiellement sensibles
- Quelques fonctionnalités de recherche dans le log
- Annotations des requêtes : ajout manuel d'annotations, enregistrement et chargement d'annotations
- Anonymisation efficace de logs Apache

C *LogAnalyzer*



1. Présentation

LogAnalyzer est un logiciel pour le traitement d'un flux de données. Il permet de tester la méthode adaptative de diagnostic.

2. Fonctionnalités

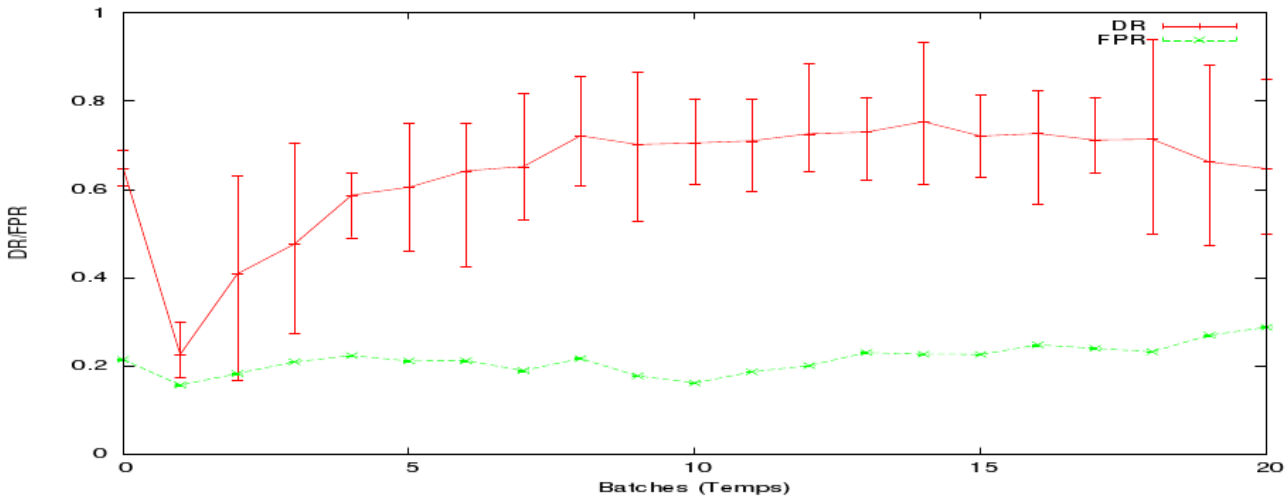
- Génération d'un flux de requêtes avec insertions aléatoires d'intrusions,
- Calcul de statistiques par lots de données (*batches*) : FP, TP, rappel, précision, accuracy, ...
- Détection d'intrusion et détection de changements (nouveaux types d'intrusions),
- Fonctionnement en batch pour traiter de gros jeux de données (1M de requêtes)

4 *Quelques résultats*

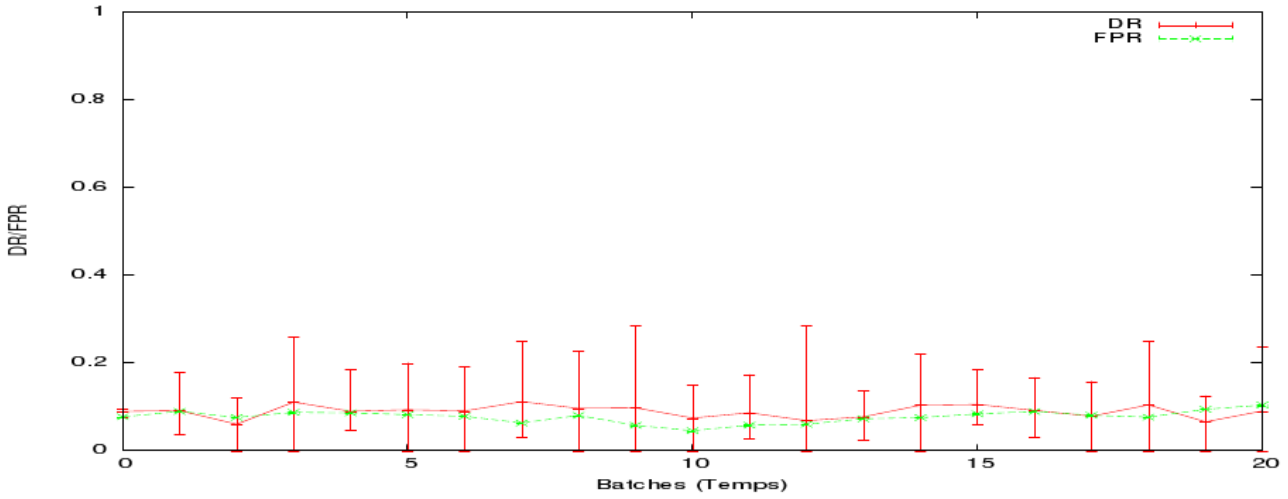
Évolution du taux de détection (DR, en rouge) et du taux de faux positif (FPR, en vert) dans le temps. La courbe est une moyenne des taux de détection et des taux de faux positifs obtenu pour plusieurs insertions aléatoires d'intrusions sur un même jeu de données.

Chaque point est calculé pour un lot de données comportant 1000 requêtes.

Avec adaptation



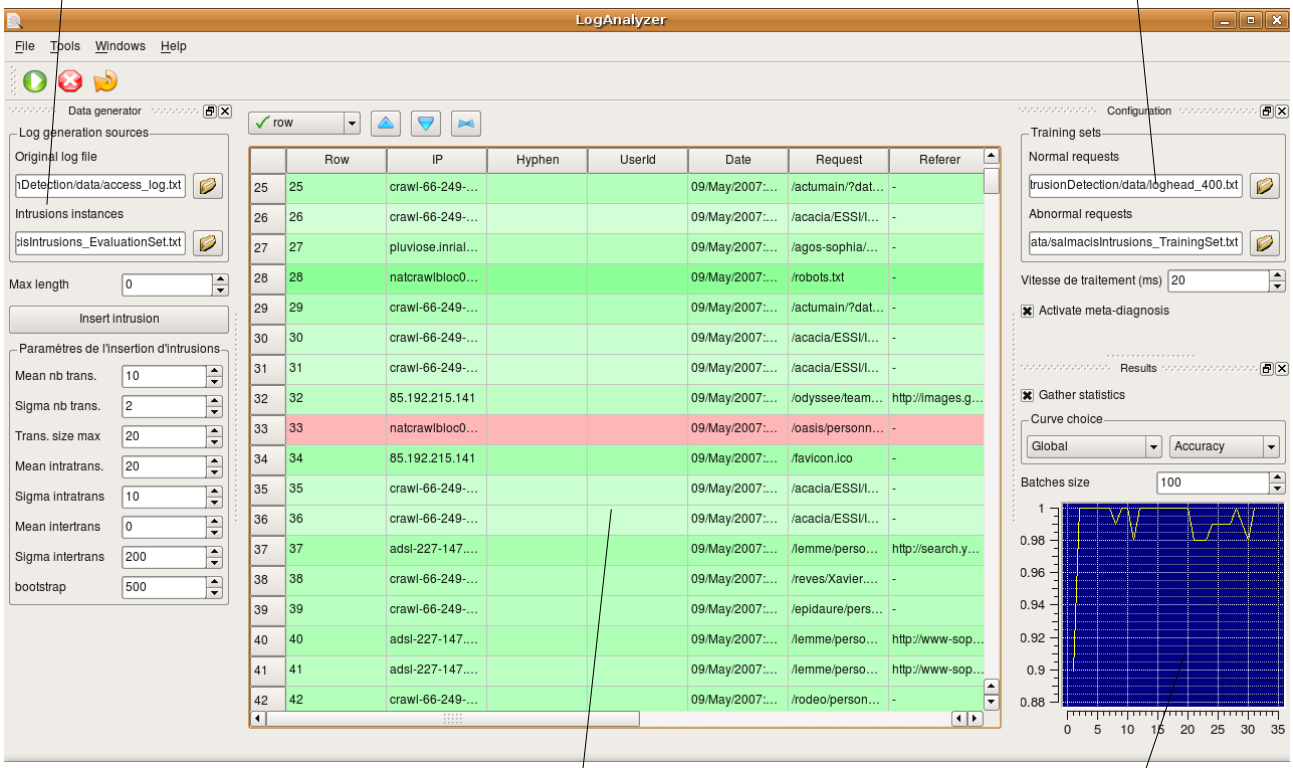
Sans adaptation :



5 Interface

Options de la simulation des intrusions dans le log

Option de l'ensemble de diagnostiqueurs.



Visualisation du log en cours d'analyse.

Vert = ok Rouge = intrusion

Plus la saturation d'une couleur est importante, plus le diagnostiqueur est sûr de son diagnostic.

Affichage dynamique des statistiques de détection

6 Publications

1. Thomas Guyet, Wei Wang, René Quiniou, Marie-Odile Cordier: Diagnostic multi-sources adaptatif. Application à la détection d'intrusion dans des serveurs Web. EGC 2009: 325-336.
2. Wei Wang, Thomas Guyet, Rene Quiniou, Marie-Odile Cordier, Florent Masseglia: Online and adaptive anomaly Detection: detecting intrusions in unlabelled audit data streams. EGC 2009: 457-458
3. Thomas Guyet, René Quiniou, Wei Wang and Marie-Odile Cordier: Self-adaptive web intrusion detection system, INRIA Research Report June 2009