République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université ABBAS LAGHROUR Khenchela

Faculté des Sciences et Technologies

Département de Mathématique et Informatique

Mémoire de fin d'étude

Pour l'obtention du diplôme de Master en Informatique

Spécialité : Sécurité et technologies web

Thème

Détection d'intrusion dans les environnements informatiques d'apprentissage humain à partir du comportement de l'utilisateur

---

Encadré par

- Dr. Tarek DJOUAD

Réalisé par

- Khiari Walid

- Kadri Mohamed Essadek

Année Universitaire 2017/2018

Résumé

Les environnements informatiques d'apprentissage humain « **EIAH** » sont aujourd'hui utilisés par la plupart des établissements d'enseignement dans le monde, et la vie privée des utilisateurs de ces environnement est exposée à des attaques externes par des utilisateurs non autorisés. Nous présentons dans ce mémoire une nouvelle approche pour détecter les intrusions à l'aide d'une ingénierie dirigée par des modèles, guidée par les traces d'activité.

Nous proposons d'utiliser un système de détection d'intrusion « **IDS** » basé sur les traces pour identifier les changements dans le comportement des utilisateurs et nous proposons une étude de cas réelle dans Moodle pour illustrer notre approche.

Mots clés : EIAH, IDS, trace, SGBT.

Abstract

Technology Enhanced Learning (TEL) systems are used today by most of educational institutions in the world, and users' life privacy is exposed to external attacks by unauthorized users. We present in this work a new approach to detect intrusion using model driven engineering led by activity traces. We propose to use a Trace Based System to identify changes in learners' behavior and we propose a real case study in Moodle to illustrate our approach.

Key words: *Information Security, Intrusion detection, TEL systems, Activity Trace, Trace Based Systems*

ملخص

أنظمة التدريب الذكية هي برمجيات حاسوبية تهدف إلى تمكين الأفراد من التعلم بطرق فعالة ذات مغزى و هذا عن طريق تقنيات حاسوبية متنوعة، و قد أصبحت نماذج كثيرة من هذه النظم التدريبية مستخدمة في المؤسسات التعليمية عبر العالم. تتعرض خصوصية حياة المستخدمين لهاته الأنظمة لهجمات خارجية بواسطة مستخدمين غير مصرح لهم. نقدم في هذا البحث منهجًا جديدًا لاكتشاف التسلل إلى هاته الأنظمة باستخدام هندسة مبنية على نماذج تقودها الآثار الإلكترونية للمستخدمين. نقترح استخدام نظام قائم على التتبع لتحديد التغييرات في سلوك المتعلمين ونقترح دراسة حالة حقيقية في موودل لتوضيح النظام.

الكلمات المفتاحية : أنظمة التدريب الذكية، ، نظام كشف التسلل، الآثار الإلكترونية، النظم القائمة على الآثار.

Table des matières

Liste des figures

Liste des abréviations

EAO	Enseignement Assisté par Ordinateur
EIAO	Enseignements Intelligemment assistés par Ordinateur
EIAO	Environnements Interactifs d'Apprentissage avec Ordinateur
TICE	Technologie d'Information et de Communication pour l'Enseignement
EIAH	Environnement informatique d'apprentissage humain
IGPDE	Institut de la Gestion et du Développement Economique
SCORM	Sharable Content Object Reference Model
Moodle	Modular Object-Oriented Dynamic Learning Environment
SGC	Système de gestion de contenu
PHP	Hypertext Preprocessor
MSSQL	Microsoft SQL
SGBT	Système de gestion à base de trace
Obsels	Observed elements
SGBDR	Système de Gestion de Bases de Données Relationnelles
M-trace	Modeled trace
SGBD	Système de Gestion de Bases de Données
XML	Extensible Markup Language
KTBS	Kernel for Trace-Based Systems
LIRIS	Laboratoire d'Informatique en Image et Systèmes d'information
RDF	Resource Description Framework
JSON	JavaScript Object Notation
Turtle	Terse RDF Triple Language
IDS	Intrusion detection system
NIDS	Network Based Intrusion Detection System
HIDS	HostBased Intrusion Detection System
CGI	Common Gateway Interface
SMB	Server Message Block
EMERALD	Event Monitoring Enabling Responses to Anomalous Live Disturbances
HyIDS	Hybrid IDS
FTP	File Transfer Protocol
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
RFC	Requests for comments

Remerciements

En premier lieu, nous remercions Dieu le très haut qui nous a donné la patience, le courage et la volonté durant ces années d'étude

Nous tenons à saisir cette occasion et adresser nos profonds remerciements et nos profondes reconnaissances à toutes personnes qui nous ont aidés de près ou de loin dans la réalisation de ce mémoire.

Nous remercions ***Dr. Tarek Djouad****, en tant que encadreur de projet, pour ses précieux conseils et son orientation tout au long de notre recherche.*

Enfin nous exprimons notre profonde reconnaissance à tous responsables et enseignants de l'équipe de la spécialité sécurité et technologies web qui ont contribués à notre formation, et pour la richesse et la qualité de leurs enseignements.

Nous tenons aussi à remercier les membres du jury qui ont accepté d'examiner notre mémoire.

''Your work is going to fill a large part of your life, and the only way to be truly satisfied is to do what you believe is great work. And the only way to do great work is to love what you do. If you haven't found it yet, keep looking. Don't settle. As with all matters of the heart, you'll know when you find it. ''

Steve Jobs

Dédicaces

Je dédie humblement ce manuscrit:

A celle qui s'est toujours dévouée et sacrifiée pour moi;
Celle qui m'a aidée du mieux qu'elle pouvait pour réussir;
Celle qui a toujours été là dans mes moments de détresse, ***ma très chère mère.***

A celui qui m'a toujours encouragée et soutenue moralement,
Mon très cher père.

A *mes très chers frères et sœurs*
Que dieu vous protège et vous procure santé et bonheur...

A tous ceux ou celles qui me sont chers et que j'ai omis involontairement de citer.
A tous Mes enseignants tout au long de mes études.
A tous mes amis/amies et mes proches.
A tous ceux qui ont participé de près ou de loin à la réalisation de ce travail.

Kadri Mohamed Essadek.

Introduction générale

Dans nos jours on connaît un véritable engouement en matière d'enseignement à distance, aujourd'hui le terme e-Learning est désormais employé pour désigner un dispositif de formation utilisant l'internet ou l'intranet comme vecteurs de diffusion de connaissances et de formations.

Le terme environnement informatique pour l'apprentissage humain (**EIAH**) est apparu au début des années 2000 pour caractériser un environnement informatique conçu dans le but de favoriser l'apprentissage humain, ces environnements apportent de multiples avantages liés à ses modes de production, de présentation et de diffusion de contenu. La plupart des plateformes d'apprentissage humain permettent aux apprenants et aux enseignants de travailler ensemble en utilisant un ensemble d'outils et de ressources. Ces outils comme les chats, les messages privés, les forums, les wikis, etc., et les ressources comme les cours, les exercices, les outils de simulation, les notes d'évaluation, etc. sont le cœur des environnements d'apprentissage et doivent être sécurisés.

La plupart des EIAH's assurent des services de confidentialité en utilisant un processus d'authentification / d'identification.

Malheureusement, les utilisateurs non autorisés peuvent toujours accéder aux informations privées relatives aux utilisateurs des différents EIAH's et ensuite à leurs activités connexes, face à l'accès du grand nombre d'utilisateurs avec leurs différentes intentions qui peuvent être parfois destructifs, les EIAH's sont exposés au plusieurs menaces cela rend les données ainsi que les ressources des utilisateurs et des établissements vulnérables.

La sécurité des EIAH's ainsi que toutes systèmes informatique est devenue un enjeu stratégique et pour assurer cette sécurité, différents outils ont été utilisés, tels que les pare-feu et les anti-virus mais la plupart du temps ses outils sont inefficaces face à des nouvelles menaces sophistiquées pour cela la naissance des systèmes des détection d'intrusion a beaucoup aider dans la protection de toutes système informatique ainsi que les EIAH's contre ses tentatives malicieux.

Le système de détection des intrusions (**IDS**) est l'une de techniques qui offre un contrôle permanent et permettant ainsi de détecter toute tentative de violation de la politique de sécurité, c'est-à-dire toute intrusion.

Problématique :

L'objectif de ce travail est de proposer un outil pour la détection des intrusions dans les EIAH's à partir du comportement de l'utilisateur, nous présentons dans cet mémoire notre approche pour créer un IDS basés sur des traces d'activité afin de détecter d'éventuelles intrusions. Nous proposons d'utiliser une ingénierie dirigée par les traces d'activité des utilisateurs pour détecter les intrusions possibles en utilisant l'analyse des traces d'activité et les modèles de comportement de l'utilisateur.

Afin de bien comprendre, nous présentons notre travail dans quatre chapitres, le premier chapitre est consacré aux environnements informatiques d'apprentissage humain, définition, caractéristiques, objectifs et architecture et on va voir Moodle l'un des outils les plus utilisés pour développés un EIAH.

Dans le deuxième chapitre on va voir la définition d'un système de gestion à base de trace, c'est quoi une trace d'activité et ces composants et on va présenter KTBS l'outil qui sert a la gestion des systèmes à base de trace, quant au troisième chapitre on va voir les systèmes de détection d'intrusion leur définition et types, les modes de détection d'intrusion, et on va finir par la conception de notre outil et son implémentation.

1 *Chapitre 1*

Les environnements informatiques d'apprentissage humain

Introduction

Aujourd'hui, l'enseignement à distance est de plus en plus répandu aussi bien dans les institutions publiques que privées. Ces systèmes représentent une méthode d'apprentissage utilisée par tous et un moyen de faciliter et d'encourager la conception de cursus pédagogiques par les éducateurs.

Ces systèmes ont d'abord été appelés "**Enseignement Programmé**" puis "**Enseignement Assisté par Ordinateur**" (**EAO**), ils sont devenus par la suite, les "**Enseignements Intelligemment assistés par Ordinateur**" (**EIAO**) avec l'introduction des techniques d'Intelligence Artificielle, puis elles sont évoluées pour aboutir aux "**Environnements Interactifs d'Apprentissage avec Ordinateur**"

Vers la fin des années 90 et avec l'intégration des **TICE** (Technologie d'Information et de Communication pour l'Enseignement) donne la naissance des **Environnements Informatique d'Apprentissage Humain** (**EIAH**), dont la notion de partenariat est entre l'homme et sa machine.

1.1 Définition d'un EIAH

Tchounikine définit les Environnements Informatiques pour l'Apprentissage Humain comme « un environnement informatique conçu dans le but de favoriser l'apprentissage humain, c'est-à-dire la construction de connaissances chez un apprenant. Ce type d'environnement mobilise des agents humains (élève, enseignant, tuteur) et artificiels (agents informatiques, qui peuvent eux aussi tenir différents rôles) et leur offre des situations d'interaction, localement ou à travers les réseaux informatiques, ainsi que des conditions d'accès à des ressources formatives (humaines et/ou médiatisées), ici encore locales ou distribuées ». **[1][2]**

Ainsi l'apprentissage est amélioré par la technologie, c'est comme une culture où un large éventail d'étudiant dispose d'un environnement technologique robuste qui offre des possibilités d'apprentissage efficaces, partout où l'étudiant choisit d'apprendre.

1.2 Quelques caractéristiques des EIAH

Parmi les caractéristiques générales des EIAH's **[3]**, les caractéristiques les plus connus sont:

- ***des systèmes informatiques*** : en considérant un EIAH comme un système informatique, nous pouvons lui attribuer les caractéristiques spécifiques à un logiciel.
- ***un domaine de recherche pluridisciplinaire*** : un EIAH fait appel, directement ou indirectement, à plusieurs disciplines comme la pédagogie, la didactique, la psychologie cognitive, les sciences de l'éducation et informatique, les sciences de l'information et de la communication, etc.

- *un domaine complexe :* La complexité des EIAH porte sur deux volets.

* Le premier volet concerne la complexité de leur mise en place (conception – réalisation – expérimentation – évaluation – diffusion). **[4]**

* Le deuxième volet concerne la complexité d'utilisation. Les EIAH sont en effet, des environnements informatiques complexes, rarement «intuitifs» malgré les efforts de leurs concepteurs. **[5]**

1.3 Objectif des EIAH

Les EIAH constituent l'un des moyens les plus efficaces pour l'amélioration de l'apprentissage pour tous les secteurs de la connaissance pour lesquels on vise un transfert de savoir et de savoir-faire, parmi ces objectifs on trouve **[3]**:

- ***Un EIAH permet d'individualiser l'enseignement*** : l'EIAH est venu pour pallier aux problèmes de l'enseignement traditionnel. Il s'agit notamment du volet lié à l'adaptabilité et à l'individualisation des contenus pédagogiques aux besoins des différents apprenants.
- ***Un EIAH permet le suivi et la gestion des parcours*** : des dispositifs logiciels de suivi et de gestion des apprenants et de leurs parcours d'apprentissage peuvent être intégrés dans un EIAH **[6]**.
- ***Un EIAH permet de faire des économies d'échelle sur les ressources*** : les recherches effectuées sur le domaine des EIAH, ont permis de mettre en place des standards et normes permettant la réalisation de ressources pédagogiques autonomes, interopérables et réutilisables tout en prenant en compte la diversité des systèmes utilisés. Ceci a permis d'optimiser les ressources pédagogiques réalisées en réduisant les coûts de production et de maintenance **[7].**
- ***Un EIAH permet l'exploitation des traces réalisées par ses apprenants*** : l'exploitation permet non seulement de produire des éléments intéressants pour la modélisation comportementale ou conceptuelle, mais aussi une personnalisation pertinente des EIAH par production de feedbacks ou évolution des interfaces. Différents procédés liant compréhension, formalisation et action sont pris en compte **[8]**.

1.4 Architecture des EIAH's

Les premières architectures des EIAH's proposées comportaient quatre composantes principales: un modèle de domaine, un modèle d'étudiant, un modèle pédagogique, et une interface ou bien un modèle de communication **[9]**. Depuis, cette architecture de base a été étendue par de nombreux chercheurs.

Les EIAH's contiennent généralement un modèle de domaine, un modèle d'étudiant, un modèle pédagogique, et une interface. Ces modules échangent des informations entre eux à travers une série de liens.

- **Le modèle de domaine** expert du domaine relatif aux compétences expertes concernant le domaine enseigné.
- **Le modèle de l'étudiant** module de gestion du profil de l'apprenant relatif à la prise en compte de caractéristiques personnelles (capacités d'apprentissage, profil psychologique, profil des erreurs, historique de son parcours dans le système EIAO…)
- **Le modèle pédagogique** ou expert pédagogue qui a pour objectif de faciliter l'acquisition de connaissances par l'apprenant.
- **Le modèle de l'interface** relatif aux interactions entre le système et l'apprenant.

1.5 Différents types d'EIAH [2]

La période récente a cependant vu l'explosion de systèmes fondés sur les capacités offertes par Internet et les technologies de l'information et de la communication. Plusieurs système EIAH ont vu le jour les plus connus parmi eux sont :

- ***Micromondes*** environnements de simulation ou encore environnements de réalité virtuelle, représentent un autre type d'EIAH, fondé sur l'idée de l'immersion de l'apprenant dans un monde virtuel.
- ***Tuteurs intelligents*** : les tuteurs intelligents sont pour Anderson des outils pour tester une théorie, une façon de mettre en œuvre une théorie psychologique de l'apprentissage et de la tester empiriquement. Si le tuteur intelligent fonctionne, il permet aux apprenants d'apprendre ce qui avait été prévu, il constitue une validation de la théorie. Si le tuteur intelligent ne produit pas l'apprentissage prévu, cela constitue une invalidation de la théorie.
- ***Hypermédias pour l'apprentissage*** : et plus largement les documents électroniques, sont des outils de présentation de l'information. Ils sont utilisés par les enseignants ou les concepteurs pour représenter des connaissances au moyen de mots, d'images, de vidéos, d'équations, etc., par le biais de modalités visuelles ou auditives

1.6 Acteurs des EIAH

On trouve trois acteurs classés selon leurs rôles dans un EIAH qui sont : les apprenants, les enseignants et les administrateurs**. [1][2]**

1- **L'apprenant :** C'est la personne qui utilise le dispositif de formation pour acquérir des connaissances, il peut être un étudiant, un employé d'entreprise, ou une personne désirant perfectionner ses connaissances dans une branche quelconque.

2- **L'enseignant :** il y a plusieurs types d'enseignants, différenciés par leurs rôles. On peut distinguer quatre types d'enseignants.

- C**oncepteur de cours :** celui qui développe un cours en utilisant les outils de la plateforme selon ses objectifs pédagogiques.
- **Tuteur :** son rôle est de superviser le déroulement du cours.
- **Orienteur :** c'est l'enseignant qui a pour principales tâches, l'élaboration des cursus des apprenants ou des groupes d'apprenants.

- **Evaluateur :** son rôle est de créer les tests, de suivre les apprenants et de gérer les tests d'évaluation.

3- **L'administrateur :** gère la plateforme (installation et maintenance).

1.7 Les plates-formes

L'IGPDE **[10]** (Institut de la Gestion et du Développement Economique) définit une plate-forme d'apprentissage, ou plate-forme de téléformation, comme un « système informatique conçu pour optimiser, sur un réseau Internet ou Intranet, la gestion de l'ensemble des activités de formation, depuis l'information sur l'offre, l'inscription des participants, la distribution des ressources, l'organisation de parcours individualisés, l'animation de communautés d'apprentissage »

Il existe environ plus de 200 plates-formes d'apprentissage en ligne dont une trentaine sous licences libres. La norme **SCORM** (Sharable Content Object Reference Model) **[11]** permet de transposer du contenu d'une plate-forme à une autre plate-forme de formation en ligne. A titre d'illustration de plateforme on pourrait mentionner quelques unes telles : **Moodle, WebCT, Claroline, Ganesha, Dokeos, Spiral, Acolad, Virtual U, Top Class, etc.**

La plate-forme la plus utilisée dans les Universités et les grandes écoles est **Moodle**

1.7.1 La Plate-forme Moodle

Le terme Moodle est l'acronyme de **Modular Object-Oriented Dynamic Learning Environment,** c'est une plateforme d'apprentissage en ligne servant à créer des communautés d'apprenants autour de contenus et d'activités pédagogiques (Fig.1). À un système de gestion de contenu (SGC), Moodle ajoute des fonctions pédagogiques ou communicatives pour créer un environnement d'apprentissage en ligne : c'est une application permettant de créer, par l'intermédiaire du réseau, des interactions entre des pédagogues, des apprenants et des ressources pédagogiques**. [12]**

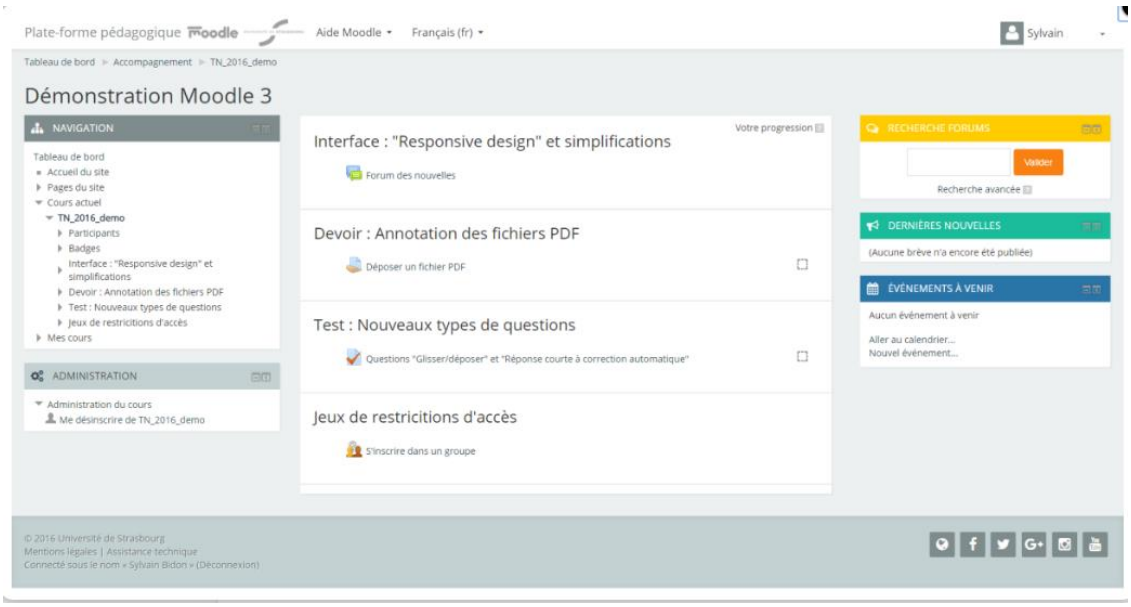

Figure.1. Plate-forme Moodle

Les rôles (*roles*) pédagogiques par défaut de Moodle sont, par droits décroissants **[12]:**

- **Administrateur** : tous les pouvoirs sur la plate-forme
- **Responsable de cours** (*course creator*) : enseignant pouvant créer ses propres cours
- **Enseignant** (*teacher*) : ses cours sont créés par l'administrateur, il ne peut modifier que ceux qui lui ont été attribués.
- **Enseignant non éditeur** (*non-editing teacher*) : ne peut pas modifier le contenu du cours
- **Étudiant** (*student*)
- **Utilisateur authentifié** (*authenticated user*) : rôle par défaut pour tous les utilisateurs autres qu'invité
- **Invité** (*guest*)

Les membres d'un cours ont accès aux activités suivantes si l'enseignant les a sélectionnées :

- **Chat**: "chat" ou salon de discussion (possibilité de l'ouvrir un certain jour, à une heure précise, de manière hebdomadaire, etc.).
- **Forum**: différents types de forums (sujets imposés par l'enseignant, sujets proposés par les étudiants, évaluation ou commentaire possibles, etc.).
- **Devoir**: remise de travaux avec évaluation de l'enseignant (de différents types : texte en ligne, dépôt de fichier, dépôt avancé de fichiers, activité hors ligne).
- **Test**: suite de QCU, QCM, de questions vrai/faux, questions numériques, appariements, textes à trous, etc.
- **Leçon**: document comprenant des questions et des aiguillages vers des parcours différents en fonction des réponses (évaluation possible).
- **Atelier**: remise de travaux avec évaluation par les étudiants.
- **Glossaire**: production collective d'un document organisé alphabétiquement (commentaire, validation et évaluation possibles).
- **Wiki**: production collective d'un document hypertexte (commentaires possibles de l'enseignant).
- **Base de données**: création d'enregistrements comportant des champs personnalisés, et recherche par critères dans la base.
- **Sondage**: question posée comportant une série d'options au choix.
- **Dialogues**: messagerie interne entre membres du cours.
- **Groupes / groupements**: les membres d'un cours peuvent être séparés en groupes (et avoir accès à des parties réservées de forum, par exemple) ou des groupements de groupe (qui restreignent complètement l'accès à des ressources/activités).

ENVIRONNEMENT DU LOGICIEL

- **Serveur** : Moodle tourne sans modification sur Unix, Linux, Windows, Mac OS X, et autres systèmes qui supportent un serveur web,
- **PHP**
- **Système de gestion de base de données** (MySQL, Postgres, MSSQL, Oracle ...).

Conclusion

Les EIAH constituent l'un des moyens les plus efficaces pour individualiser l'apprentissage pour tous les domaines de la connaissance.

Le présent chapitre a essayé de cerner quelques définitions liées au domaine des EIAH. Une étude détaillée des EIAH requiert une étude exhaustive des différentes disciplines en liaison avec ce domaine ce qui dépasse de loin le cadre de ce travail. A la fin de ce chapitre, nous avons fait une petite présentation de la plate-forme d'apprentissage en ligne Moodle qui va être notre champ d'étude par la suite.

2 Chapitre 2

SGBT (Système de gestion à base de trace)

Introduction

Les Environnements Informatiques pour l'Apprentissage Humain (EIAH) offrent souvent la possibilité de tracer l'activité de ces utilisateurs aux cours des différentes sessions d'apprentissage. Ces traces peuvent être utiles à tous les administrateurs des EIAH's, dans notre cas on va les utilisés pour détecter les intrusions à partir du comportement de l'utilisateur (apprenants et enseignants), dans ce chapitre on va d'abord donner une petite définition de trace puis expliquer c'est quoi un système de gestion à base de trace et ses composantes, on verra aussi l'outil qu'on va utiliser pour notre SGBT.

2.1 Définition d'une trace

De son définition dans le langage courant une trace signifie « l'empreinte ou la suite d'empreintes laissées par le passage d'un homme ou d'un animal » prenant exemple les traces physiques peuvent être visible comme les traces des pieds dans le sable ou invisible comme les empreint digitales, de la même façon dans le monde numérique mais à une échelle plus grand (toutes action laisse une trace), ya aussi des traces visibles comme les commentaires, les likes sur facebook, les twitts ... etc., et il ya aussi des traces invisible comme l'historique de recherches, ou des sites visités, ...etc.

Ainsi une trace numérique est « un ensemble de données numériques laissées de manière active et passive par une personne à la suite de ses actions dans le système » **[13]**.

Dans notre contexte, une trace numérique est issue de l'observation d'une activité, elle représente une signature d'un processus interactionnel **[14].**

Plus précisément une trace vise à représenter une activité comme un ensemble d'obsels (élément observé). Chaque obsel a, au moins, un type et un intervalle de temps (début et fin). Il peut également avoir un nombre arbitraire d'attributs et de relations avec d'autres obsels **[15].**

Une trace est également liée à un modèle de trace, qui peut être stocké dans la même SGBT (système de gestion à base de trace). Enfin, les traces peuvent être stockées, calculées ou réutilisées à tout moment.

2.2 Les obsels

Les obsels (abréviation de « **Observed elements** » en anglais) sont les éléments atomiques des traces. Un obsel est décrit par les éléments suivants **[15]**:

- un type d'obsel,
- un intervalle de temps : temps de début et temps de fin (qui peut être égal au temps de début),
- un sujet optionnel (l'utilisateur étant tracé).

Il peut aussi avoir plusieurs attributs, et être lié à d'autres obsels de la même trace par des relations binaires. Le type obsel, les attributs et les relations sont décrits par un modèle de trace.

2.3 Le modèle de trace

Le modèle de trace décrit les types obsel que la trace peut contenir, leurs attributs et leurs relations. Un modèle de trace permet de tracer à peu près ce qu'est un schéma sur une trace dans un SGBDR, sauf qu'un modèle de trace a une identité propre et peut être partagé par plusieurs traces. Un modèle de trace définit les éléments suivants **[15] [16]:**

- une hiérarchie de types obsel,
- attributs que les obsels de chaque type peuvent avoir,
- les relations qui peuvent exister entre les obsels de chaque type.

Les types obsel sont organisés dans une hiérarchie de spécialisation: chaque obsel d'un sous-type appartient aussi au supertype. En conséquence, les attributs et les relations sont hérités d'un supertype par ses sous-types.

2.4 Une M-trace

Une m-trace (pour modeled trace / trace modélisée) est une trace associée à un modèle qui en fournit un guide de construction et de manipulation. Un modèle de trace doit définir **[17]** :

- La manière de représenter le temps,
- Les types d'obsels permettant de décrire l'activité,
- Pour chaque type d'obsel, les types d'attributs possibles,
- Les types de *relations* binaires que peuvent entretenir les obsels entre eux.

2.5 Le système de gestion à base de trace

Un Système de Gestion de Bases de Traces (*SGBT*) est un outil informatique destiné à collecter, transformer, stocker et à visualiser des traces modélisées dans une base de trace.

En effet, un SGBT joue le même rôle qu'un SGBD (Système de Gestion de Bases de données) dans les applications standard, mais gère plutôt des traces modélisées (m-traces).

Le SGBT comme un système informatique permettant et facilitant l'exploitation des traces. Il est composé de différents composants comme le montre la figure (fig.2) **[18]**.

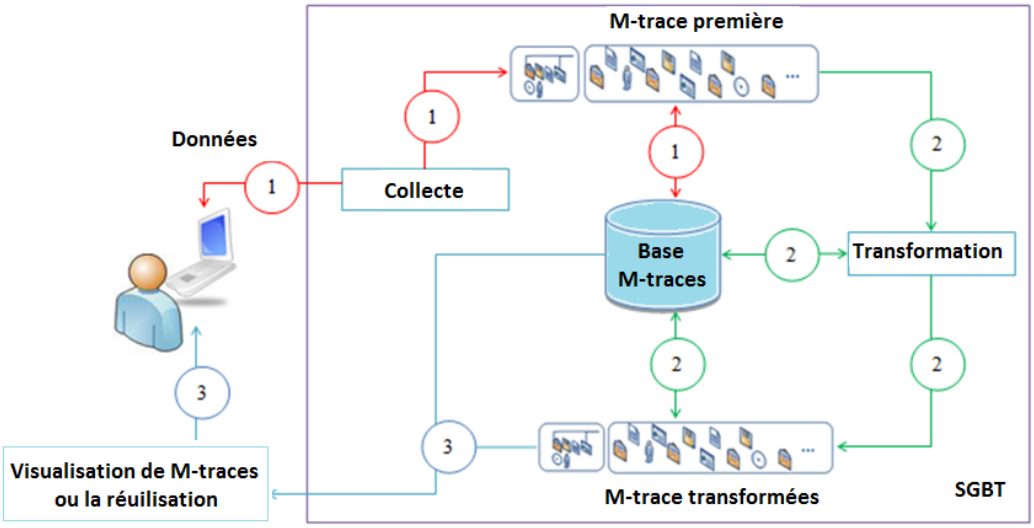

Figure.2. Composantes d'un SGBT

2.6 Les composantes d'un SGBT [19] [8]

2.6.1 Système de Collecte

La collecte permet de mettre en place l'observation de l'utilisation d'un système à partir de sources de traçage.

Elle consiste à transformer de façon automatique ou semi-automatique, pendant l'activité ou a posteriori, des informations générées par l'interaction utilisateur/système en une m-trace première du SGBT (Fig.3). On appelle source de traçage tout flux d'information structuré à partir duquel il est possible de mettre en place un processus de collecte de traces pour un SGBT (XML, texte, vidéo, audio) **[8].**

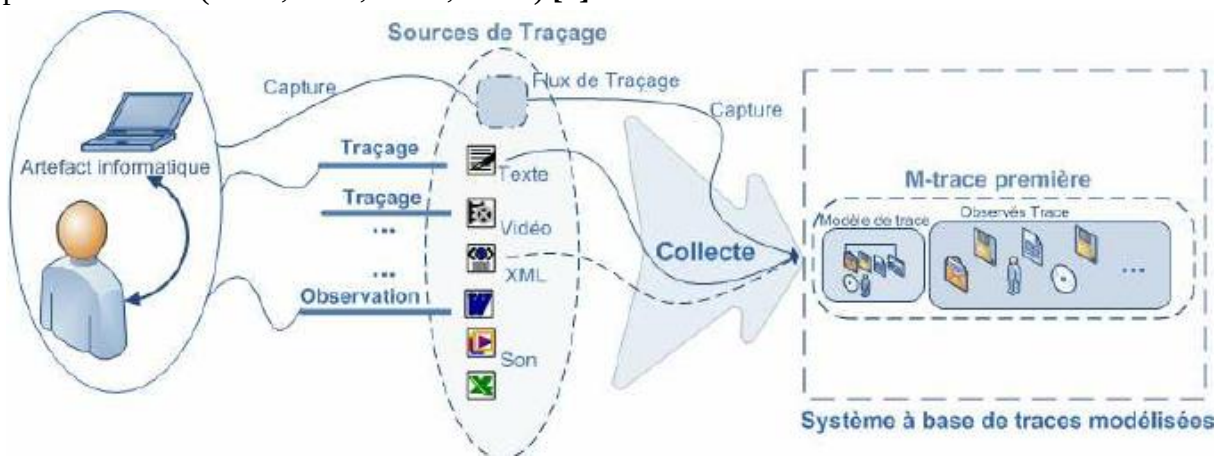

Figure.3. Le système de collecte[8]

2.6.2 Système de Transformation

La trace obtenue à l'issue de la collecte est qualifiée de m-trace première. Celle-ci n'est cependant pas toujours exploitable directement, et il faut parfois passer par une ou plusieurs transformations pour atteindre une trace d'un niveau d'abstraction cohérent avec l'activité.

On appelle transformation de m-traces tout processus qui transforme une ou plusieurs m-trace gérée(s) par un système à base de m-traces en une autre m-trace gérée par le même système. Les m-traces premières d'une base de m-traces d'un SGBT sont les seules m-traces non transformées de ce SGBT **[9]**.

2.6.3 Système de Visualisation

L'une des plus importantes tâches au sein d'un SGBT est la visualisation des traces, en effet, la visualisation des traces permet de faciliter l'analyse et l'interprétation des traces.

Le système de visualisation donne la possibilité à l'utilisateur d'avoir une vue ergonomique des traces du SGBT, et permettre aussi d'accéder aux sources de traçage et aux données relatives aux traces **[8] [9].**

2.7 KTBS (Kernel for Trace-Based Systems)

Depuis plusieurs années, l'équipe **Tweak** (Traces, Web, Education, Adaptation, Knowledge, précédemment appellé **Silex**) **[20],** propose des outils et modèles pour la construction de systèmes à base de traces : des systèmes exploitant la connaissance présente dans les traces d'interaction des utilisateurs. Ces systèmes doivent être capables d'assister les utilisateurs, et d'apprendre comme eux de leurs interactions, résultant dans un couple utilisateur-système co-évoluant.

KTBS (Kernel for Trace-Based Systems en français noyau pour les systèmes à base de trace) est un système informatique mettant en pratique la notion de SGBT, développé au sein du laboratoire **LIRIS. KTBS** (fig.4) propose plusieurs fonctionnalités de gestion des traces, à savoir la création, l'interrogation, la transformation et la visualisation.

KTBS utilise le formalisme RDF pour décrire les traces et les modèles, et expose ses données dans différents formats (XML, JSON, Turtle) **[15].**

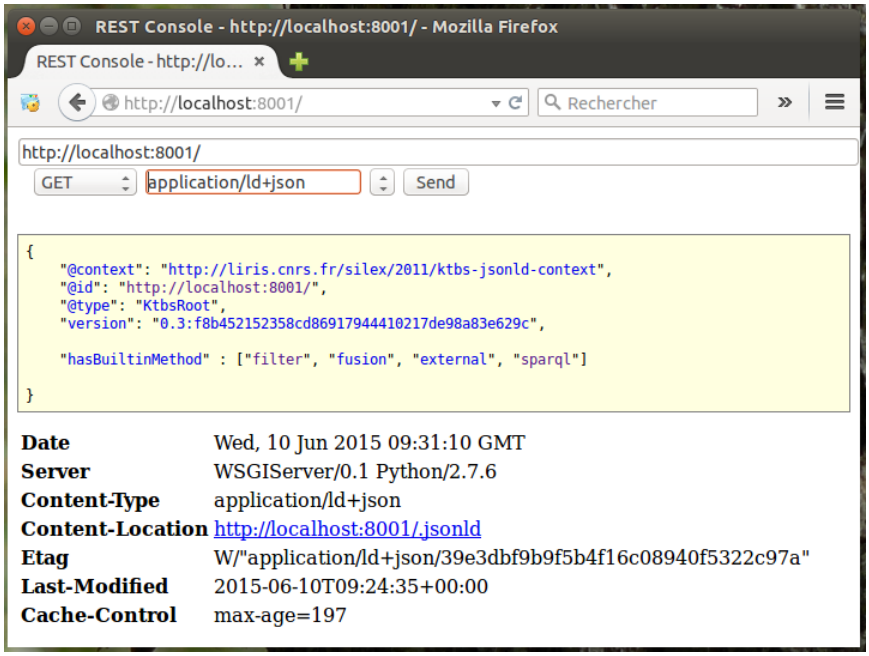

Figure.4. KTBS, REST Console

Turtle

Turtle **[21]** (Terse RDF Triple Language) est un format pour exprimer des données dans le modèle de données Resource Description Framework (RDF) avec une syntaxe similaire à SPARQL. RDF, à son tour, représente des informations en utilisant des "triplets", dont chacun consiste en un sujet, un prédicat et un objet. Chacun de ces éléments est exprimé en URI.

Turtle fournit un moyen de regrouper trois URI pour en faire un triple, et fournit des moyens d'abréger ces informations, par exemple en factorisant des portions communes d'URI.

Json

JavaScript Object Notation ou JSON **[22]** est un format de données textuelles dérivé de la notation des objets du langage JavaScript. Il permet de représenter de l'information structurée comme le permet XML par exemple. C'est un format de données très commun utilisé pour la communication asynchrone navigateur-serveur, y compris en remplacement de XML dans certains systèmes de style AJAX.

Conclusion

La plupart des plates-formes d'apprentissage produisent des traces d'activité. Ces traces sont enregistrées dans des fichiers journaux, des bases de données, des fichiers XML, des fichiers audio et vidéo, … etc. Ces traces d'activité présentent des actions réalisées par les utilisateurs de ces plates-formes (apprenants et enseignants) au cours de leur formation. Les SGBT sont des systèmes susceptibles d'intéresser beaucoup d'acteurs intervenant sur des systèmes tracés, étant données les fonctionnalités qu'ils proposent, cependant les traces colléctées et stockées dans ces systèmes sont très utiles.

3 Chapitre 3

IDS (Système de détection d'intrusion)

Introduction

L'utilisation fréquente des environnements informatiques d'apprentissage humain attire de plus en plus de utilisateur qui essayent de s'y intégrer de façon illégale afin d'y accéder et de modifier des données. L'administrateur dans plusieurs cas ne sera pas en mesure de contrer les différentes attaques et intrusion vu que l'EIAH n'est jamais assez protégé. C'est pour pallier ce manque que sont apparus de nouvelles solutions de sécurité appelées systèmes de détection des intrusions(IDS).

3.1 Définition d'un IDS

Un système de détection d'intrusion (ou IDS: Intrusion Detection System) est un appareil ou un logiciel qui collecte des données par surveiller les différentes activités du réseau ou du système, puis analyser les données recueillies pour déterminent si des opérations malveillantes se produisent **[23].**

Les IDS utilisent des mécanismes visant à identifier tout type de trafic potentiellement malveillant créé par des utilisations non autorisées telles que tentatives d'intrusion, attaques virales, flux excessif, trafic suspect, etc. Les IDS détectent une activité suspecte et lancent une alerte. Les activités d'une cible pouvant être un réseau ou une machine hôte. Il existe trois types d'IDS: NIDS pour surveiller les réseaux, HIDS pour surveiller les systèmes, et IDS hybride qu'utilise les deux types NIDS et HIDS.

Il y a aussi des modes de classification pour l'approche de détection des intrusions les plus connus sont : l'approche comportementale et l'approche par signature.

3.2 Les différents types d'IDS

Plusieurs types de technologies IDS existent, chaque type a des avantages et désavantage dans la détection, la configuration et le coût. Principalement, il y a trois familles importantes d'IDS: NIDS's, HIDS's et les IDS's hybrides

3.2.1 NIDS (Network Based IDS)

NIDS ou bien le système de détection d'intrusion destiné au réseau contrôle l'état de la sécurité du réseau et se trouve sur un réseau isolé pour contrôler ses paquets de données **[24].** En cas de détection de menace, le NIDS déclenche des alertes et ordonne des actions pour arrêter le flux de données.

On peut citer, Snort **[25]** comme un NIDS, Snort (Fig.5) est un des plus actifs NIDS Open Source et possède une communauté importante contribuant à son succès, capable d'effectuer en temps réel des analyses de trafic et de logger les paquets sur un réseau IP. Il peut effectuer des analyses de protocole, recherche par correspondance de contenu et peut être utilisé pour détecter une grande variété d'attaques et de sondes comme des dépassements de buffers, scans, attaques sur des CGI, sondes SMB, essai d'OS fingerprintings et bien plus.

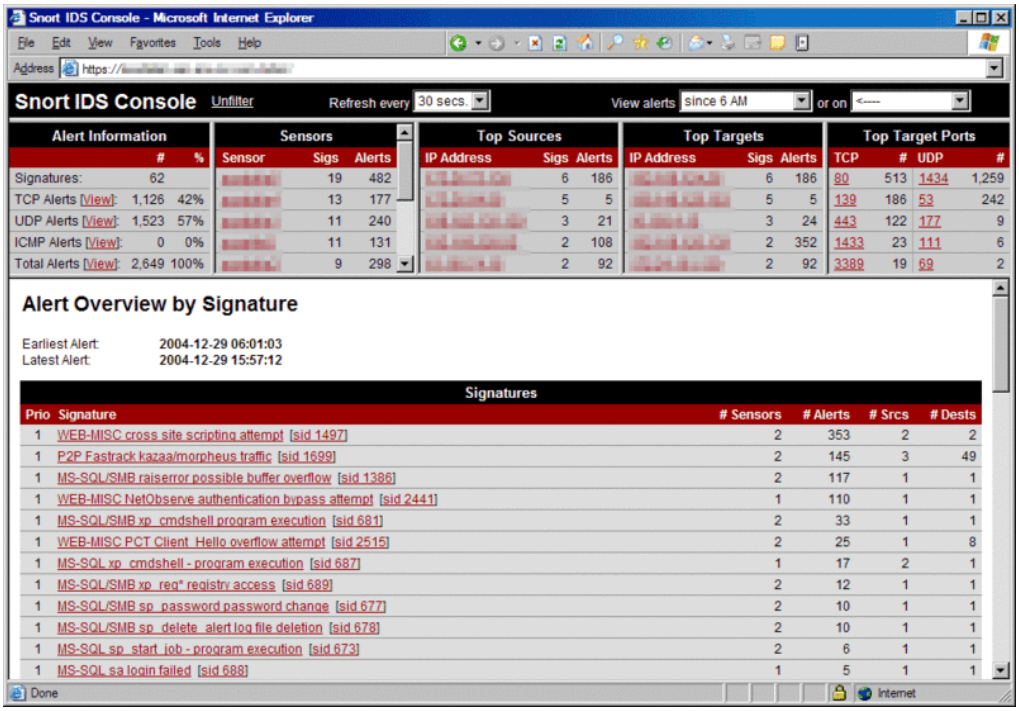

Figure.5. Snort IDS Console

Il y a aussi Bro-IDS **[26]** utilise la détection d'intrusion basée sur les anomalies, et est généralement utilisé conjointement avec Snort, car les deux se complètent très bien.

Bro est programmé de façon totalement différente de Snort, il s'appuie sur les mêmes bases théoriques (filtrage par motif), mais il intègre un atout majeur : l'analyse de flux réseau. Cette analyse permet de concevoir une cartographie du réseau et d'en générer un modèle. Ce modèle est comparé en temps réel au flux de données et toute déviance lève une alerte.

3.2.2 HIDS (Host Based IDS)

Les HIDS, (Host based IDS), ou bien système de détection d'intrusion machine, installer sure une machine et surveille l'activité se passant sur cette machin, surveille l'état de sécurit des hôtes selon déférents critères il ya tout d'abord l'activité de la machine comme par exemple le nombre et listes des processus, le nombre d'utilisateur, les ressources consommées, le seconde critère de surveillance est l'activité de l'utilisateur comme par exemple les horaires et durée des connexions, les commandes utilisées, les programmes activé de l'utilisateur… etc.**[24]**.

Et évidement le HIDS analyse toutes activités potentielles liées à l'activité d'un ver, d'un virus, cheval de Troie …etc. Un HIDS a besoin d'un système sain pour vérifier l'intégrité des données. Si le système a été compromis par un pirate, le HIDS ne sera plus efficace.

On peut citer comme HIDS, EMERALD eXpert **[27]** qui offre un niveau de surveillance de sécurité en temps pour les serveurs d'applications et les stations de travail, il fournit la base de connaissances la plus complète pour détecter les violations de règles, les abus de privilèges ou la manipulation illégale de ressources et les autres violations de politique de site pour les systèmes. Ce composant est empaqueté et distribué en tant que solution de détection d'intrusion complète, fournissant une collecte de données, une analyse de détection d'intrusion, une interface de gestion d'alertes et des directives de réponse détaillées.

3.2.3 IDS Hybride

Hybrid IDS ou HyIDS combine NIDS et HIDS **[28]**, la conception des IDS hybrides se base sur la possibilité de maximiser les forces des deux types d'IDS's (HIDS et NIDS). En pratique, un IDS hybride fonctionne comme un NIDS, par la collection et le traitement du flux réseau afin de détecter les attaques, d'un autre côté, en tant que HIDS, le système hybride se concentre sur chaque hôte, traitant uniquement les paquets adressés à son propre système. Le système hybride peut résoudre le problème de faible performance de NIDS, mais laisse toujours reste le problème HIDS (IDS doit être installé sur chaque machine).

L'exemple le plus connu dans le monde Open-Source est Prelude **[29].** Ce Framework permet de stocker dans une base de données des alertes provenant de différents systèmes relativement variés. Utilisant Snort comme NIDS, et d'autres logiciels tels que Samhain en tant que HIDS, il permet de combiner des outils puissants tous ensemble pour permettre une visualisation centralisée des attaques.

3.3 Détection d'intrusion

On distingue deux grands types d'approches pour détecter des intrusions. La première consiste à rechercher des signatures connues d'attaques tandis que la seconde consiste à définir un comportement normal du système et à rechercher ce qui ne rentre pas dans ce comportement. Un système de détection d'intrusions par recherche de signatures connaît ce qui est mal, alors qu'un système de détection d'intrusions par analyse de comportement connaît ce qui est bien **[30]**

3.3.1 L'approche par scénario

Cette technique s'appuie sur la connaissance des techniques utilisées par les attaquants pour déduire des scénarios typiques. Elle ne tient pas compte des actions passées de l'utilisateur et utilise des signatures d'attaques (une chaîne alphanumérique, une taille de paquet inhabituelle, une trame formatée de manière suspecte) **[31].**

Différentes méthodes dans l'approche par scénario **[31] [32]**

- ***Recherche de motifs (pattern matching)***

La méthode la plus connue et la plus à facile à comprendre. Elle se base sur la recherche de motifs (chaînes de caractères ou suite d'octets) au sein du flux de données. L'IDS comporte une base de signatures où chaque signature contient le protocole et le port utilisé par l'attaque ainsi que le motif qui permettra de reconnaître les paquets suspects. Le principal inconvénient de cette méthode est que seules les attaques reconnues par les signatures seront détectées. Il est donc nécessaire de mettre à jour régulièrement la base de signatures.

- ***Recherche de motifs dynamiques***

Le principe de cette méthode est le même que précédemment mais les signatures des attaques évoluent dynamiquement. L'IDS est de ce fait doté de fonctionnalités d'adaptation et d'apprentissage.

- ***Analyse de protocoles***

Cette méthode se base sur une vérification de la conformité (par rapport aux RFC) des flux, ainsi que sur l'observation des champs et paramètres suspects dans les paquets. Cependant, les éditeurs de logiciels et les constructeurs respectent rarement à la lettre les RFC et cette méthode n'est pas toujours très performante. L'analyse protocolaire est souvent implémentée par un ensemble de préprocesseurs, où chaque préprocesseur est chargé d'analyser un protocole particulier (FTP, HTTP, ICMP, ...).

- ***Analyse heuristique et détection d'anomalies***

Le but de cette méthode est, par une analyse intelligente, de détecter une activité suspecte ou toute autre anomalie. Par exemple : une analyse heuristique permet de générer une alarme quand le nombre de sessions à destination d'un port donné dépasse un seuil dans un intervalle de temps prédéfini.

3.3.2 L'approche comportementale

Cette approche se base sur l'hypothèse que l'exploitation d'une faille du système nécessite une utilisation anormale de ce système, et donc un comportement inhabituel de l'utilisateur. Elle cherche donc à répondre à la question « le comportement actuel de l'utilisateur ou du système est-il cohérent avec son comportement passé ? ».

Cette approche peut être appliquée non seulement à des utilisateurs mais aussi à des applications et services. Plusieurs métriques sont possibles : la charge CPU, le volume de données échangées, le temps de connexion sur des ressources, la répartition statistique des protocoles et applications utilisés, les heures de connexion, ...

Cette approche est peu fiable, tout changement dans les habitudes de l'utilisateur provoque une alerte, Elle nécessite une période de non fonctionnement pour mettre en œuvre les mécanismes d'auto-apprentissage **[33]**.

Différentes méthodes dans l'approche comportementale **[31]**

- ***Approche probabiliste :***

Des probabilités sont établies permettant de représenter une utilisation courante d'une application ou d'un protocole. Toute activité ne respectant pas le modèle probabiliste provoquera la génération d'une alerte.

- ***Approche statistique:***

Le but est de quantifier les paramètres liés à l'utilisateur (taux d'occupation de la mémoire, utilisation des processeurs, valeur de la charge réseau, nombre d'accès à l'intranet par jour, vitesse de frappe au clavier, sites les plus visités, ...). Elle n'est actuellement présente que dans le domaine de la recherche, où les chercheurs utilisent des réseaux neuronaux et la fouille de données pour tenter d'avoir des résultats convaincants.

Conclusion

La détection d'intrusion a pour objectif de détecter toute violation de la politique de sécurité sur un système informatique. Elle permet ainsi de signaler les attaques (en temps réel ou en différé) portant atteinte à la sécurité de ce système.

Pour mettre en œuvre ce concept de détection d'intrusion, des outils spécifiques sont nécessaires : les IDS ou systèmes de détection d'intrusions. Ils vont permettre de collecter de façon automatisée les données représentant l'activité des systèmes(serveurs, applications, systèmes, réseaux), de les analyser et d'avertir les administrateurs en cas de détection de signes d'attaques.

Les systèmes de détection d'intrusion sont devenus indispensables lors de la mise en place d'une infrastructure de sécurité opérationnelle.

L'efficacité des détections passe aussi par une bonne implémentation des algorithmes de recherche. L'étude que nous avons menée nous a permis d'observer combien il est indispensable d'utiliser des structures de données appropriées lors d'un développement.

4

Chapitre 4

Conception et Implémentation

Introduction

Dans ce chapitre on va essayer de détailler le processus qu'on va utiliser pour détecter les intrusions dans les environnements informatique d'apprentissage humaine, on va expliquer le processus d'une vue générale ensuite on passe à l'implémentation du module de détection d'intrusion.

4.1Module de détection d'intrusion (IDM)

Nous proposons de créer un système de détection d'intrusion dans les EIAH's en utilisant les services du KTBS qu'on va appeler IDM (Intrusion Detection Module), nous proposons une recherche pour la détection des intrusions basée sur des automates finis en utilisant le principe de pattern matching (recherche par motif), le module de détection d'intrusion comporte les étapes suivantes (Figure 6):

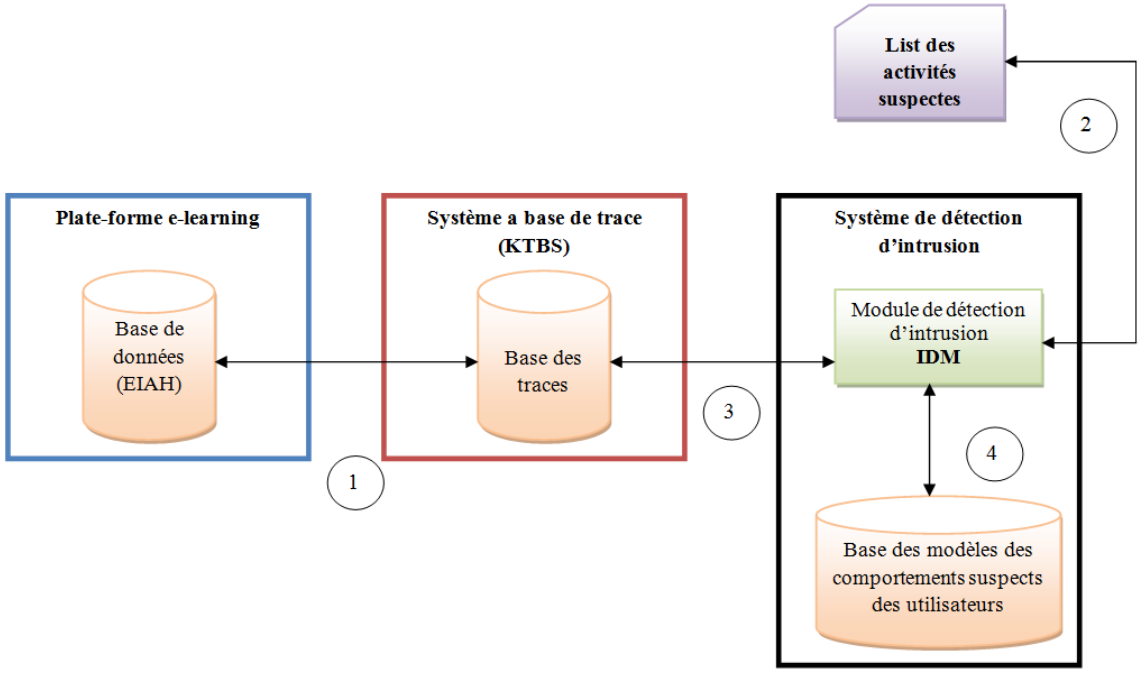

Fig.6. Module de détection d'intrusion dans une EIAH (IDM)

1- la première étape consiste à la collection des traces, pour cela nous utilisons KTBS pour créer un m-trace première à partir des données brutes des plates-formes d'apprentissage. Ce m-trace est sauvegardé dans la base des traces de KTBS.

Nous mettons à jour ce m-trace en temps réel à partir de données brutes pour assurer la qualité du système de détection d'intrusion que nous proposons.

2- pour commencer le processus de détection d'intrusion il faut choisir une séquence d'activité suspect, et faire une recherche des ces activités dans m-trace première que nous créons dans la base des traces KTBS à l'étape 1

3- Le système de détection d'intrusion vérifie la m-trace primaire que nous créons à l'étape 1. Pour détecter les intrusions, notre IDM se base sur l'approche par scénario et utilise la méthode de recherche de motifs (pattern matching), la recherche des chaînes de caractères correspond aux séquences d'activités suspects au sein du m-trace première. L'IDS comporte une base de modèles des comportements suspects des utilisateurs, chaque modèle contient une séquence d'activité suspecte.

La détection d'intrusion se fait se forme d'une automate infinie (Figure 7),

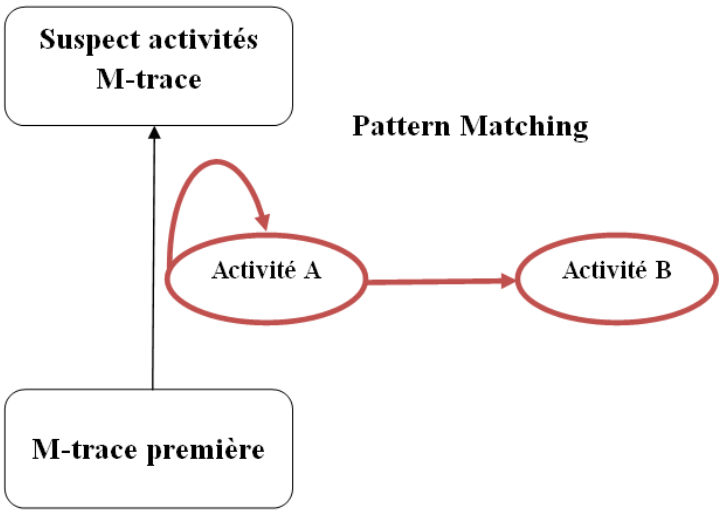

Fig.7. Exemple d'un pattern matching pour recherche une activité suspect de séquence AB

4- Enfin, la liste des activités des utilisateurs suspects est affichée sur le système, chaque fois un comportement anormal est détecté, le système d'intrusion ajoute cette anomalie à la base des modèles des comportements suspects des utilisateurs liste des activités suspectes, question de mettre à jour régulièrement la base des modèles et cela rend notre IDS plus puissant.

D'une vue générale notre IDS fonction sous deux faces essentielle la première consiste a la collection des traces d'activités, en la deuxième face qui consiste à la détection des intrusions

4.2Diagramme de classe et Diagramme de séquence :

D'une vue générale notre IDS fonction sous deux faces essentielle la première consiste a la collection des traces d'activités, en la deuxième face qui consiste à la détection des intrusions, pour cela on peut diviser notre IDS sous forme de deux sous-systèmes chacun son diagramme de classe et diagramme de séquence

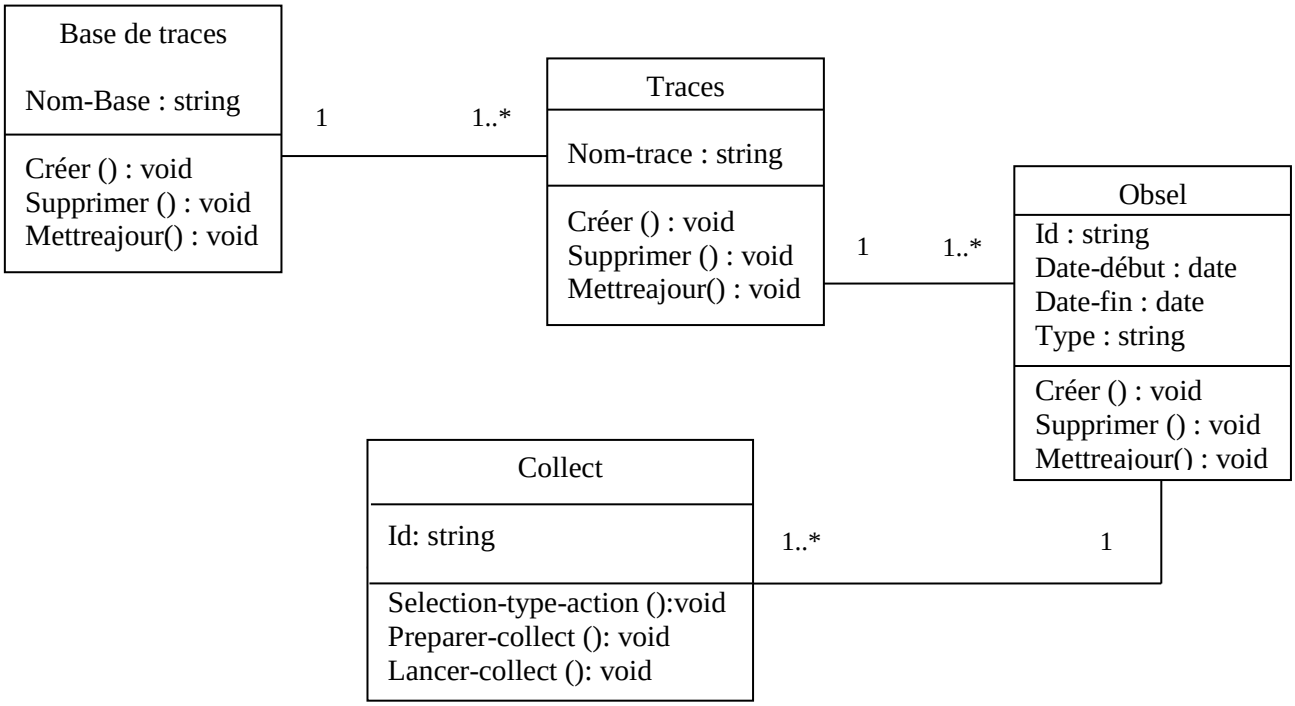

Fig.8. Diagramme de classe pour la collecte des traces

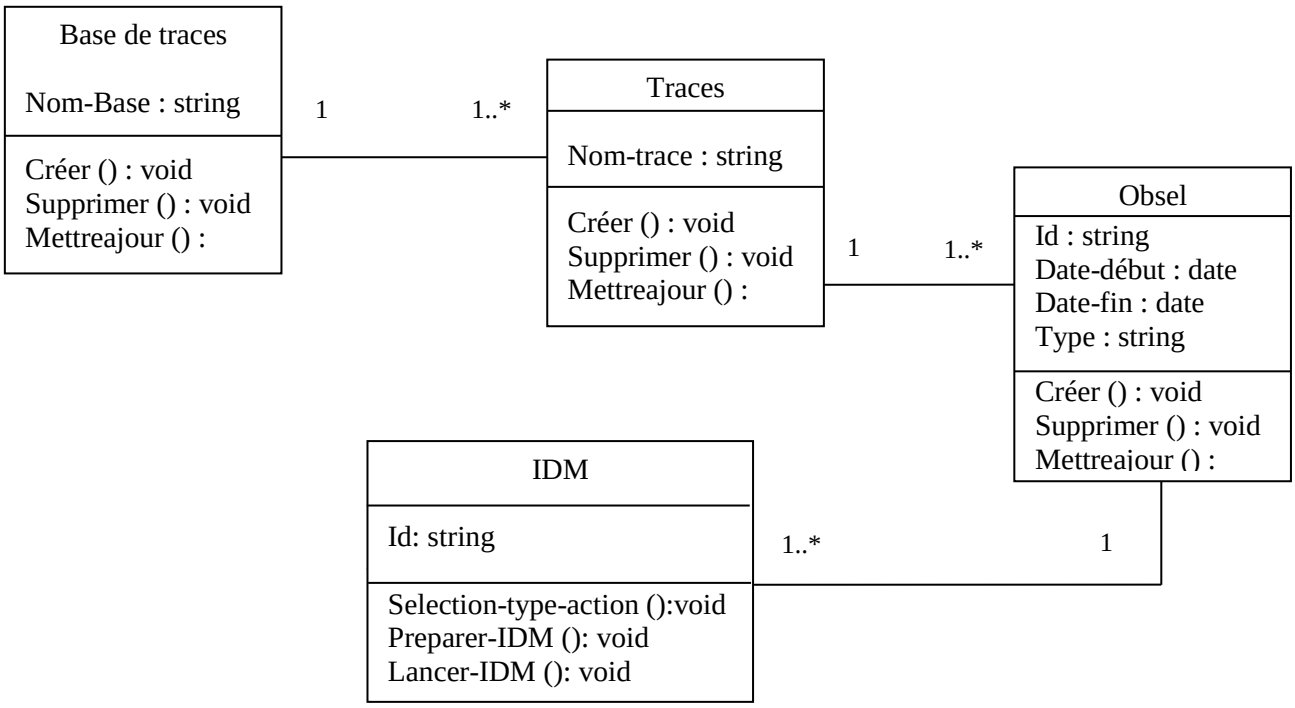

Fig.9. Diagramme de classe pour la détection des intrusions

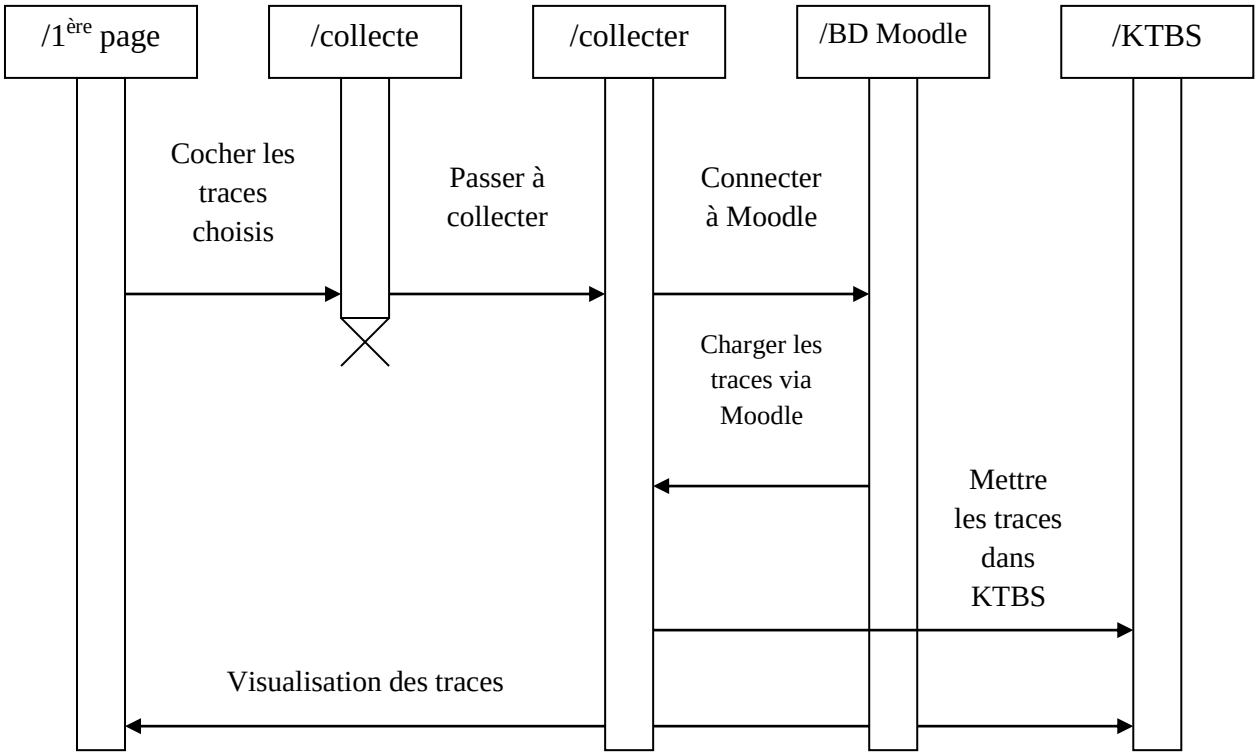

Fig.10. diagramme de séquence pour la collecte des traces

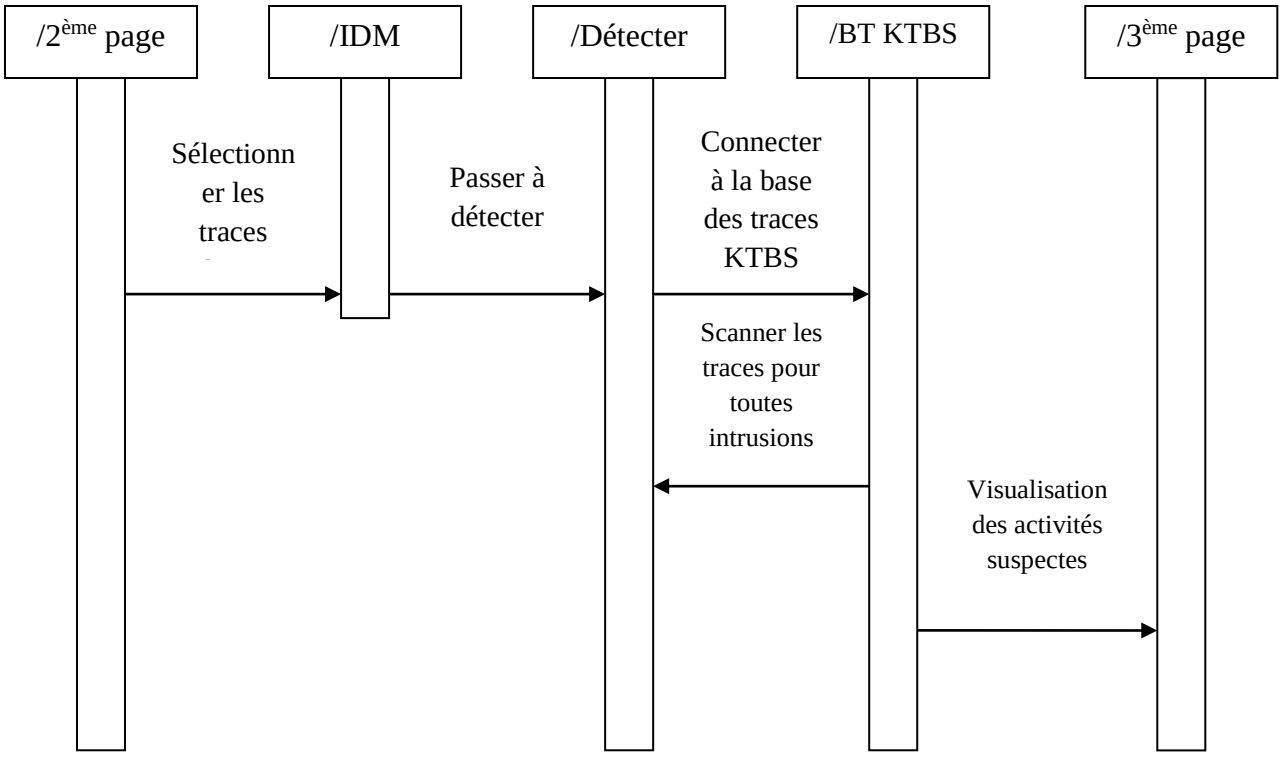

Fig.11. diagramme de séquence pour la détection des intrusions

4.3Etude de cas: Détection d'intrusion dans Moodle

À titre d'exemple comme un EIAH on va utiliser Moodle, qui se trouve la plate-forme la plus utilisée dans les Universités et les grandes écoles, et propose un ensemble d'outils d'activités tels que chats, forums, wikis, devoir, test, sondage, etc., et une grande base de données (plus de 200 tables MySQL).

Nous suivrons ces étapes pour construire notre système:

- On travail sous ubuntu **[34]** comme système d'exploitation qui est une GNU/linux basé sure la distribution linux debian avec une interface simple, intuitive, et sécurisée, plusieurs version sont disponible gratuitement sur le web.
- Nous utilisons KTBS pour gérer la base de m-traces. KTBS est l'implémentation de TBS. Pour l'installer il faut suivre les étapes sur le site officiel **[35].**
- Par la suite l'installation de la plate-forme Moodle **[36]**, et la création d'un environnement informatique d'apprentissage humaine

Pour développer notre système de détection d'intrusion nous avons utilisé PHP comme langage de programmation simplifier l'interaction avec le KTBS et Moodle, le lancement de notre IDS (Figure 12) donne une la page d'accueil de notre IDS qui donne des liens

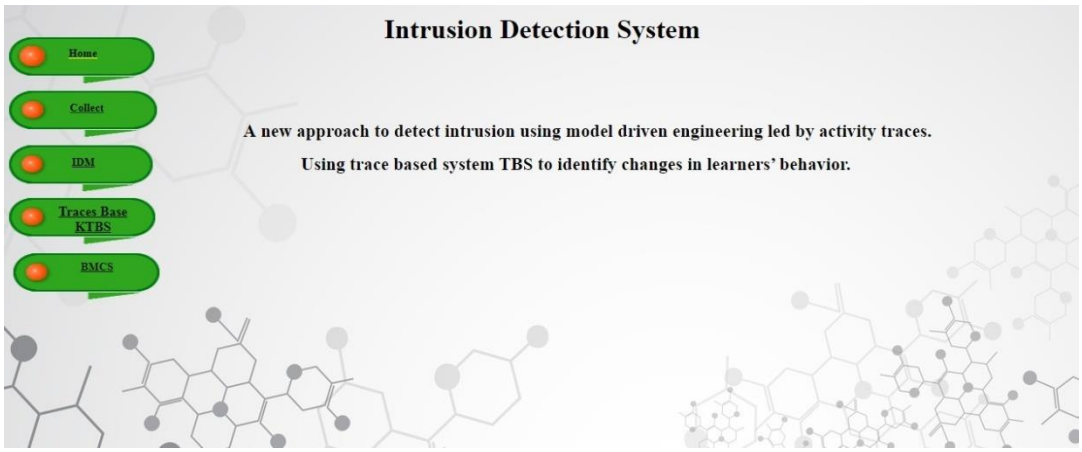

Fig.12. page d'accueil IDS

- **Home:** la page d'accueil de notre IDS.
- **Collect :** lien vers le système de collecte des traces.
- **IDM :** Intrusion detction module lien vers le module de détection des intrusions.
- **KTBS Trace Base:** lien vers la base des traces KTBS.
- **BMCS :** lien vers la base des modèles des comportements suspects des utilisateurs.

Nous commençons par la collecte des traces en accèdent au lien du sous système de collecte (Figure 13), La collecte se fait par cours, en choisissons le nom de cours, et la date de déroulement de cours, ainsi que la liste des activités a tracés.

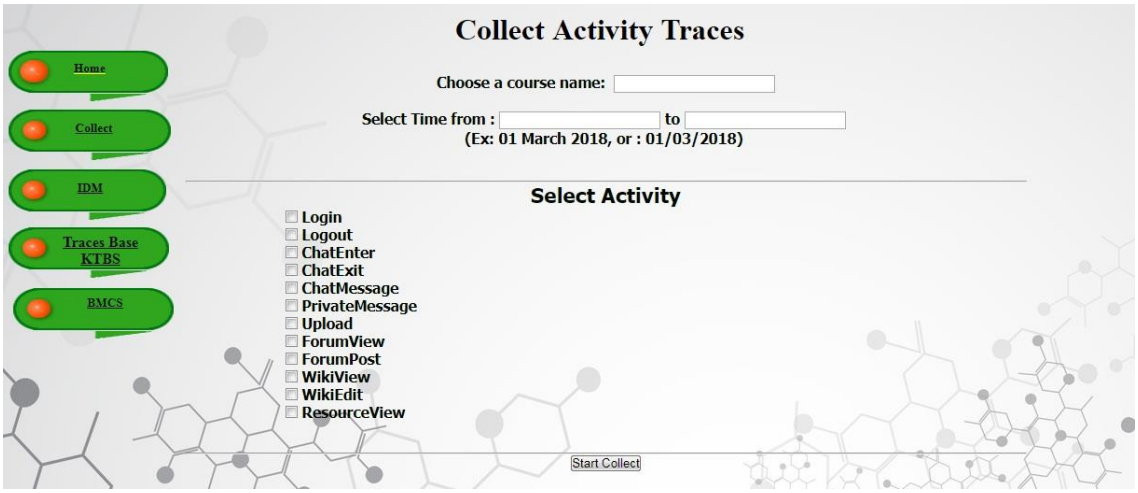

Fig.13. Lien ver le sous système Collecte

Pour vérification la consultation de la base des traces KTBS (via le lien Traces Base KTBS) nous donne notre trace collectée avec une série des activités qui correspond à la liste des obsels (pour notre exemple on trouve des obsels qui correspond aux activités **ChatEnter** et **WikiView**)

Par la suite on passe au lien du module de détection d'intrusion IDM (Figure 14), on commence la recherche d'éventuelle intrusion par la sélection d'une séquence de l'activité suspecte.

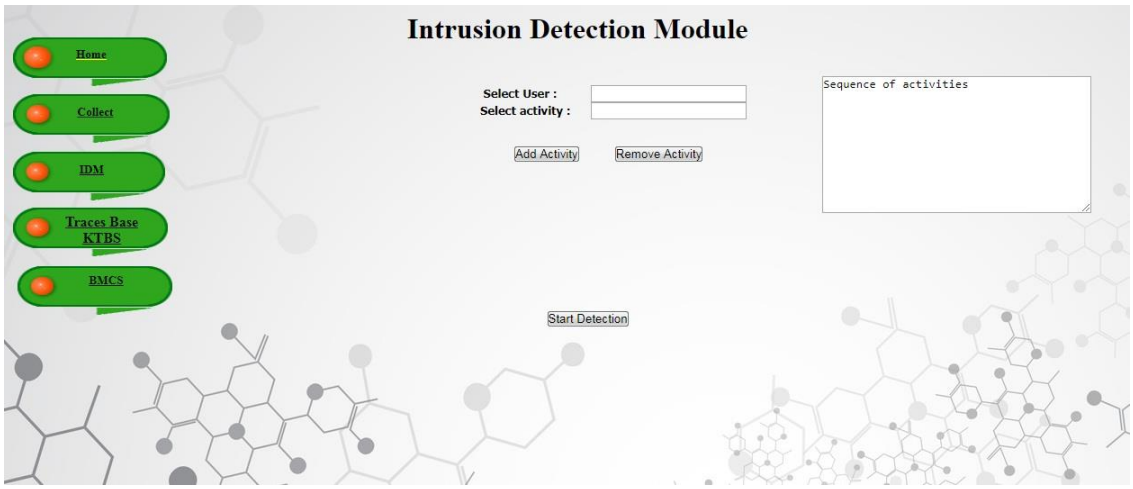

Fig.14. Lien ver le module de détection d'intrusion

Le résultat de la recherche s'affiche, et elle soit l'une des deux cas

1- **Cas positif** : intrusion détecté et séquence des activités ajouté à la base des modèles de comportements suspects des utilisateurs (Figure 15), (la séquence d'activité suspect est ChatEnter et WikiView).

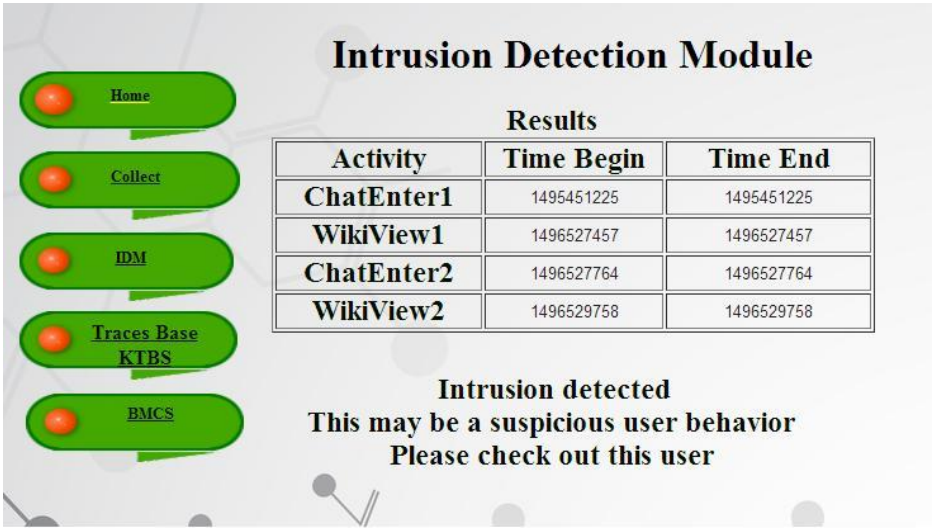

Fig.15. Exemple de comportement suspect, ''intrusion détecté''

2- Cas négatif : aucune intrusion détecté (Figure 16), (la séquence d'activité suspect est **WikiEdit** et **ResourceView**)

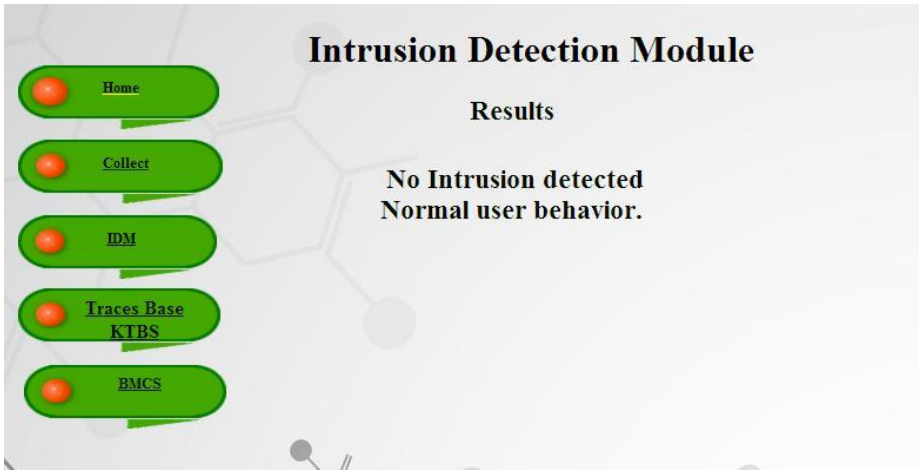

Fig.16. Exemple de comportement normale, ''aucune intrusion détecté''

Les deux autres liens, KTBS Trace Base pour consulter la base des traces, et BMCS pour consulter la base des modèles de comportement suspect des utilisateurs.

Conclusion

Dans ce chapitre nous avons implémenté notre méthode pour la détection des intrusions dans les environnements informatiques d'apprentissage humain, comme environnement nous avons choisi Moodle.

Conclusion générale

Les environnements informatiques d'apprentissages humains jouent un rôle grandissant dans notre société. Un grand nombre d'université et d'école possède des leur propre EIAH et donne l'accès à un grand nombre d'étudiant, enseignant et invité. La sécurité des EIAH's devient alors une problématique essentielle tant pour les individus que pour les entreprises de formation ou les universités. Il est donc important de définir une politique de sécurité pour ces EIAH's et de veiller à son respect. Néanmoins les mécanismes de sécurité préventifs mis en place ne sont pas incontournables. Il est nécessaire de mettre en œuvre des outils permettant de détecter toute violation de la politique de sécurité, c'est-à-dire toute intrusion. Ces outils sont appelés des systèmes de détection d'intrusions ou IDS.

Notre travail s'inscrit dans le domaine de la détection d'intrusions, de manière essentielle, et permet une certaine tolérance aux intrusions. Nous proposons l'utilisation des traces pour la détection des intrusions en se basent sur le changement du comportement de l'utilisateur.

Une approche pour détecter les intrusions à l'aide d'une ingénierie dirigée par des modèles, guidée par les traces d'activité, nous avons pris en compte dans notre modèle général de détection d'intrusions les spécificités liées à l'utilisation des traces

Nous avons réalisé une implémentation de cette approche dans Moodle et nous avons évalué pratiquement la pertinence et la fiabilité de notre IDS.

Au cours de la réalisation de ce modeste travail nous avons eu la chance d'aborder le sujet des EIAH's et leur importance phénoménale dans les centres de formations, les écoles et les universités, il s'agit d'apprendre tout en étant connecté avec des machines, des enseignants, tuteurs, apprenants, … etc. Nous avons aussi vu le TBS (système à base de traces) qui est considéré comme un système informatique permettant et facilitant l'exploitation des traces, ces derniers qui s'avèrent très important dans le domaine des EIAH, par la suite nous avons vu le système de détection d'intrusion, ces modèles et ces techniques de détection d'intrusion.

Par la suit nous avons donnez le résultat de notre implémentation, notre système de détection d'intrusions dans Moodle, à partir du comportement de l'utilisateur, qui se trouve très efficace pour renforcé la sécurité de la plate-forme Moodle et la sécurité des EIAH's en générale, mais comme toutes système en voie de développement ya toujours des inconvénients.

Les principales inconvénients pour notre système c'est qu'il se base sur l'approche comportementale pour cela tout changement dans les habitudes de l'utilisateur provoque une alerte , se qui nous rend vers les cas de **faux positif** qui sont des alertes qui ne correspondent pas à des attaques réelles, et le cas de **faux négatifs** qui sont des intrusions réelles qui n'ont pas été détectée, et pour remédier à ces inconvénients nous proposons de faire un période

d’essaie pour collecter toutes comportement siens possible a fins d’élargir notre base des traces le plus possibles, et de mettre à jour régulièrement la base des modèles de comportement suspectes des utilisateurs. Comme ca on peut améliorer notre IDS et le proposer comme un plugin pour la plate-forme Moodle.

Bibliographie

[1] *Pierre Tchounikine. Pour* ***une ingénierie des Environnements Informatiques pour l'apprentissage Humain****. LIUM - Université du Mans .Avenue Laennec - 72085 Le Mans cedex-9.France. 2002.*

[2] *Pierre Tchounikine.* ***Quelques éléments sur la conception et l'ingénierie des EIAH****. Actes des 2ème assises nationales du GdR I3 - Groupe de Recherche Information Interaction Intelligence, décembre 2002, 2002, Nancy, France. 13 p., 2002.*

[3] *Cherkaoui, C., (2006).* ***Les Environnements Informatiques d'Apprentissage Humain : Fondements, problématiques, Méthodologies et Ingénierie, Habilitation à diriger des Recherches****, Soutenue le 18 Février 2006.*

[4] *Tchounikine & al, P. (2004). Platon-1 :* ***quelques Dimensions pour l'analyse des travaux de recherche en conception d'EIAH. Rapport de l'Action Spécifique : Fondements théoriques et méthodologiques de la conception des EIAH*** *.département STIC du CNRS.*

[5] *Ollagnier -Beldame, M., Mille, A. (2007).* ***Faciliter l'appropriation des EIAH par les apprenants via les traces informatiques d'interactions****. Rapport de recherche RR-LIRIS 2007-023, Soumis à sticef pécial traces.*

[6] *Mellet-D'huart, D., Michel, G. (2006). Réalité virtuelle et apprentissage. In GRANDBASTIEN M.& LABAT J.-M. (Eds.)* ***Les environnements Informatiques pour l'Apprentissage Humain****. Editeur Hermes, Collection "Traité IC2 Information Commande Communication", 2006.*

[7] *Amal Battou, C. Cherkaoui, Driss Mammass****. Approche granulaire des objets pédagogiques en vue de l'adaptabilité dans les EIAHs. Environnements Informatiques pour l'Apprentissage Humain****. Faculté des Sciences d'Agadir, 2012*

[8] *Tarek Djouad.* ***Ingénierie des indicateurs d'activités à partir de traces modélisées pour un environnement informatique d'apprentissage humain****. Thèese de doctorat université Claude Bernard - Lyon I, 2011.*

[9] *Lotfi Sofiane SETTOUTI,* ***Systèmes à Base de Traces Modélisées: Modèles et Langages pour l'exploitation des traces d'Interactions****, thèse de doctorat université Claude Bernard Lyon 1, 2011.*

[10] ***Institut de la Gestion et du Développement Economique*** *– site web: http://www.institut.minefi.gouv.fr.*

[11] ***Sharable Content Object Reference Model****, The Advanced Distributed Learning (ADL) site web: http://adlnet.gov/research/scorm/.*

[12] *Documentation de la plateforme MOODLE : Modular Object-Oriented Dynamic Learning Environment. site web: https://docs.moodle.org/3x/fr/Accueil.*

[13] ***Le grand dictionnaire terminologique*** *site web: http://www.granddictionnaire.com/.*

[14] *Tarek Djouad, Lotfi Sofiane Settouti, Yannick Prié, Christophe Reffay, Alain Mille.* ***Un Système à Base de Traces pour la modélisation et l'élaboration d'indicateurs d'activités éducatives individuelles et collectives. Mise à l'épreuve sur Moodle****.. TSI, 2010, 6, 29, pp.721-741.*

[15] *Documentation KTBS: a kernel for Trace-Based Systems, site web: https://kernel-for-trace-based-systems.readthedocs.io/en/latest/.*

[16] *LS Settouti, Y Prié, A Mille, JC Marty -* ***Systèmes à base de traces pour l'apprentissage humain****, Communication in the international TICE, Technologies, 2006.*

[17] *Li, Qiang.* ***Modélisation et exploitation des traces d'interactions dans l'environnement de travail collaborati****v, Thèse de doctorat université de technologie de Compiègne (2013).*

[18] *Zarka, Raafat and Amélie Cordier.* ***Trace-based reasoning for user assistance and recommendations****, Thèse de doctorat, université de lyon, 2013.*

[19] *Hajer Chebil.* ***Corpus de traces d'activité dans les environnements informatiques pour l'apprentissage humain : modélisation, étude d'une plateforme de gestion, et application à la construction de corpus de référence****, Thèse de doctorat Ecole Nationale Supérieure des Mines de Saint-Etienne, 2013.*

[20] *Rémi Casado, Nathalie Guin, Pierre-Antoine Champin, Marie Lefevre.* ***kTBS4LA : une plateforme d'analyse de traces fondée sur une modélisation sémantique des traces. Méthodologies et outils pour le recueil, l'analyse et la visualisation des traces d'interaction*** *- ORPHEE-RDV, Jan 2017, Font-Romeu, France.*

[21] *https://www.w3.org/TeamSubmission/turtle/.*

[22] *https://www.w3schools.com/js/js_json_intro.asp.*

[23] *Tian, Xueqi Cheng, Miyi Duan, Rui Liao, Hong Chen, Xiaojuan Chen,* ***Network intrusion detection based on system calls and data mining****, Frontiers of Computer Science in China, December 2010, Volume 4, Issue 4, pp 522–528.*

[24] *B. Santos Kumar et al,* ***Intrusion Détection System- Types and Prévention****, (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 4 (1) , 2013.*

[25] ***Snort*** *- Network Intrusion Detection & Prevention System site web: https://www.snort.org.*

[26] *The* ***Bro*** *Network Security Monitor site web: https://www.bro.org*

[27] *Event Monitoring Enabling Responses to Anomalous Live Disturbances (****EMERALD****) site: www.csl.sri.com/projects/emerald/*

[28] *Megha Gupta.* ***Hybrid Intrusion Detection System: Technology and Development****. International Journal of Computer Applications (0975 – 8887) Volume 115 – No. 9, April 2015.*

[29] *Welcome to the Prelude Universal Open-Source SIEM project site web: https://www.prelude-siem.org*

[30] *Philippe Biondi,* ***Architecture expérimentale pour la détection d'intrusions dans un système informatique****, 2001.*

[31] *Claude Duvallet* ***Les systèmes de détection d'intrusions réseaux*** *Université du Havre, 2014.*

[32] *David Burgermeister, Jonathan Krier,* ***Les systèmes de détection d'intrusion****, 2006.*

[33] *Jonathan-Christofer Demay.* ***Génération et évaluation de mécanismes de détection des intrusions au niveau applicatif****. Thèse de doctorat Université Rennes 1, 2011.*

[34] Système d"exploitation Ubuntu site web: https://www.ubuntu.com/.

[35] Tutoriel pour installer KTBS site web: https://kernel-for-trace-based-systems.readthedocs.io/en/latest/tutorials/install.html.

[36] Tutoriel pour installer Moodle site web https://docs.moodle.org/34/en/Step-by-step_Installation_Guide_for_Ubuntu.