Use the Brute_Force Pattern Matching Algorithm for Misuse Intrusion Detection System

Haleema Essa Sulaiman
haleema_essa@uomosul.edu.iq
College of Computer Sciences and Mathematics
University of Mosul, Mosul, Iraq

Received on: 19/07/2018 **Accepted on: 24/01/2019**

ABSTRACT

Security issues, like network intrusion and viruses, have been increased widely with the growth of computer applications and networks. Therefore, it becomes necessary to develop methods to protect information from malicious attacks within the different environments. One of these methods is to use intrusion detection system for the detection of different interventions.

The research was presented a way to detect misuse intrusion (Misuse Detection System), as was performed classification of events, which will be either the events of Normal Events or Intrusion Events. This classification process has been based on one of the String Pattern matching Algorithms, which is Brute_Force algorithm.

Brute_Force algorithm is used after making a comparison between this algorithm and another two algorithm (Knuth – Morris – Pratt String Matching and Boyer-Moore Algorithm).

Data processed in the work is taken from the KDD list. The written version of this data, which will be similar to the data format in the comma separated values files (CSV), This data has been converted to tables and then a comparison between these tables is made for the purpose of categorizing events based on the algorithm mentioned above. Java language has been used in this work as one of the most powerful programming languages, has been the adoption of Eclipse environment to write Java classes used in the work.

Keywords: intrusion detection, misuse intrusion detection, string pattern matching algorithms, tables comparison, events classification, java language.

استخدام خوارزمية مطابقة الانماط Brute_Force لنظام كشف تطفل سوء الاستخدام

حليمة عيسى سليمان
كلية علوم الحاسوب والرياضيات
جامعة الموصل، الموصل، العراق

تاريخ الاستلام: 2018/07/19 **تاريخ القبول: 2019/01/24**

الملخص

قدم البحث طريقة لكشف سوء الاستخدام (Misuse Detection System)، إذ اجري تصنيف Classification للأحداث التي ستكون إما أحداث اعتيادية Events Normal أو أحداث تطفلية Intrusion Events، وعملية التصنيف هذه تمت بالاعتماد على إحدى خوارزميات مطابقة أنماط السلاسل (String Pattern matching Algorithms)، التي تدعى خوارزمية Brute_Force.

اسنخدمت هذه الخوارزمية في النظام المقترح كونها الأكفأ عند التعامل مع بيانات النظام مقارنة بخوارزمية (Knuth – Morris – Pratt String Matching) وخوارزمية (Boyer-Moore Algorithm).

البيانات التي تم التعامل معها في العمل مأخوذة من بيانات KDD، وقد استفيد من الصيغة المكتوبة بها هذه البيانات التي ستكون بطبيعتها مشابهة لصيغة البيانات الموجودة في ملفات CSV files (Comma Separated Values Files)، اذ حولت هذه البيانات إلى جداول ومن ثم أجريت عملية المقارنة بين هذه الجداول لغرض القيام بعملية تصنيف الأحداث بالاعتماد على الخوارزمية المذكورة سابقاً.

استخدمت لغة الجافا في هذا العمل باعتبارها من أقوى اللغات البرمجية، وقد اعتمدت بيئة Eclipse لكتابة أصناف الجافا المستخدمة في العمل.

الكلمات المفتاحية: كشف تطفل، كشف سوء الاستخدام، خوارزميات مطابقة انماط السلاسل، مقارنة الجداول، تصنيف الاحداث، لغة الجافا.

1- مقدمة

إن تعقيد أنظمة الحواسيب الموزعة وأهميتها وموارد المعلومات المتوافرة فيها تمت بسرعة كبيرة، استناداً إلى هذه الحقيقة فقد أصبحت الحواسيب وشبكاتها هدفاً لجرائم الحاسوب التي ازدادت أكثر فأكثر [1].

كلما ازداد فهم المتطفل لكيفية عمل النظام كلما أصبح ماهراً في تحديد نقاط الضعف في النظام ثم يستغلها للحصول على مستوى شرفية عالٍ الذي يستطيع به أن يسيطر على النظام. إن المتطفلين يستخدمون أنماطاً من التطفل التي من الصعوبة تحديدها أو مراقبتها [2].

استخدمت بعض التقانات لحماية البيانات المهمة مثل الجدار الناري والتشفير والخ، إن أي تقانة أمنية جديدة تحوي في تصميمها على بعض العيوب مما يجعلها هدفاً للهجمات، لذلك فإن من المهم امتلاك أنظمة كشف التطفل لحماية البيانات المهمة. وعليه أصبحت نظريات كشف التطفل في الشبكة الحاسوبية ذات اهتمام كبير لدى العديد من الباحثين في السنوات الأخيرة [3].

نظام كشف التطفل Intrusion Detection System (IDS) مكون مهم في إطار أمنية الحاسبة والمعلومات، وإن هدفه الرئيس هو تمييز بين الفعاليات الاعتيادية للنظام والسلوكيات التي ممكن أن نصفها على أنها تطفل، إن أي نظام لكشف التطفل لا يهمل استخدام التقانات الأمنية الأخرى ولكن يعمل كخط دفاع أخير في النظام [3].

يجهز (IDS) فائدتين أساسيتين هما الوضوحية (Visibility) والسيطرة (Control). وفي حالة دمج هذه الفوائد يمكن تقوية السياسة الأمنية للمؤسسة التي تعتمد على أمنية الشبكة الحاسوبية الخاصة. إن الوضوحية هي القدرة على ملاحظة وفهم طبيعة حركة سير الحزم ضمن الشبكة الحاسوبية، بينما السيطرة هي السيطرة على حركة سير الحزم في الشبكة الحاسوبية وتتضمن الوصول إلى الشبكة الحاسوبية أو جزء منها [4].

استخدم الباحثون العديد من التقانات في بناء أنظمة كشف التطفل؛ اذ هنالك طرائق إحصائية (Statistical Approaches) [5] وتقانات ذكائية اصطناعية (Artificial Intelligence Techniques) [6].

تستخدم خوارزميات مطابقة الأنماط (Pattern Matching Algorithms) لحل المشاكل المعقدة التي تكون غير قابلة للحساب أو التحليل المباشر وتطبيقات هذا المجال تشمل خوارزميات البحث المتسلسل Linear Search, البحث الثنائي Binary Search, Boyer-Moore, Prefix Searching, Aho-Corasick, Soundix Searching, أنظمة كشف التطفل، إثبات النظريات، الألعاب وغيرها من المجالات، في هذا البحث

استخدمت خوارزمية مطابقة أنماط السلاسل (Brute_Force String Pattern Matching Algorithm) في عملية كشف التطفل وتصنيفه، واستخدمت لغة الجافا في هذا البحث، وبالاعتماد على الجداول التي كوِنت باستخدام الصنف JTable ادخلت البيانات التي حصل عليها من مجموعة بيانات KDD CUP 99 التي تعتبر الأكثر استخداماً في تقييم نظم كشف التطفل.

يمكن إيجاز أهم الأهداف للبحث بالنقاط الآتية:

1. تصميم نموذج لكشف سوء الاستخدام وتنفيذه (Misuse Detection) الذي له القدرة على تصنيف مجموعة أحداث النظام إلى صنفين حدث اعتيادي أو حدث تطفلي.
2. إيجاد طريقة سهلة للتعامل مع ملفات(csv files) (comma separated value files) وذلك بتحويلها إلى جداول من نوع Jtable ؛ اذ أن التعامل مع الجداول أسهل بكثير من التعامل مع بيانات تعتمد مثل هذه الصيغ.
3. استخدام خوارزمية مطابقة الأنماط Brute Force لغرض تصنيف الاحداث، إذ استخدمت هذه الخوارزمية كونها أسهل وأنسب خوارزمية للتعامل مع مثل هذه البيانات التي تضمنها الجداول.
4. استخدام الأدوات الأفضل والأكفأ لإجراء عملية المقارنة التي تتمثل باستخدام لغة الجافا والتي شغلت باستخدام بيئة eclipse لتنفيذ الانموذج الامني.

2- نظام كشف التطفل Intrusion Detection System

تواجه جدران النار عدداً من نقاط الضعف منها أنها عملية (نعم أو كلا)، وهذا يعني أن ما يقوم به جدار النار هو منع المتطفل من الدخول إلى نظام الهدف، وأنه يعمل على فتح منفذ (Port) محدد فقط، في حين يغلق كل المنافذ الأخرى. وأخيراً، لا يمكن لجدار النار اكتشاف الاختراق الأمني عند تعلقه بحركة المرور التي لا تمر من خلاله، ولا يمكنه أن يفحص حركة المرور بين أي جهازين داخل الشبكة.

وتعد نقاط الضعف المذكورة سابقاً محفزات باتجاه استخدام آلية أمنية أخرى تعمل على الشبكة، ويتوجب أن تكون أكثر ذكاءً من جدار النار؛ إذ إنها ترفع تحذيراً لأي متطفل وتكشف عن الحزم كلها في الشبكة في وقت حقيقي، وتعمل على غلق الجلسات غير القانونية. ويمكن لهذه الآلية أن تحذر من الاختراق الحاصل في حركة المرور في الشبكة الداخلية. هذه الآلية هي نظام كشف التطفل، يضاف إلى ذلك أنه يمكن عدها طبقة دفاعية ثانية لحماية الشبكة [7].

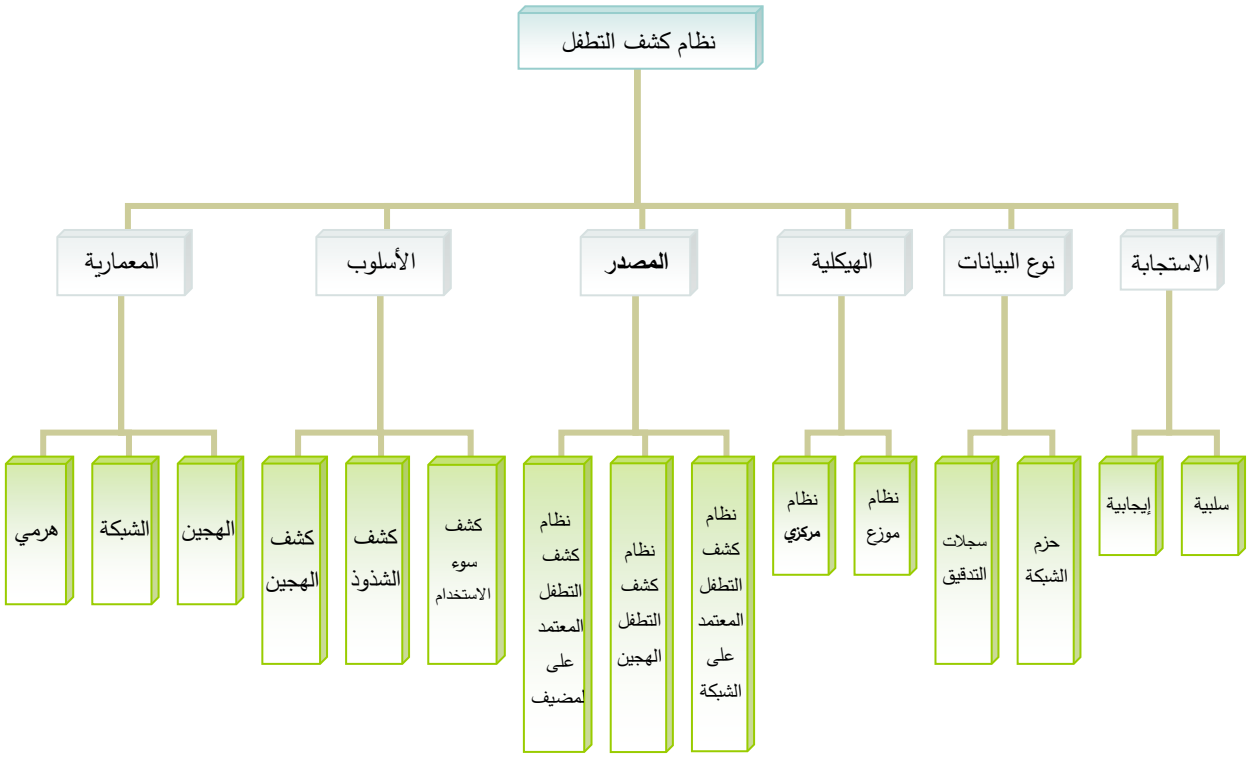
وتُعرف أنظمة كشف التطفل بأنها المُكون الذي يقوم بتحليل عمليات النظام وعمليات المستخدم في الحاسوب وأنظمة الشبكة للبحث عن نشاطات غير مرغوب فيها من وجهة نظر أمنية. وتُعد طريقة نظام كشف التطفل طريقة مُرخصة في تحديد المستخدمين غير القانونيين وتعريفهم، والمهاجمين الذين يمكنهم التأثير الفعال لأنظمة الحاسوب. وتقوم أنظمة كشف التطفل بالكشف عن عدد من التطفلات وتنفذ قسماً من الأعمال المحددة مسبقًا عندما يتم الكشف عن التطفل [8].

وأنظمة كشف التطفل عبارة عن منتجات من البرامج الحاسوبية أو المعدات الحاسوبية التي تعمل على مراقبة العملية وتحليلها أوتوماتيكياً. ويعمل أي نظام للكشف عن التطفل على التحقق من أي نشاط متجه إلى داخل

الشبكة أو خارجها، وسجلات النظام والأحداث، وتحديد النماذج المشتبه بها أو الأحداث التي قد تشير إلى وجود هجوم على النظام أو الشبكة من شخص ما يحاول الدخول عنوة أو يعمل على تخريب النظام.

3- تصنيف نظام كشف التطفل Classification of Intrusion Detection System

هناك عدة تصنيفات ودراسات متلاحقة للكشف عن التطفل. وعلى الرغم من هذه الجهود السابقة حتى الوقت الحاضر فإن الكشف عن التطفل لايزال يفتقر إلى تصنيف وقابلية للتطبيق [8,9]. ويبين الشكل (1) تصنيف نظام أنموذجي للكشف عن التطفل.



الشكل (1): تصنيف نظام كشف التطفل [9].

4- كشف إساءة الاستخدام Misuse detection

يشير كشف إساءة الاستخدام إلى التقانات التي تميز أساليب معروفة لاختراق النظام. ويمكن وصف هذه الاختراقات بأنها أنموذج *pattern* أو توقيع *signature* يقوم نظام كشف التطفل بالبحث عنه؛ إذ يكون هذا الأنموذج أو التوقيع عبارة عن سلسلة ثابتة أو مجموعة من الإجراءات وتستند استجابات النظام على الاختراقات التي تم تحديدها[10].

هناك أربع مراحل رئيسة لتحليل عملية كشف إساءة الاستخدام:

- **المعالجة الأولية *Pre-processing***: الخطوة الأولى هي تجميع بيانات حول عمليات الاختراق ونقاط الضعف والاعتداءات ووضعها في مخطط تصنيفي أو مخطط وصف الأنماط, من مخطط التصنيف الأنموذج السلوكي و يوضع بعد ذلك في شكل محدد.

- **التحليل Analysis** : تقارن بيانات السلوك مع قاعدة المعرفة باستخدام محرك التحليل *Analysis Engine* الذي يقوم بمطابقة الأنماط *pattern-matching* يقوم هذا المحرك بالبحث عن النماذج التي يتم اعتبارها هجمات.
- **الاستجابة Response** : إذا تطابقت بيانات السلوك مع نماذج الهجمات يقوم محرك التحليل بإرسال تنبيه إلى مسؤول النظام.
- **دقة تطابق البيانات Accuracy** : إن دقة تطابق البيانات ***pattern-matching*** تعتمد على تحديث التواقيع، فكلما كان التحديث سريعاً كان نظام كشف التطفل أفضل في التعرف على الهجمات، ويشابه في هذه الصفة البرامج المضادة للفيروسات *Antivirus*. وتسمح أنظمة كشف التطفل بالتحديث اليدوي أو الأوتوماتيكي للتواقيع الخاصة بالهجوم [11].

هناك ثلاثة أساليب مختلفة للكشف الخاص بإساءة الاستخدام:

1- **مطابقة النمط *Pattern-matching***: تحديد الهجمات عن طريق عمليات فحص مرور البيانات وتحديد توقيع نماذج الهجمات المعروفة.

2- **تحليل البروتوكول *Protocol analysis*** : التدقيق يكون أيضا عبر مراقبة مرور البيانات ومشاهدة أي معلمات مشتبه بها تكون مشابهة لبعض البروتوكولات لذلك يجب التأكد من أنها لن يستخدمها المهاجم بطريقة خبيثة *malicious*.

طريقة التخمين *Heuristic method* : تكون هذه النظرية مقاربة لنظرية كشف الفيروسات في الأنظمة المضادة للفيروسات *Antivirus systems* [11,12].

من **محاسن** أنظمة كشف إساءة الاستخدام أن التحليل الذي يستخدم في مطابقة الأنماط يكون أكثر كفاءة من تقانة كشف الشذوذ بسبب عدم وجود حسابات للقياسات الإحصائية، ولكن في المقابل هنالك بعض **المساوئ** لأنظمة كشف الاستخدام، مثلاً مطلوب تحديث قاعدة البيانات للنماذج بصورة مستمرة كلما كانت هناك أصناف جديدة من الهجمات، وأيضاً هنالك

مشكلة في توسيع النظام بسبب عدم وجود لغة وصف للأغراض العامة لوصف النمـــاذج

وكذلك من الصعب تحويل وصف اللغة الطبيعية للإساءة إلى أنموذج [13].

5- وصف مجموعة بيانات KDD CUP 99

كل حزمة اتصال من الـ*KDD99* تتكون من41 ميزة فضلاً عن الميزة 42 التي تصف الحزمة من حيث كونها طبيعية أو هجوماً عن طريق ذكر نوع محدد من أنواع الهجمات، وبيانات التدريب تحتوي على 22 نوعاً من الهجمات بينما بيانات الاختبار تحتوي على 39 نوعاً من أنواع الهجمات؛ اذ ان الهجمات الموجودة في بيانات التدريب تسمى بالهجمات المعروفة *known attack*، أما الهجمات الإضافية الموجودة في بيانات الاختبار تسمى بالهجمات الجديدة *novel attacks* وتكون غير متوافرة في بيانات التدريب[14]. والجدول رقم (1) يوضح الهجمات المعروفة والهجمات الجديدة.

الهجمات جميعها سواء أكانت معروفة أم جديدة تقع في واحدة من الفئات الأربعة الآتية:

1. ***Denial of Service Attack (DoS)*** : هجوم منع الخدمة أو الحرمان من الخدمة

هو أن يقوم المهاجم بعمل بعض الحسابات وجعل المصادر مشغولة جدا أو جعل الذاكرة ممتلئة أو منع وصول المستخدمين القانونيين إلى الحاسبات.

2. ***Probing Attack :*** هو محاولة لجمع معلومات من أجهزة الحاسوب لمعرفة نقاط الضعف في الحاسبة والتحايل عليها.
3. ***User to Root Attack (U2R):*** يقوم المهاجم باستغلال المستخدم العادي للنظام حيث يدخل النظام بصلاحيات المستخدم العادي مثلاً عن طريق الحصول على كلمة السر ومن ثم استغلال بعض نقاط الضعف للوصول إلى جذر النظام.
4. ***Remote to Local Attack (R2L) :*** تكون للمهاجم القدرة على إرسال حزم إلى حاسبة من خلال الشبكة ولكن ليس له حساب في تلك الحاسبة ومن ثم يقوم باستغلال نقاط الضعف للدخول إلى تلك الحاسبة بوصفه مستخدم طبيعي لها [14,15].

الجدول(1) الهجمات المعروفة والهجمات الجديدة

DOS	PROBE	R2L	U2R
الهجمات المعروفة			
Back, land, Neptune, Pod, smurf, teardrop	ipsweep, satan, nmap, portsweep	ftp_write, guess_passwd, warezmaster, warezclient, imap, phf, spy, multihop	Rootkit, loadmodule, buffer_overflow, perl
الهجمات الجديدة			
Apache2, udpstorm, processtable, mailbomb	Saint, mscan	Named, xlock, sendmail, xsnoop, worm, snmpgeattack, snmpguess	Xterm, ps, sqlattack, httptunnel

6- خوارزميات مطابقة أنماط السلاسل String Pattern Matching Algorithms:

6-1 خوارزمية Brute_Force Searching Algorithm.

6-2خوارزمية Knuth – Morris – Pratt String Matching.

6-3 خوارزمية Boyer-Moore Algorithm.

شرحت هذه الخوارزميات وبشكل مفصل ضمن الملحق الخاص بخوارزميات مطابقة الأنماط.

وبعد الاطلاع على الخوارزميات التي تستخدم في عملية مطابقة أنماط السلاسل يتضح أن الخوارزمية الأكفأ والأنسب للتعامل مع بيانات النظام هي خوارزمية Brute_Force والتي استخدمت عند المقارنة بين بيانات

الجداول المستخدمة في النظام المقترح وكما هو موضح في الجدول (2)؛ اذ ان كلتا الخوارزميتين KMP و BM ستحتاج إلى عملية تحليل النمط قبل البدء بعملية البحث بالإضافة إلى حاجة خوارزمية KMP إلى مصفوفات تحتوي على عدد الأحرف المراد مقارنتها مع النمط الحالي، وايضاً عند استخدام خوارزمية BM سيستخدم جدول الإزاحة الذي يكون بطول مكافئ لطول السلسلة المراد البحث عنها ضمن النمط، وهذا بدوره سيؤدي إلى زيادة في استخدام مصادر النظام وحجز مواقع إضافية في الذاكرة، وباستخدام خوارزمية Brute_Force سيتجاوز هذه المشاكل وخاصة أن البيانات التي يتم التعامل في النظام كبيرة الحجم الا وهي بيانات KDD Data Set فضلا عن الاستخدام الأمثل لمصادر النظام.

الجدول(2) مقارنة بين خوارزميات مطابقة الأنماط

الاحتياجات	KMP	BM	Brute_Force
عملية تحليل النمط	تحتاج لعملية تحليل النمط	تحتاج إلى عملية تحليل النمط	لاتحتاج إلى عملية تحليل النمط
الحاجة إلى مصفوفات	تحتاج إلى مصفوفات	لاتحتاج إلى مصفوفات	لا تحتاج إلى مصفوفات
استخدام جدول الازاحة	لا تستخدم جدول الإزاحة	تستخدم جدول الإزاحة	لا تستخدم جدول الإزاحة
استخدام مصادر النظام	زيادة في استخدام مصادر النظام	زيادة في استخدام مصادر النظام	استخدام مصادر النظام بكفاءة

7- الخوارزمية المتبعة في Brute_Force (Brute_Force Algorithm):

Input text ***T*** of size ***n*** and pattern

P of size ***m***

n=T. length

m=P. length

Output starting index of a

Substring of ***T*** equal to ***P*** or −1

If no such substring exists

For i = 0 to n−m

8- خطوات عمل النظام:

إن تنفيذ النظام اعتمد على خوارزمية مطابقة أنماط السلاسل Brute-force، باستخدام بيئة eclipse وباعتماد لغة الجافا من خلال الخطوات الآتية:

الخطوة الأولى: قمنا بتهيئة الملفات النصية الخاصة بالعمل والمأخوذة من بيانات KDD، وإنشاء ملفين، الأول يحتوي احداثاً تطفلية ومن ثم برمجياً يحول الملف النصي إلى ملف بامتداد CSV file (ملفات اكسل بامتداد csv.) ومن ثم إنشاء الجدول الأول عن طريق صنف الجافا JTable.

والجدول الثاني سيحوي أحداثاً تطفلية وأحداثاً اعتيادية وأيضاً سيحول إلى ملف csv ومن ثم إلى جدول باستخدام الصنف JTable.

الخطوة الثالثة: ادخال الجدولين إلى خوارزمية Brute-force لكي تتم عملية المقارنة وتبيان هل هي احداث تطفليه او طبيعية، لذلك فان خطوات تنفيذ النظام ستكون على النحو الآتي:

أ. إذا كانت النتيجة مطابقة (أي بمعنى أن الحدث هو تطفل)، عندئذ سيقوم النظام بعرض النتيجة أسفل الشاشة كلمة intrusion.

ب. أما إذا كانت النتيجة غير مطابقة (أي بمعنى أن الحدث هو حدث طبيعي)، عندئذ سيقوم النظام بعرض النتيجة أسفل الشاشة كلمة Normal.

ويمكن الاطلاع على الواجهات التنفيذية للنظام من خلال الملحق الخاص بالنتائج.

9_ الاستنتاجات Conclusions

من خلال تصميم الانموذج الأمني وتنفيذه في البحث لغرض كشف التطفل ووفق النتائج التي تم الحصول عليها، نستنتج ما يأتي:

1- بعد الاطلاع على الصيغة المكتوبة بها بيانات KDD، التي ستكون مشابهة لصيغة ملفات CSV، اتضح ان هناك صعوبة في التعامل مع هذه الصيغ من الناحية البرمجية لذلك حولت إلى صيغة سهلة التعامل حيث تم تحويلها إلى جداول؛ إذ إن التعامل مع الجداول يكون أسهل بكثير من التعامل مع بياناتٍ CSV.

2- إن استخدام خوارزمية ال brute-force في إجراء عملية مطابقة أنماط السلاسل على هذه الجداول كان الطريقة الأكفأ للحصول على النتائج وتصنيف الأحداث.

3- إن استخدام بيئة Eclipse سهل الصعوبات التي واجهت العمل بوصفها بيئة لتنفيذ برامج الجافا؛ إذ إن لها واجهات رسومية غنية بالقوائم والتسهيلات التي يمكن استخدامها في العمل. فضلاً عن أن تكوين جداول النظام تم بطريقة مباشرة باستخدام صنف الجافا (JTable)، دون الحاجة إلى إضافة مكتبات جافا جديدة تستخدم في عملية تصميم الجداول.

المصادر

[1] د.علاء حسين الحمامي، د.سعد عبد العزيز العاني (2007) ، "تكنولوجيا أمنية المعلومات وأنظمة الحماية "، جامعة عمان الأهلية .

[2] Najla Badie Ibraheem Al-Dabagh, (2006)," Modeling and Implementation of Intrusion Detection Techniques", Ph.D. Thesis, University of Mosul, Computer Sciences .

[3] Adel Sabry Issa, (2009), "A Comparative Study among Several Modified Intrusion Detection System Techniques ", Master Thesis, University of Duhok.

[4] Sahar Lazem Kadoory, (2009), " Design and Implementation of an Embedded Intrusion Detection System (IDS) for Wireless Application", Master Thesis, University of Mosul, Computer Engineering .

[5] Yongro Park, (2005), " A Statistical Process Control Approach for Network Intrusion Detection ", Ph.D. Thesis, The Academic Faculty, Georgia Institute of Technology .

[6] Matti Manninen, (2007), " Using Artificial Intelligence in Intrusion Detection Systems ", Helsinki University of Technology .

[7] DIDS(Distributed Intrusion Detection System)-Motivation, Architecture, and An Early Prototype Snapp (2017), SR; Brentano, J; Dias, G; Goan, TL; Heberlein, LT; Ho, CL; Levitt, KN

[8] Gogoi P, Bhattacharyya DK, Borah B, Kalita K. MLH-IDS (2015), a multi-level hybrid intrusion detection method. The Computer Journal ; 57:602–623.

[9] Wafa' S. Al-Sharafat, Reyadh Sh.Naoum, (2009), " Adaptive Framework for Network Intrusion Detection by Using Genetic-Based Machine Learning Algorithm", Al Al-Bayt University, Information Technology College, Jordan .

[10] Host-based misuse intrusion detection using PCA feature extraction and kNN classification algorithms(2018), Serpen, Gursel* | Aghaei, Ehsan, Electrical Engineering and Computer Science, University of Toledo, Toledo, OH 43606, USA

[11] Saidat Adebukola Onashoga, Adebayo D. Akinde, and Adesina Simon Sodiya, 2009, "A Strategic Review of Existing Mobile Agent-Based Intrusion Detection Systems", University of Agriculture, Nigeria, Issues in Informing Science and Information Technology Volume 6.

[12] Misuse Detection in a Simulated IaaS Environment(2018), Burhan Al-BayatiNathan Clarke, Part of the Lecture Notes in Computer Science book series (LNCS, volume 11263)

[13] An Effective Intrusion Detection System Using Flawless Feature Selection, Outlier Detection and Classification(2018), Rajesh Kambattan Kovarasan, Manimegalai Rajkumar, art of the Advances in Intelligent Systems and Computing book series (AISC, volume 713)

[14] Analysis of NSL-KDD Dataset Using K-Means and Canopy Clustering Algorithms Based on Distance Metrics,(2018), H. P. Vinutha, B. Poornima, Part of the Studies in Computational Intelligence book series (SCI, volume 771)

[15] M. Sabhnani and G. Serpen, (2004), " Why Machine Learning Algorithms Fail in Misuse Detection on KDD Intrusion Detection Data Set", Intelligent Data Analysis, 8(4):403-415.

[16] Anithakumari, S,2009, Anithakumari S Dept. of Computer Science & Engg. Dept. of Computer Science & Engg. TKM College of Engg. LBSITW, Poojappura Kollam, Kerala Thiruvananthapuram, Kierala {IVSL}

[17] Intrusion Detection System Using Pattern Matching Techniques for Wireless Sensor Networks(2018),Jayashree Agarkhed, Gauri Kalnoor Siddarama R. Patil, Part of the Lecture Notes in Networks and Systems book series (LNNS, volume 32)

[18] IEEE Xplore Full-Text HTML : Research and optimization of pattern matching algorithm based on Intrusion Detection System {IVSL}

[19] IEEE Xplore Abstract - A Composite Boyer-Moore Algorithm for the String Matching Problem {IVSL}

الملحق الخاص بخوارزميات مطابقة الانماط

1-خوارزمية Brute_Force Searching Algorithm

تعتبر أكفا الخوارزميات وأبسطها وفكرتها تكون عن طريق مقارنة حرف من النمط مع حرف مع النص، فإذا تطابق الحرفان انتقل إلى الحرف التالي في كل من النمط والنص. أما في حال لم يتطابقا يحرك مؤشر الحرف في النص إلى الحرف التالي ويرجع مؤشر الحرف في النمط إلى البداية، وسوف تبدأ عملية المقارنة مرة أخرى.

وسوف تستمر عملية هذه إلى أن يتم إيجاد التطابق في كل حروف النمط ، أو الوصول لنهاية النص. وسيشار إلى طول النمط بالحرف M أما طول النص سوف يتم الاشارة له بالحرف N. وسيتوقف البحث عندما نصل ل N–M؛ لأنه عند الوصول إلى تلك الخانة وحتى إذا كان الحرف التالي متطابقاً يتم التوقف؛ لأن النص سوف يكون أصغر من النمط.

الشكل (1) يوضح عمليه البحث عن النمط needle، إذ قورن الحرف الأول في النمط مع الحرف الأول من الtext وكانت النتيجة صحيحة، وسينتقل إلى الحرف التالي في كل من text وpattern. وهنا عندما تقارن فالنتيجة غير صحيحة وبالتالي يحرك مؤشر النص إلى الأمام، ويرجع النمط إلى لبداية، وتبدأ عملية البحث مرة أخرى. من الممكن ملاحظة عدد مرات الshift وهي 3 مرات حيث تم الوصول إلى موقع الحرف المطابق في ال Index ثلاثة [16] .

```
h a y n e e d s a n n e e d l e x
n e e d l e
 n e e d l e
  n e e d l e
   n e e d l e
    n e e d l e
     n e e d l e
      n e e d l e
       n e e d l e
        n e e d l e
         n e e d l e
          n e e d l e
```

الشكل (1) البحث عن نمط معين باستخدام خوارزمية ال Brute_Force Search

تقوم هذه الخوارزمية بمقارنة كل حرف مع الحرف الآخر، وفي حال لم يتطابقا ستعاد عملية المقارنة لجميع حروف النمط مع الحرف التالي من النص، فإذا كان طول النص عدد N ووالpattern عدد M فسوف يكون ال Big–O لهذه العملية هو N*M.

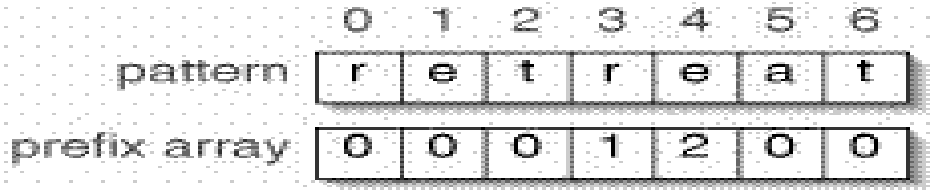
2- خوارزمية Knuth – Morris – Pratt String Matching:

في عام 1977 نشر الثلاثي Knuth (صاحب كتاب Art of programming الشهير) وMorris وPratt مقالة Fast Pattern Matching in Strings وتحدثوا فيها عن ايجاد خوارزمية أسرع من الطريقة التي تعتمد على Brute-Force وسميت هذه الخوارزمية باسم هؤلاء الأشخاص واختصاراً KMP.

بالنظر قليلا إلى Brute-Force يمكن ملاحظة أنها عندما تجد حرفاً في النمط لا يطابق الحرف في النص فإنها تقوم بالذهاب إلى الحرف التالي من النص وتعيد المقارنة من أول النمط. خوارزمية KMP جاءت لتحسين تلك الخوارزمية حيث تم التخلص من عمليات المقارنة المتكررة، طريقه عملها على النحو الاتي:

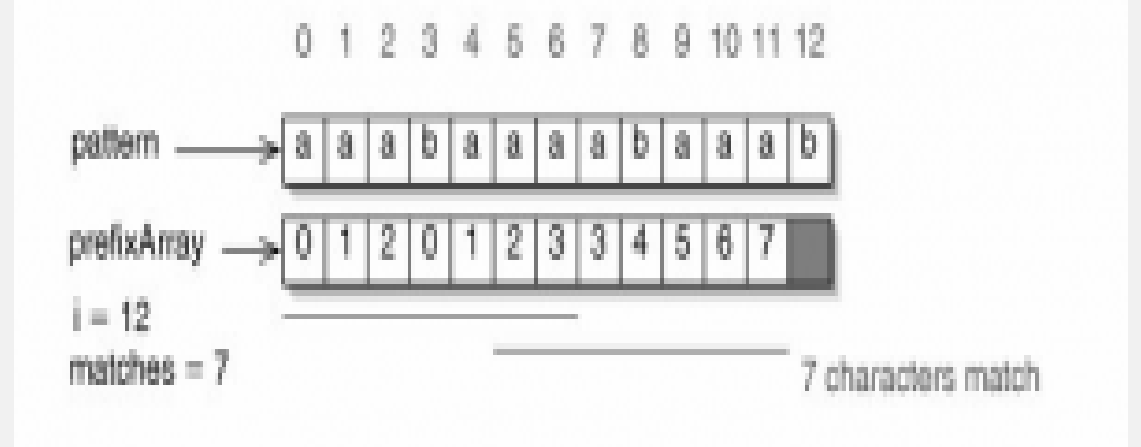
أولاً تقارن الحرف الأول في النمط مع الحرف الأول من النص في حالة كانا مختلفين ينتقل إلى الموقع التالي من النص (كما في Brute-Force بالضبط) ولكن الاختلاف يكون في حال قمنا بمقارنة ثلاثة حروف من النص مع النمط، وكانت النتيجة صحيحة ولكن الحرف الرابع يختلف ، هنا في هذه الحالة لن تتم كما في الbrute-force إعادة مقارنة جميع الحروف مع الحرف التالي من النص، لكن بما أن هذه الحروف تطوبقت سابقاً فيتم الذهاب إلى الموقع التالي من النص الذي يشابه هذه الحروف التي تطوبقت [17].

في البداية يتوجب حساب Prefix للنمط وسوف يستخدم عند مقارنة النمط مع النص، إذ إن هذه prefix array تحتوي على عدد الأحرف المشابهة من أول النمط للحرف الحالي من النمط، الشكل(2) يوضح كيف يمكن حسابه:



الشكل(2) كيفية حساب الPrefix Array

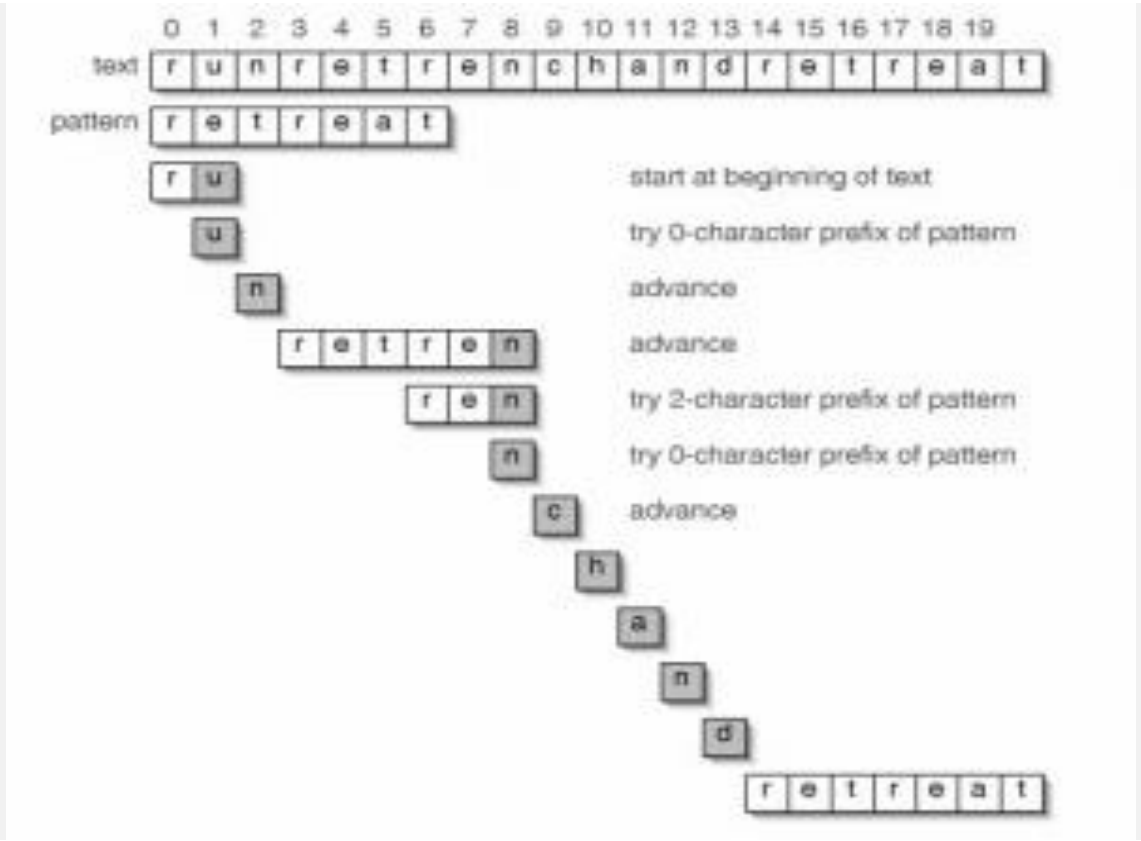
يمكن ملاحظة الموقع 3 (الحرف r) يحتوي على 1 والسبب أن الحرف r موجود في الموقع 0. 1 تشير إلى عدد الأحرف الprefix . أيضاً الموقع التالي 4 (الحرف e) يحتوي على 2 والسبب أن الحرفين في الموقعين 0و1 مشابهان للحرفين في الموقع 3و4. يعني الprefix تعني إذا كانت هناك عدد n من الحروف من أول النمط مشابه لعدد n من الحروف في الحرف الحالي من النمط إذاً سيكتب ذلك العدد n في الprefix array . مثال أخر كما في الشكل(3):



الشكل(3) كيفية حساب الPrefix Array

كما هو موضح مسبقاً فإن هذه الprefix array سيستخدم في خوارزميه البحث عندما لا يتطابق الحرف من النمط مع الحرف من النص وبالتالي يتم الرجوع إلى الخانة المكتوبة في الPrefix array وبالتالي لن تكون هناك عملية المقارنة للأحرف التي أجريت عملية المقارنة عليها [16].

الشكل(4) يوضح كيفية البحث عن النمط في النص:



الشكل(4) كيفية البحث عن النمط في النص الأصلي

في هذه الخوارزمية ، في البدء سيقارن الحرف الأول من النمط مع الحرف الأول من النص، عندما يكونان متطابقين، يتم الذهاب إلى الحرف التالي في كل من النمط والنص. تجرى عملية المقارنة وسيلاحظ أن الحرفين غير متطابقين، إذاً في هذه الحالة سوف يكون الرقم الموجود في الخانة الأولى في prefix array وسيبدأ من هذا الرقم (وهو في الحالة 0)، اذاً سيختبر مرة أخرى من الحرف r. الاختبار يكون مع u ويمكن ملاحظة أنهم غير متساويين، مرة أخرى يختبر مع الحرف التالي n وبالامكان ملاحظة أنهما غير متساويين ثم يختبر مرة أخرى مع r وبالامكان ملاحظة أنهما متساويين، يختبر الحرف التالي في كل من pattern وtext وسيستنتج أن الحروف e وt وr وe متطابقين، يتم الانتقال للحرف التالي وهو a في pattern وn في text ، يختبر وسيلاحظ أنهم غير متساويين؟ إذاً ينتقل إلى الرقم الموجود في الخانة match−1 (المتغير match يشير إلى Index الحرف في pattern) في prefix array وهو 2، وسيختبر مره أخرى الحرف t من الpattern مع الحرف n من ال text؟ سيلاحظ أنهم غير متطابقين، ينتقل الآن إلى الخانة match−1 في ال prefix Array وهو هنا 0 أي الحرف r، سيختبر مع الحرف n في text؟ والنتيجة غير متطابقين، وسينتقل إلى الحرف التالي وهو c وسيلاحظ أنه غير كذلك، يتم الانتقال للحرف التالي ثم التالي إلى أن يتم الوصول للحرف r. وسيكون التطابق للحرف r وما بعده.

3- خوارزمية Boyer−Moore Algorithm :

تعد هذه الخوارزمية إحدى أسرع الخوارزميات المستخدمة في عمليه البحث عن النصوص وسميت بذلك بالطبع نسبه لمخترعي هذه الخوارزمية حيث قدموها في عام 1977 بديلاً للطريقة البحث البدائية أو ما يسمى ب Naive Searching.

المقارنة في هذه الخوارزمية تعتمد على البدء من اليمين إلى اليسار وليس كما هو الحال مع الخوارزميات العادية، فضلاً عن انها تقلل كثيراً من عمليات المقارنة خصوصاً في حال لم يكن الحرف في text موجود في الـ pattern إذ تقفز بمقدار معين نقوم بحسابه وسيتبين الآن، هذا المقدار سوف يكون في جدول الإزاحة skip (أو بالاسم الصحيح Bad−Character Shift) بمعنى أن هناك عمليه تحليل لـ pattern قبل البدء في البحث، كما هو الحال مع KMP.

جدول الإزاحة يجب أن يكون حجمه مساوياً لحجم ال character set التي سيتكون منها النص text والنمط pattern، وبما أنه حاليا سيبحث عن الحروف الإنجليزية سوف يكون حجم الجدول بعدد حروف ASCII وهي 256. وقبل البدء في البحث في النص، يجب أن يحلل النمط وتعبئة الحرف المقابل في الجدول للحرف المقابل للنمط بمقدار ظهور آخر index للنمط.

المثال التالي يوضح طريقة البحث، فإذا كانت character set مكونه من الحروف A،B، C، D، E والنمط هو DECADE، سوف يكون شكل الجدول كما في الشكل(5)

A	B	C	D	E
3	-1	2	4	5

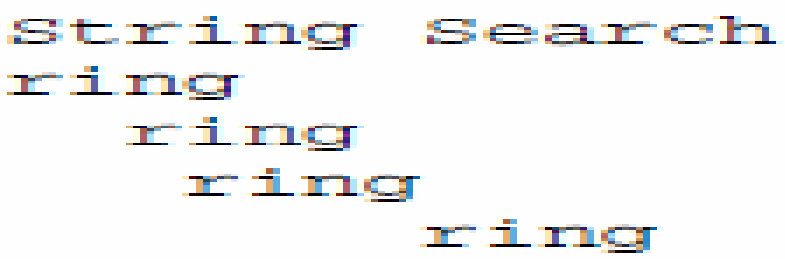
الشكل(5) جدول الإزاحة

حسب هذا الجدول على النحو الآتي، يتم البدء من اليسار لليمين ويوضع ال Index لكل حرف في ال pattern في الجدول. أولاً D موقعه هو 0، ووضع 0 في الخانة المناسبة للحرف D في الجدول (الخانة الثالثة – الترقيم من 0–). ثانياً الحرف E موقعه هو 1 ووضع 1 في الخانة المناسبة للحروف E في الجدول. وتستمر هذه العملية.

يمكن ملاحظة أن الحرف D سيتكرر مرة ثانية وسيكتب الIndex الجديد (4) في الخانة القديمة، وبنفس الأمر للحرف E. وهذا هو أساس عملية التحليل في البداية، فقط المطلوب موقع آخر ظهور للحرف، وهذا هو المطلوب في عملية البحث في حال اختلف الحرف من النص مع النمط.

يمكن أن تكون بقية الحروف في ال character set التي لا توجد لها قيم وسيوضع لها (1–). [17,18]

طريقه البحث في خوارزميه Boyer–Moore تكون كما توضح بالشكل(6):

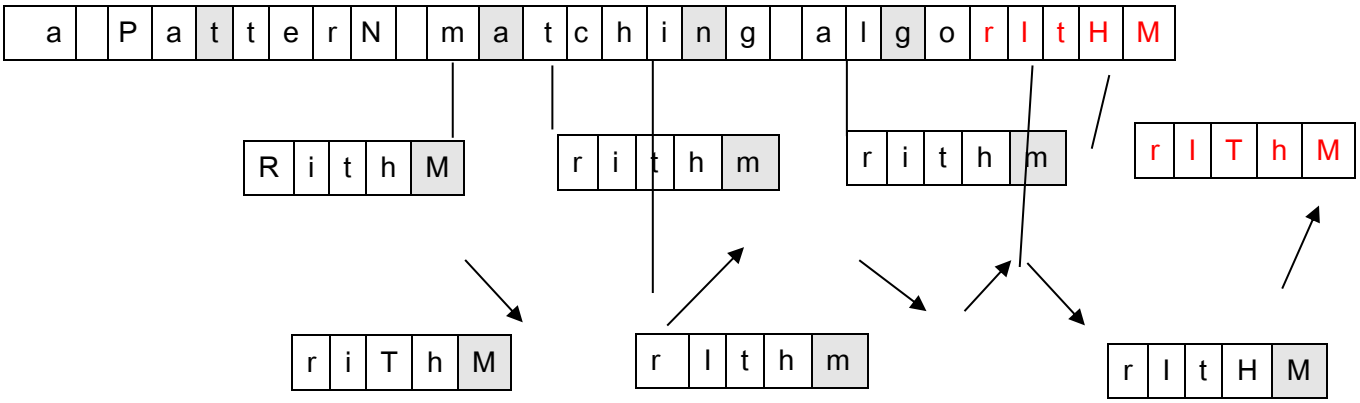


الشكل(6) طريقة البحث في خوارزميةBM

أولاً البحث كما ذكر في هذه الخوارزمية سوف يكون من اليمين لليسار بمعنى أول حرف سوف يتم البدء بمقارنته هو الحرف i من text والحرف g من pattern. في حال لم يتساوى الحرفان والحرف i موجود في النمط سوف يحرك النمط حرفين إلى الأمام وهكذا يتطابق i مع i. وهكذا تطابقت الكلمتان.

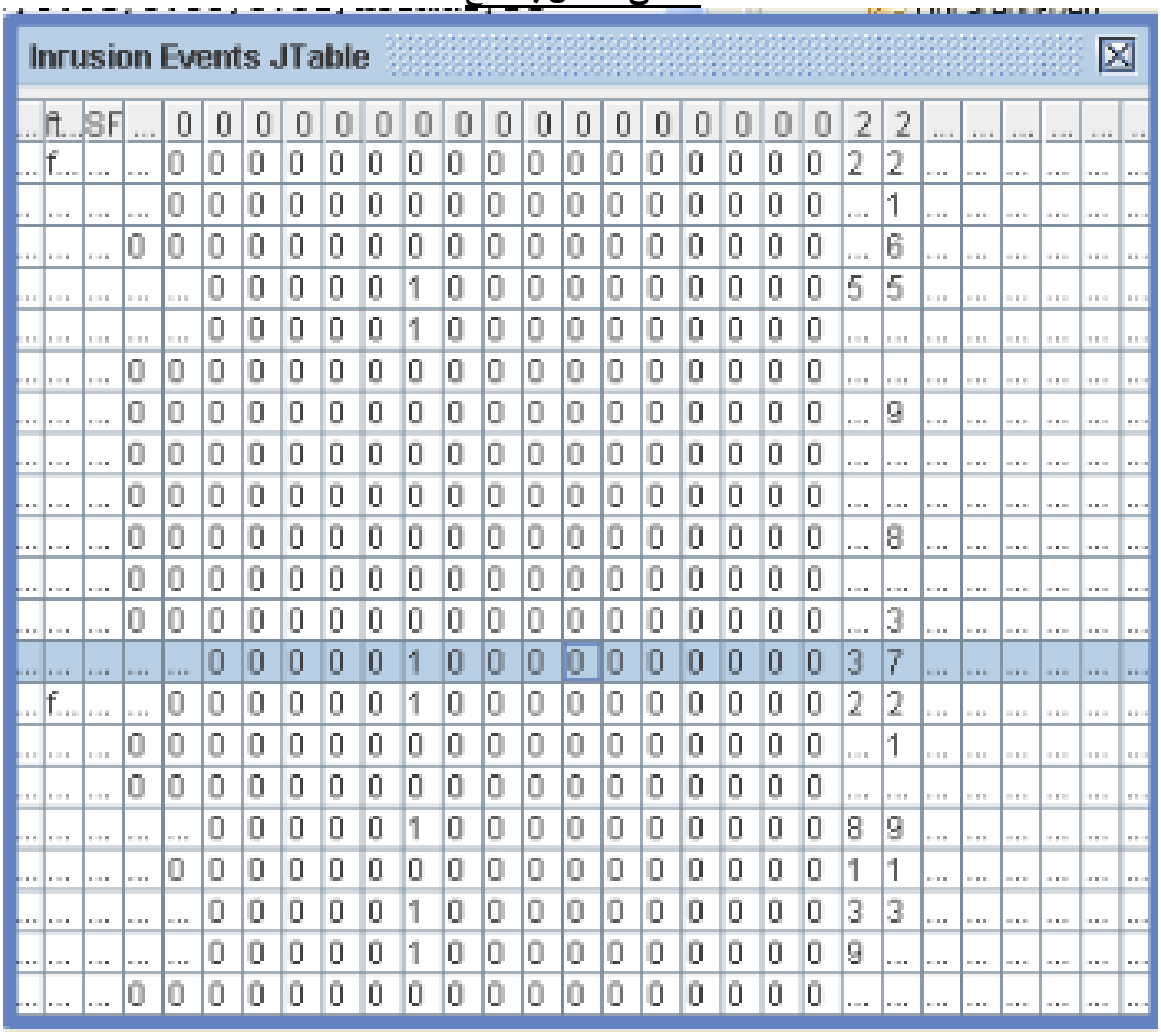
الحالة الثانية في حال عدم التطابق وعدم وجود الحرف في النمط، السطر التالي يقارن g من النمط مع " " مسافة من text، وبما أنهم لا يتطابقان والحرف " " لا يوجد في النمط سوف يحرك النمط بعدد حروف النمط وهي 4 في هذه الحالة [19].

يمكن النظر للشكل(7) يبين طريقة البحث في خوارزمية BM وهي تبين كيف يتم البحث بشكل أوضح وأفضل:

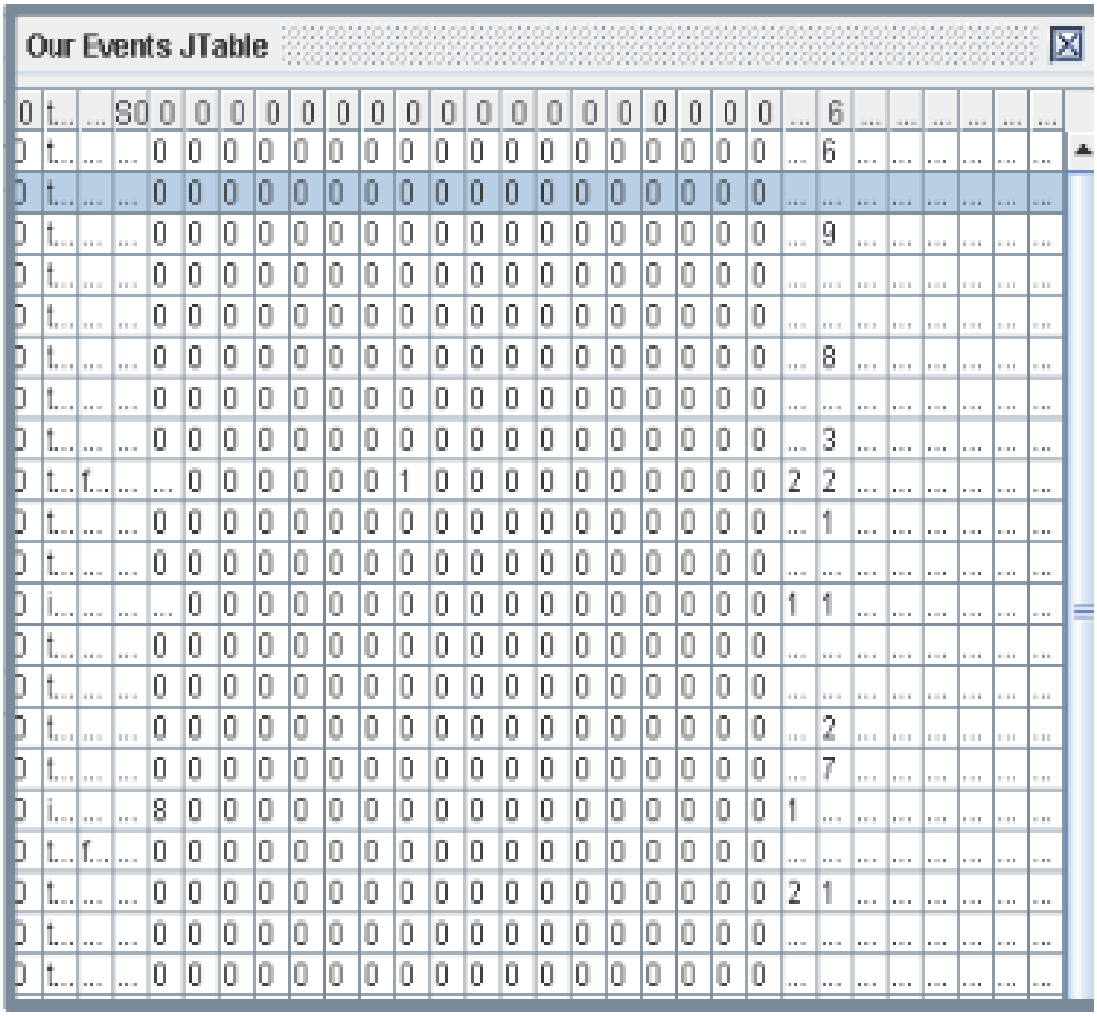


للشكل(7) طريقة البحث في خوارزمية BM

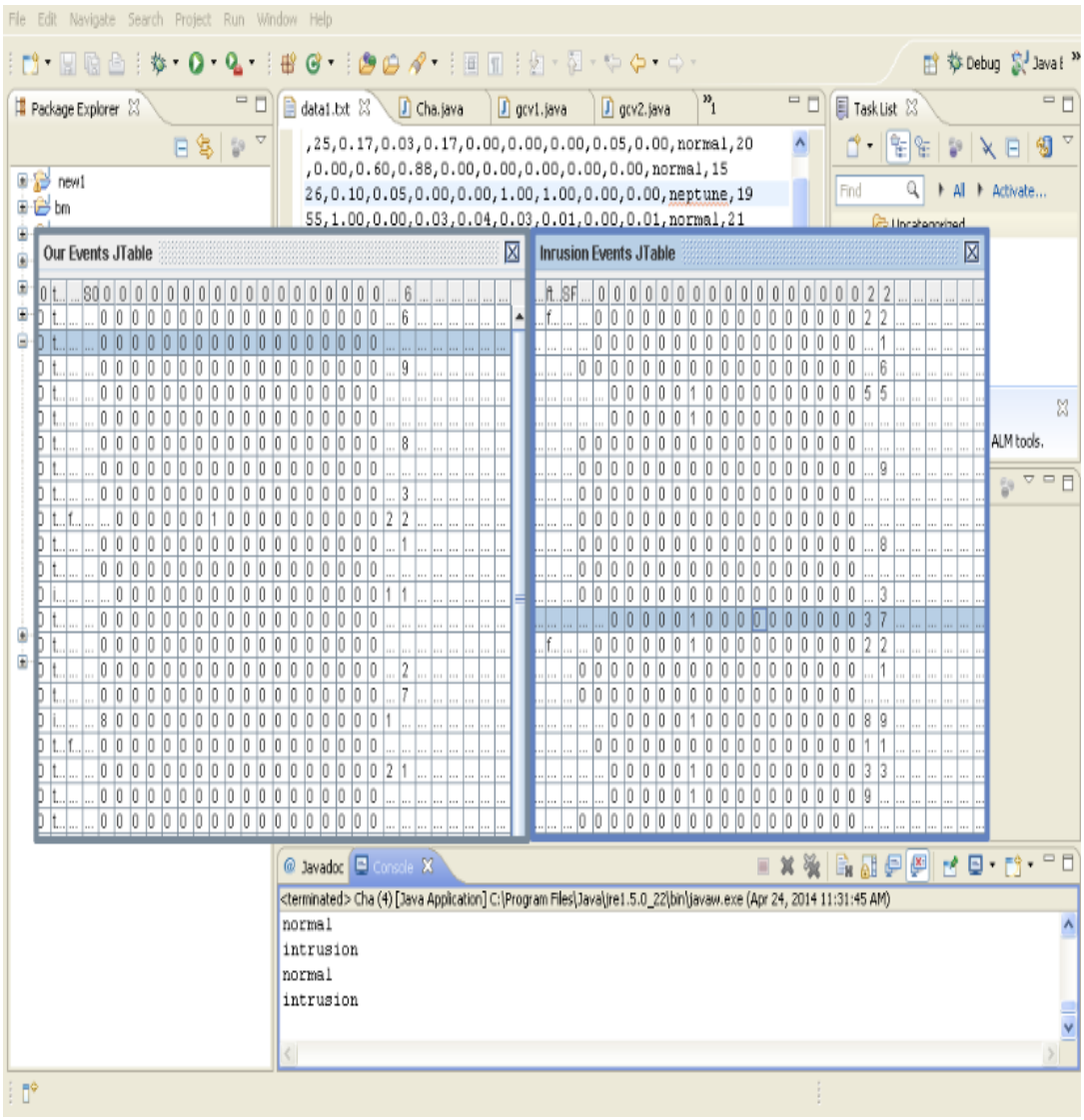
الملحق الخاص بالنتائج



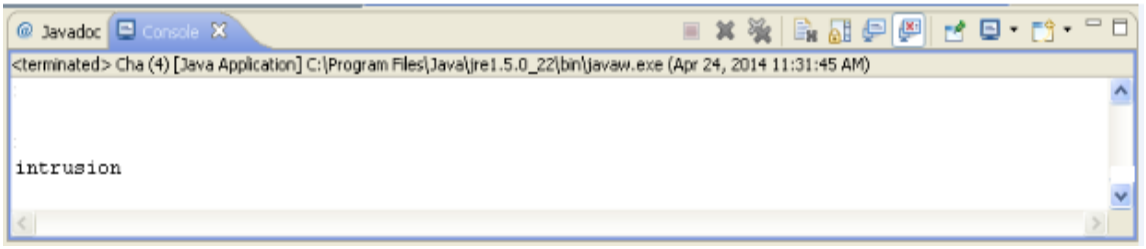
الشكل (1) جدول يحتوي على أحداث تطفلية



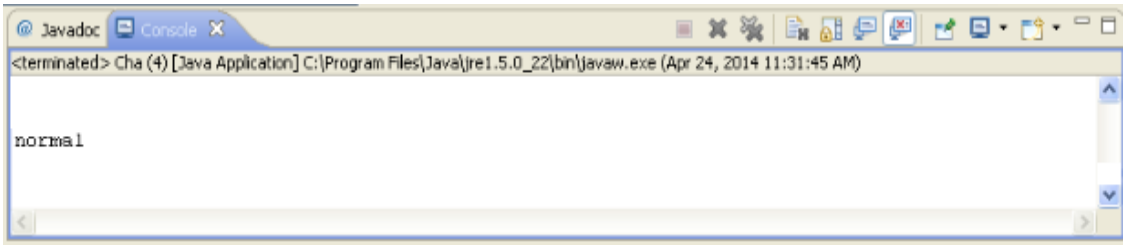
الشكل (2) جدول يحتوي على احداث تطفلية واعتيادية



الشكل (3) عملية مقارنة الجدولين



الشكل (4) نتيجة المقارنة المطابقة



الشكل (5) نتيجة المقارنة الغير المطابقة