Designing a Monitoring Tool for Mosul University Network to Detect Worm Infected Computers

AsmaaYaseen Hamoo *asmahamo@uomosul.edu.iq*
Sundus Abdulmuttalib Muhamed *sundus_abid7@uomosul.edu.iq*
College of Computer Science and Mathematics
University of Mosul, Mosul, Iraq

Received on: 05/06/2011 **Accepted on: 16/08/2011**

ABSTRACT

In order to make use of the web services, it has recently become inevitable to connect computers to the internet. This connection, however, make the computers prone to the challenges of intrusion and hacking.

The present study tackles the problem of computers' vulnerability to malware such as worm: a self-replicate computer program that spontaneously copies itself to the vulnerable systems and spreads through the web exploiting security gaps and posing a great danger to the web community.

The study resorts to the design and implementation of a fast scanning worm detection tool. The tool depends on counting failed connection attempts after study of the indicators of failed connection.

The tool performance is examined offline by using the real traffic for inbound and outbound packets of the network of the university of Mosul. After examining the net, we used the core switch to monitor the university's inbound and outbound traffic, where the collecting process of data took place on different periods to show the public layout of the net. The study comes up to the conclusion that the monitoring tool was capable of detecting the infected computers which performs anomalous behavior and allocating worm propagation periods (the growth phase of worm) accurately.

The tool is implemented by using the sixth version of java. It is applied under the Microsoft windows operating system environment and the protocol suites known as TCP/IP.

Keywords: Malware, Worms, Worm Detection

تصميم أداة مراقبة لشبكة جامعة الموصل لاكتشاف الحاسبات المصابة بالدودة الالكترونية

أسماء ياسين حمو **سندس عبد المطلب محمد**
كلية علوم الحاسوب والرياضيات، جامعة الموصل

تاريخ استلام البحث: 2011/06/05 **تاريخ قبول البحث: 2011/08/16**

الملخص

في الآونة الأخيرة أصبح ارتباط الحاسبات بشبكة الانترنيت أمراً لابد منه للاستفادة من الخدمات التي تقدمها هذه الشبكة. غير أن هذا الاتصال جعلها بمواجهات مع تحديات الاختراق والتطفل.

وتكمن مشكلة البحث في إحدى هذه التحديات الموجودة على هيئة البرنامج الخبيث المتمثل بالدودة وهو برنامج حاسوبي ذاتي الانتشار ينسخ نفسه تلقائياً إلى النظم غير المحصنة وينتشر عبر الانترنيت. تنتشر الدودة من خلال استغلال الثغرات الأمنية في الخدمات على الشبكة. وتشكل دودة الانترنيت خطراً كبيراً على مجتمع الشبكة.

وتم في هذا البحث تصميم وتنفيذ أداة للكشف عن الديدان – التي تعتمد على عملية المسح السريعة في انتقاء أهدافها – والمعتمدة على حساب عدد محاولات الاتصال الفاشلة وذلك بعد دراسة المؤشرات الدالة على فشل الاتصال.

تم اختبار أداء الأداة دون الاتصال المباشر بالشبكة وباستخدام سيل بيانات حقيقي للحزم الصادرة من والى شبكة جامعة الموصل. وبعد دراسة هذه الشبكة تم اختيار المبدل الرئيس لمراقبة سيل البيانات الصادر والوارد من والى

شبكة الجامعة حيث تمت عملية تجميع البيانات بفترات مختلفة لعكس الوضع العام للشبكة وقد تبين بان أداة المراقبة قادرة على اكتشاف الحاسبات التي تقوم بنشاط شاذ وتعيين فترات انتشار الدودة (مرحلة النمو للدودة) بدقة.

تم تمثيل الأداة باستخدام لغة جافا JAVA الإصدار السادس، وتنفيذها تحت بيئة نظم التشغيل مايكروسوفت ويندوز Microsoft Windows، وطقم البروتوكولات المعروفة بـ TCP/IP.

الكلمات المفتاحية: البرامج الخبيثة، الديدان، الكشف عن الدودة.

1. مقدمة Introduction

منذ اخترع الإنسان الحاسوب الشخصي في أواخر سبعينيات القرن العشرين، واستخدامات هذا الجهاز تزيد وتتنامى بشكل لم يسبق له مثيل، ثم جاءت شبكة الإنترنت لتجعل امتلاك كل شخص لحاسوب مرتبط بها أمراً ضرورياً للحياة العصرية. وبمرور الوقت أصبح اعتماد الإنسان على هذه التقنيات الجديدة كبيراً، فقد أصبحت ضرورية لتسيير الأعمال على جميع المستويات سواء للحكومات أو الشركات وحتى الأفراد، وأصبح تطور أية شركة وربما بقاؤها مرهوناً بوجودها على الشبكة العالمية، كما أصبح الشخص الذي لا يستخدم هذه التقنيات يوصم بعدم مواكبة العصر [1].

غير أن هذه التقنيات تشوبها بعض العيوب ، فأي جهاز حاسوب مرتبط بالانترنت يكون عرضة لخطر الإصابة بالبرامج الخبيثة Malicious Software (Malware) مثل الفيروسات، الديدان، أحصنة طروادة وغيرها من البرامج الخبيثة، فضلاً عن هجمات القراصنة Hackers [1].

الدودة برنامج حاسوبي ذاتي الانتشار ينسخ نفسه تلقائياً إلى النظم غير المحصنة وينتشر عبر الانترنيت. تشبه فيروس الحاسوب، ولكنها تختلف عن الفيروس الذي يلتصق أو يصبح جزءاً من برنامج آخر قابل للتنفيذ في أنها قائمة بحد ذاتها ولا تحتاج إلى أن تكون جزءاً من برنامج آخر لنشر نفسها. وبينما يلتصق الفيروس ببرنامج المضيف تنتشر الدودة من خلال استغلال الثغرات الأمنية في الخدمات على الشبكة [12]. وتشكل دودة الانترنيت خطراً كبيراً على مجتمع الشبكة [16].

2. الدراسات السابقة Literature Survey

1. أنشأ كيم Kim و كارب Karp [11] نظام التوقيع الذاتي الاوتوغراف Outograph وهو نظام كشف عن توقيع الدودة الموزع قادر على التعامل مع الديدان المتحولة Metamorphic والديدان متعددة الأشكال Polymorphic. يعتمد نظام التوقيع الذاتي على عمليات المسح غير الناجحة لتحديد عنوان الانترنيت IP المشتبه بإصابته وعزل البيانات المتدفقة بوساطة منفذ الوجهة. كما يولد تلقائياً توقيعات لديدان TCP من خلال تحليل محتوى الشفرة بالاستناد إلى تسلسل الكتل الثُّمانية الأكثر حدوثاً في البيانات المتدفقة المشبوهة.
2. صمم تشين وتانغ [5] معمارية مكافحة الديدان الموزعة Distributed Anti-Worm (DAW) لمجهزي خدمة الانترنيت ISPs لتقديم خدمة مكافحة الديدان لزبائنها. تشخص هذه المعمارية مصادر المسح وذلك بتعقب كل من حزم TCP SYN و TCP RST أي معالجة ديدان TCP فقط .
3. وضع تشيتشير وزملاؤه [10] خوارزمية قائمة على خوارزمية فحص فرضية التتابع العكسي Reverse Sequential Hypothesis Testing Algorithm لمراقبة فشل الاتصال وخوارزمية معدل الاتصال القائم على الثقة المحدود Credit Based Connection Rate Limiting (CBCRL) لتحديد المعدل الذي يمكن أن يبتدئ فيه المضيف بالاتصالات الأولى على الشبكة.

4. وضع وو وزملائه [19] خوارزمية أساسها عدد الضحايا. وتعرف هذه الخوارزمية الضحية "بالعنوان IP الذي ترسل منه حزمة إلى أحد العناوين غير الفعالة". وهذا يعني أن عنوان الانترنيت IP المصدر– عند إرسال عنوان الانترنيت IP لحزمة إلى عنوان انترنيت IP غير مستخدم – يعد ضحية. ولمنع إطلاق إنذارات كاذبة جمعَ مصممو الخوارزمية بين هذا التعريف مع قاعدة قرار المسح الثنائي Two Scan Decision Rule (TSDR) للإشارة– في حالة اكتشاف النظام لحزمتين مُرسلَتين إلى عناوين انترنيت IP غير مستخدمة من نفس المضيف– إلى أن هذا المضيف ضحية. وعندما يصل معداد الضحية إلى قيمة حد عتبة معينة، يولد النظام إنذاراً عن الديدان.

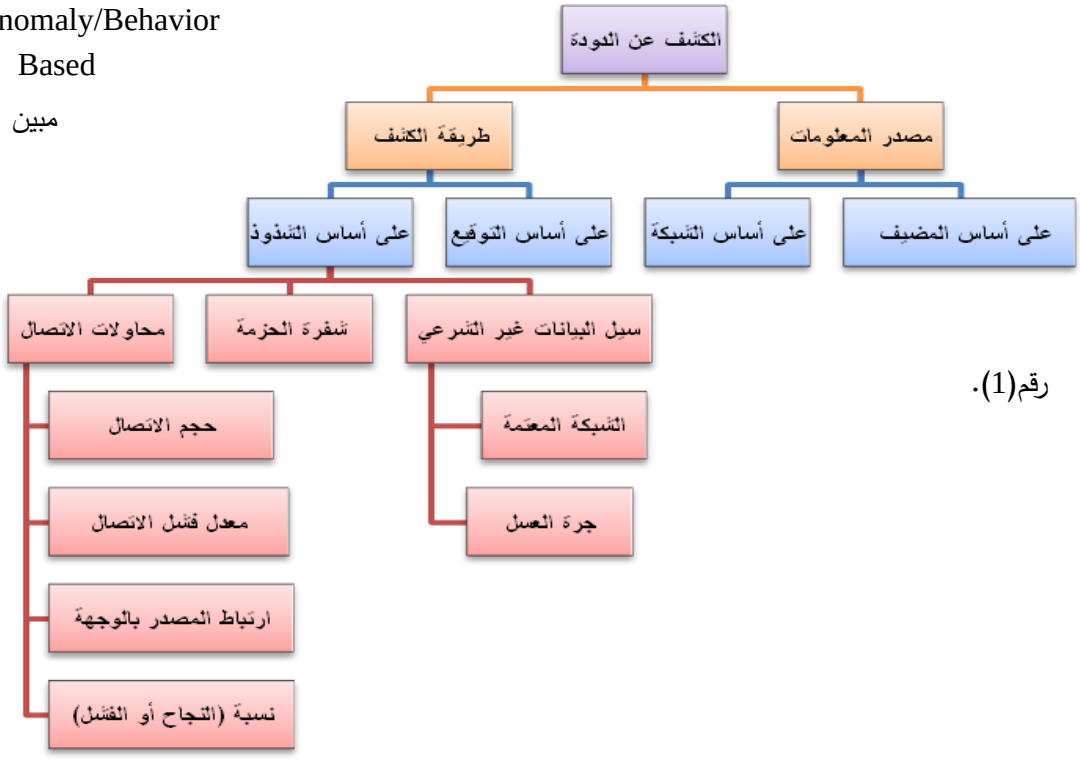
3. أهداف البحث Thesis Objectives

يهدف هذا البحث إلى :

- دراسة طرائق الكشف عن الديدان الحاسوبية وفوائد ومساوئ كل طريقة.
- تصميم أداة مراقبة لشبكة جامعة الموصل لغرض الكشف عن الأجهزة الحاسوبية المحتمل إصابتها بالدودة الحاسوبية والتي تعتمد على عملية المسح السريعة في انتقاء أهدافها.
- تبني طريقة كشف عن الدودة الحاسوبية تعطي نتائج دقيقة وبسرعة قدر الإمكان.

4. الكشف عن الدودة Worm Detection

هو أحد أنواع تقنيات الكشف عن التطفل التي تطورت خلال هذه السنوات وهنالك طريقتان رئيستان لتصنيف نظم الكشف عن الدودة. التصنيف الأول يعتمد على مصدر المعلومات Information Source ويضم نوعين أساسيين من نظم الكشف عن الديدان هما : الأول على أساس المضيف Host-Based والثاني على أساس الشبكة Network-Based . أما التصنيف الثاني فيعتمد على طريقة الكشف Detection Method. ويندرج تحت هذا التصنيف نوعان هما : نوع على أساس التوقيع Signature-Based ونوع على أساس السلوك/الشذوذ Anomaly/Behavior Based وكما مبين في الشكل رقم(1).

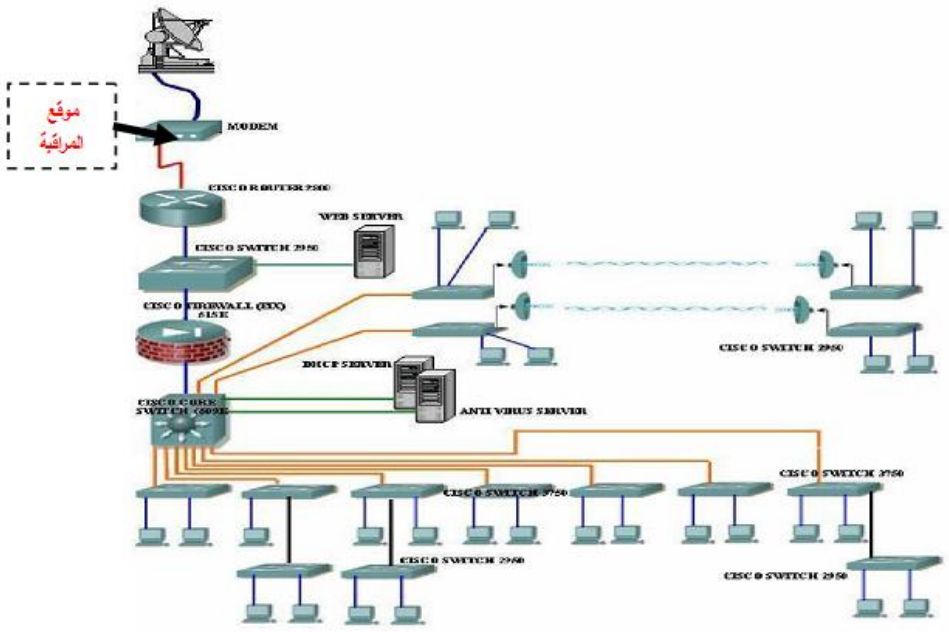


الشكل رقم (1) تصنيف الديدان الحاسوبية
(من تصميم الباحث)

5. التوصيف الطبولوجي لشبكة جامعة الموصل

تأسست شبكة جامعة الموصل سنة 2004، وكان الغرض من الشبكة هو لربط المواقع المختلفة للجامعة بوصلات عالية السرعة (واحد كيكابت ايثرنت). تقدم الشبكة خدمات عديدة لزبائنها البالغ عددهم 2000 مستخدم مثل المشاركة عبر الانترنت Internet Sharing، البريد الالكتروني Email Accounts، استضافة مواقع الويب Web Hosting والمحادثة الداخلية Internal Chatting. وقد يشهد المستقبل زيادة في تطبيقات الشبكة لتشمل حقولاً أكثر تطوراً مثل المشاركة في قواعد البيانات Database Sharing وتطبيقات الوسائط المتعددة التفاعلية Interactive Multimedia Applications . يبين الشكل رقم (2) طبولوجية التركيب الأساس لشبكة جامعة الموصل [4].

تتألف شبكة جامعة الموصل من 41 مبدلاً switch من نوع سيسكو 3750 و2950 & Cisco 3750 2950 ترتبط عن طريق واحد كيكابت بالثانية ايثرنت بالمبدل الرئيسCisco 6051E core switch، وتمثل أقسام الجامعة المختلفة. كل مبدل يرتبط بالعديد من المبدلات من الطبقة الثانية layer2 switches ومضيفات القسم المختلفة. يتم الاتصال بالانترنت من خلال جهاز خادم يعمل كجدار ناري وموجه ترجمة عنوان الشبكة (NAT) من نوع سيسكو2800 Cisco2800 . ويمكن الوصول إلى خدمة الانترنت أيضاً من خلال الاتصالات المحلية اللاسلكية IEEE802.11b WLAN العديدة [4]. يبين الجدول رقم (1) وصفاً لأجهزة الشبكة المختلفة وتهيئتها الحالية.



الشكل رقم (2). طبولوجية التركيب الأساس لشبكة جامعة الموصل

يبدأ عنوان الانترنت IP لشبكة جامعة الموصل بالأرقام (172.20) وهو عبارة عن عنوان خاص private وباستخدام تقنية ترجمة عنوان الشبكة (NAT) يتم تحويله إلى عنوان عام public IP حيث أن جميع أجهزة الشبكة تكون مشتركة بهذا العنوان العام. من الواضح أن مفاهيم أمن الشبكة لم تؤخذ بنظر الاعتبار في أثناء تركيب الشبكة، لذا قمنا بتصميم أداة مراقبة للكشف عن الحاسبات المحتمل إصابتها بالدودة الحاسوبية داخل شبكة الجامعة .

الجدول رقم (1) الوضع الحالي لأجهزة شبكة جامعة الموصل[4]

اسم الجهاز	الكمية	الوصف	التهيئة الحالية
Cisco Router 2800	1	• 2 Fast Ethernet ports, 2 serial ports • IOS=12.3 • Support=Rip1,Rip2,EIGRP,IGRP,OSPF,ISIS,BGP,VPN,VLA,VTP	• Dynamic Routing= IGRP • Static Routing = default state • Extended access list • Static NAT • No Encrypted Password
Cisco Switch 2950	30	• Layer 2 switch • 24 Fast Ethernet ports • IOS=12.3 • VLAN, VTP	• 1VLAN/Switch • Default configuration • No Encrypted Password
Cisco Core switch 6051E	1	• Layer 3 switch • 3modules (fiber optic , Gigabit Ethernet, Fast Ethernet, 48 port) • IOS=12.4	• (50) Port Based VLANS • Inter VLAN Enabled • Extended access-list • Default configuration • Encrypted Password
Cisco switch	11	• 24 Fast Ethernet ports • 2 Gigabit Ethernet Ports	• 1VLAN/Switch • Default configuration

اسم الجهاز	الكمية	الوصف	التهيئة الحالية
3750 layer2 switch		• IOS=12.4	• No Encrypted Password
Antivirus Server	1	• DELL POWER EDGE 6600 SERVER	• Password Required
Access point	1	• AP 1200	• 64 bit WEP • MAC Address Filtering

6. الخوارزمية المعتمدة

تم اعتماد طريقة الكشف المبنية على حساب عدد محاولات الاتصال الفاشلة Failed Connection Attempts وأساس هذه الطريقة هو ملاحظة أن الأجهزة المصابة بالدودة تحاول الاتصال بالعديد من المضيفات التي لا يمكن الوصول إليها. ومن خلال حساب عدد محاولات الاتصال الفاشلة هذه خلال فترة زمنية محددة ومقارنته بقيمة حد عتبة threshold محددة مسبقاً سيتم الكشف عن الأجهزة المحتمل إصابتها بالدودة.

6-1 المؤشرات الدالة على محاولة الاتصال الفاشل Failed connection indicators

هنالك مؤشرات عديدة للدلالة على فشل الاتصال تعتمد على البروتوكول وهي كالآتي :

- فبالنسبة لبروتوكول TCP هناك ثلاثة مؤشرات للدلالة على فشل محاولة الاتصال TCP SYN packet وهي [7]:

أ- **حزمة ICMP لا يمكن بلوغ الوجهة Destination Unreachable مختصراً ICMP Type 3**: وتحدث في حالة الاتصال بعنوان وجهة غير مستخدم أو كانت الشبكة الوجهة معطلة، حيث تقوم الموجهات بالرد بحزمة ICMP Type3 أو قد يكتفي البعض بإهمال حزمة طلب الاتصالTCP SYN الفاشلة فقط ومن دون استجابة.

ب- **حزمة TCP RST** : وتحدث في حالة الاتصال بعنوان وجهة مستخدم ولكن رقم منفذ الوجهة يعود لخدمة غير متوفرة، حيث يقوم المضيف الوجهة بالرد بحزمة TCP RST أو الاكتفاء بإهمال حزمة طلب الاتصال TCP SYN الفاشلة فقط ومن دون استجابة.

ت- **No Answer** : في حالة عدم وصول إشعار بالاستلام متمثل بحزمة TCP SYN/ACK أو أي من الحزمتين المذكورتين في المؤشرين السابقين للدلالة على فشل الاتصال وضمن فترة زمنية محددة سابقاً predefined timeout فيمكن اعتبار هذه الحالة أيضاً مؤشراً على فشل الاتصال .

- أما بالنسبة لبروتوكول UDP، وكما هو الحال في حالة بروتوكول TCP فعند الاتصال بعنوان وجهة غير مستخدم أو كانت الشبكة الوجهة معطلة، فسيكون الرد إما بحزمة ICMP لا يمكن بلوغ الوجهة أو الاكتفاء بإهمال حزمة طلب الاتصال الفاشلة فقط وبدون أي رد.

وأما إن كان عنوان الوجهة مستخدم ولكن رقم منفذ الوجهة يعود لخدمة غير متوفرة، فسيقوم المضيف الوجهة بالرد بحزمة ICMP لا يمكن بلوغ المنفذ الوجهة وليس بحزمة TCP RST كما هو الحال في بروتوكول TCP.

وبما أن بروتوكول UDP هو بروتوكول عديم الاتصال ولا تحتاج جميع بروتوكولات طبقة التطبيق المعتمدة على بروتوكول UDP في طبقة النقل إلى استلام رد من المضيف الوجهة، بالتالي من الصعب جداً الكشف عن محاولة الاتصال الفاشلة عند إهمالها [7].

في عملنا قمنا فقط بتعقب حزمة ICMP Type 3 أما حالة عدم استلام الرد فيمكن اعتبار معالجتها من ضمن الأعمال المستقبلية .

- وأما في حالة بروتوكول ICMP، ففي الماضي تم استخدام حزمة طلب الصدى ICMP Echo للتحقق من وجود الأهداف ولهذا السبب سنقوم بتعقب هذا النوع من المسح أيضاً. هناك مؤشران للدلالة على فشل الاتصال في حالة بروتوكول ICMP هما [7] :

أ- **حزمة ICMP لا يمكن بلوغ الوجهة Destination Unreachable** : وذلك في حالة الاتصال بعنوان وجهة غير مستخدم أو كانت الشبكة الوجهة معطلة كما ذكرنا آنفاً.

ب- **No Answer** : عند إرسال حزمة طلب الصدى ولم يتم استلام حزمة رد الصدىICMP echo reply أو حزمة ICMP لا يمكن بلوغ الوجهة وضمن فترة زمنية محددة مسبقاً time out فعندئذٍ يمكن اعتبار هذه الحالة مؤشراً آخر على فشل الاتصال.

6-2 الافتراضات المعتمدة Assumptions

في هذا البحث تم الاعتماد على الافتراضات الآتية :

الافتراض الأول : حساب عدد حزم الاتصال الفاشلة الأولى First Failed Connection Packets كحزمة ICMP لا يمكن بلوغ الوجهة وحزمة TCP RST الواردة من عنوان وجهة خارجي إلى عنوان مصدر داخلي [7][20].

الافتراض الثاني : تقوم الأداة بمراقبة سيل بيانات الشبكة الصادرة والواردة من والى المبدل الرئيس core switch مابين الشبكة الداخلية والانترنيت ، وفحص الأجهزة المضيفة المحلية من الإصابة ، ولم يتم فحص الحاسبات البعيدة.

الافتراض الثالث *:* تسجيل طلبات الاتصال الأولى(FCC) First Contact Connection Requests والاستجابة على هذه الطلبات للإشارة إلى نجاح الاتصال. حزمة طلب الاتصال الأولى هي عبارة عن حزمة TCP SYN أو ICMP Echo معنونة إلى المضيف الذي لم يتصل المرسل به مسبقاً [9][10].

6-3 ملخص خوارزمية الكشف

وفي ما يأتي ملخص خوارزمية الكشف :

المدخلات : حزم طلب الاتصال من نوع TCP/SYN و ICMP Echo الصادرة من أجهزة الحاسوب المحلية في شبكة المراقبة، حزمة TCP RST، حزمة ICMP Type3 لا يمكن بلوغ الوجهة، حزمة TCP SYN/ACK ، حزمة ICMP Echo Reply وحزمة تخميد المصدر ICMP Type 4 الواردة إلى أجهزة الحاسوب المحلية في شبكة المراقبة، بحيث يتم تمثيل كل حزمة من هذه الحزم على شكل قيد Record يتألف من الحقول الآتية :

- **src_IP**: عنوان المصدرSource IP للاتصال.
- **protocol** : رقم البروتوكول (TCP=6 ، UDP=17 ، ICMP=1).
- **dst_port** : رقم منفذ الوجهة للاتصال (في حالة بروتوكولي TCP و UDP).
- **timestamp** : الختم الزمني.
- **dst_ip**: عنوان الوجهة Destination IP للاتصال.
- **status**: يعبر عن حالة الاتصال بعنوان الوجهة dst_ip، ويأخذ قيمة حرفية (char) حيث:
  - الحرف p يعني pending أي طلب اتصال ولم يتم الرد عليه لحد الآن.

- الحرف f يعني failure أي فشل الاتصال.

- **failed_Connectioncount** : معداد لحساب عدد حزم الاتصال الفاشلة الأولى First Failed Connection Packets لأجهزة الحاسوب المحلية في شبكة المراقبة.
- **firstFailed_timestamp** : زمن حدوث أول محاولة اتصال فاشلة للعنوان المصدر src_IP (يمثل الختم الزمني **timestamp** لأول حزمة اتصال فاشلة).

المخرجات : معلومات عن أجهزة الحاسوب المحلية المحتمل إصابتها بالدودة الالكترونية.

وعند استلام حزمة اتصال فاشلة أولى مرسلة إلى حاسبة محلية، خطوات المعالجة هي :

الخطوة الأولى : استخراج المعلومات من الحزمة الحالية اللازمة لتكوين القيد.

الخطوة الثانية : جعل حالة عنوان الوجهة (حقل status) تساوي الحرف f.

الخطوة الثالثة : إضافة واحد إلى معداد حزم الاتصال الفاشلة failed_Connectioncount.

الخطوة الرابعة : مقارنة قيمة failed_Connectioncount بقيمة حد العتبة threshold ولتكن x، فإذا تجاوزت القيمة x وكانت الفترة الزمنية مابين الـ timestamp للحزمة الحالية والـ firstFailed_timestamp أقل أو تساوي فترة زمنية محددة مسبقاً ولتكن T مثلاً فسيتم إعطاء إنذار عن إصابة محتملة للحاسبة المحلية src_IP بالدودة.

4-6 اللغة البرمجية المستخدمة

تم استخدام لغة جافا الإصدار السادس JAVA 2.6 في تمثيل خوارزمية الكشف المعتمدة برمجياً. لغة جافا إحدى اللغات البرمجية إنتاج شركة صن مايكروسيستم Sun Microsystem وهي لغة مفتوحة المصدر Open Source وكيانية التوجه Full Object Oriented وتمتاز بمرونتها وسهولة التعامل معها واستقلاليتها عن بيئة التشغيل Independent Platform [3]. كما تمتاز لغة جافا بالقوة Robust والسرية Secure ومتعددة المسالك Multithreading [17]. توفر لغة جافا العديد من الحزم packages والأصناف classes التي تم الاستفادة منها في تمثيل خوارزمية الكشف المعتمدة ومنها :

- ❖ حزمة jpcap : توفر هذه الحزمة العديد من الأصناف classes التي تعمل على تحليل وإرسال وخزن حزم البيانات وتعريف كائنات objects مقابلة لحزم الشبكة المتنوعة [6].
- ❖ القائمة الموصولة Linked List : عبارة عن صنف واقع ضمن حزمة *.java.util وهي عبارة عن هيكل بياني مكون من مجموعة من العناصر (عقد nodes) مرتبطة مع بعضها البعض [17].
- ❖ الخارطة الهاشية Hash Map : هي عبارة عن صنف واقع ضمن حزمة *.java.util وهو عبارة عن هيكل بياني مكون من مجموع من العناصر على شكل أزواج كل عنصر ممثل بمفتاح والقيمة المقابلة له (key/value) حيث يقوم بعمل Mapping مابين المفتاح key والقيمة value المقابلة له من خلال تحويل المفتاح باستخدام الدالة الهاشية Hash Function إلى قيمة تمثل مؤشر index للقيمة المقابلة له value. وتعد الخارطة الهاشية من أهم وأسرع هياكل البيانات على الإطلاق حيث زمن التنفيذ لها يساوي O(1). تضمن لنا هذه البنية الوصول السريع إلى البيانات مهما كان حجم تلك البيانات، إضافةً إلى أن إدخال البيانات أيضاً يتم بسرعة عالية [8].

❖ المسالك المتعددة Multithreading : تم استخدام تقنية المسالك المتعددة Multithreading التي توفرها لغة جافا في برمجة أداة المراقبة بالإضافة إلى مفهوم التزامن Synchronization الذي يمكن المسالك التنفيذية threads من الوصول إلى المصادر المشتركة Shared Resources بشكل متزامن[13].

في عملنا تم خزن عناوين الحاسبات الفعالة داخل شبكة الجامعة أثناء فترة جمع البيانات على شكل مفاتيح للخارطة الهاشية والقيمة المقابلة لكل مفتاح يمثل عنوان القائمة الموصولة التي بدورها تقوم بخزن القيود records وكل قيد يتم تمثيله على شكل صنف class، ويتم التفريق بين القيود الواقعة ضمن القائمة الواحدة بالاعتماد على قيمة حقل البروتوكول وحقل منفذ الوجهة في حالة TCP و UDP.

7. تجميع البيانات Data collection

تم اختبار أداة المراقبة بصورة غير فعالة offline وباستخدام سيل بيانات حقيقي. ولغرض الحصول على سيل البيانات تم استخدام برنامج Windump[18] اصدار ويندوز لبرنامج Tcpdump الذي يستخدم مع نظام لينكس. تم تنصيب برنامج windump على حاسبة شخصية pc ذات مواصفات (1.73GHz clocked cpu 1G DDRAM2, Pentium4,) والتي تم تنصيبها على المبدل الرئيس لشبكة جامعة الموصل وعلى المنفذ SPAN(Switched Port Analyzer) الذي تمر منه جميع الحزم الصادرة والواردة من والى شبكة الجامعة، حيث يقوم برنامج Windump بالتقاط ونسخ الحزم المارة بهذا المنفذ ومن ثم خزنها بصيغة ثنائية في ملف خاص سيتم إدخاله فيما بعد إلى أداة المراقبة لتحليل الحزم المخزونة فيه وإعطاء النتائج .

لقد تمت عملية تجميع البيانات على ثلاث مراحل، الفترة الزمنية لكل مرحلة تساوي يوم واحد، ويبين الجدول رقم (2) معلومات عن ملفات سيل البيانات الملتقطة باستخدام برنامج Windump لكل مرحلة من مراحل تجميع البيانات الثلاث، حيث تم اختيار مراحل تجميع البيانات بفترات زمنية متباينة لتعكس الوضع العام لفعالية شبكة جامعة الموصل.

الجدول رقم (2) معلومات عن ملفات سيل البيانات الملتقطة باستخدام برنامج Windump

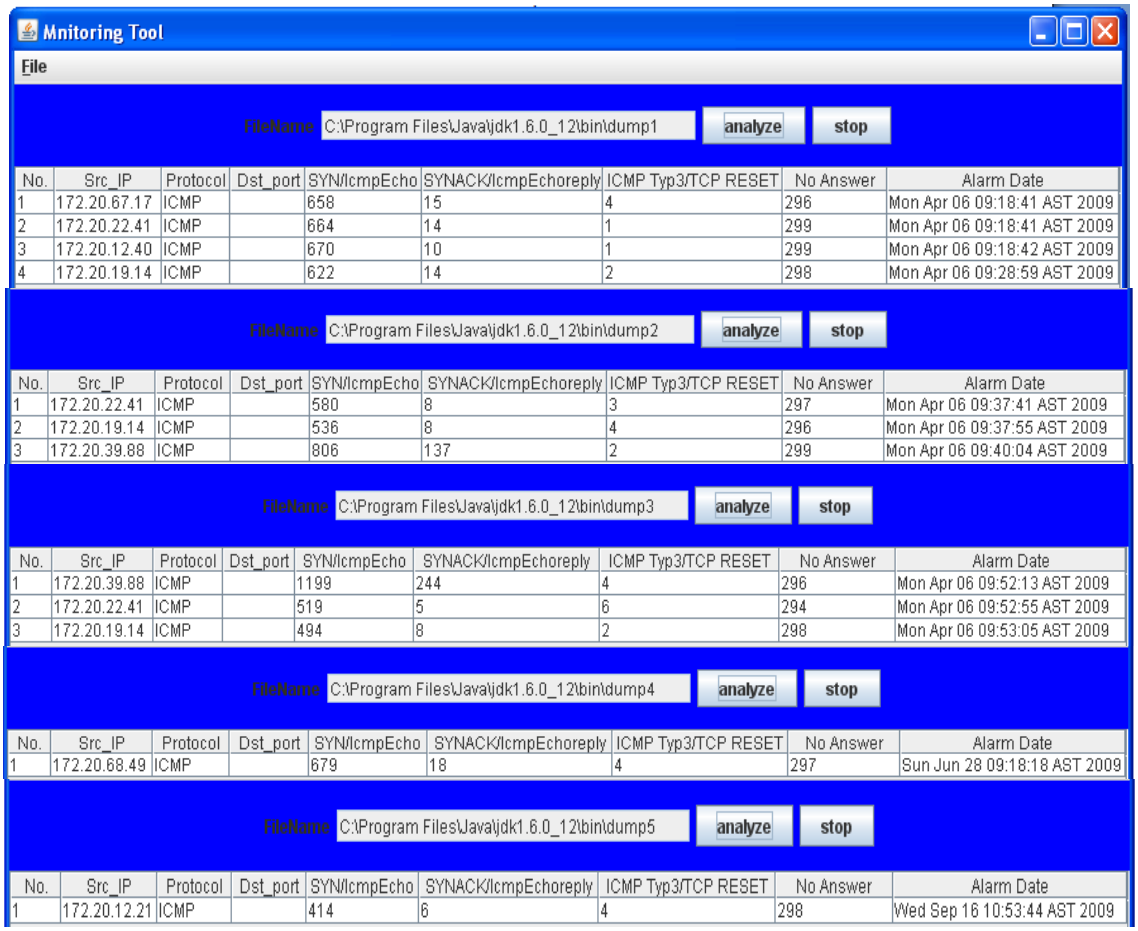
المرحلة	اسم الملف	الفترة الزمنية h:min:sec	التأريخ	حجم الملف	عدد الحزم الكلي	عدد الحاسبات الفعالة
المرحلة الأولى	Dump1	09: 17: 24 09: 32: 17	2009/4/6	52. 5 MB	709549	528
	Dump2	09: 36: 22 09: 51: 19	2009/4/6	176 MB	2135790	557
	Dump3	09: 51: 41 10: 01: 29	2009/4/6	155 MB	1424137	530
المرحلة الثانية	Dump4	08: 06: 46 12: 01: 48	2009/6/28	1. 93 GB	23009808	1011
المرحلة الثالثة	Dump5	08: 27: 53 11: 53: 59	2009/9/16	1. 78 GB	20746340	904

8. النتائج

تم تنفيذ أداة المراقبة على سيل البيانات المخزون في الملفات المبينة في الجدول رقم (2) ولكل مرحلة من مراحل تجميع البيانات الثلاث، وباستخدام قيم مختلفة لحد العتبة لعدد المحاولات الفاشلة في كل تنفيذ. وتم اختيار

النتائج المبينة في الشكل رقم (3) كونها الأكثر دلالة على احتمالية الإصابة بديدان تقوم بعملية مسح عشوائي سريع، حيث تم التعويض عن :

- قيمة time out بـ 3 ثواني في حالة TCP كما معمول به في بحوث عديدة مثل [9] [10] [14] والمعتمدة على المصدر[15].
- قيمة time out بـ 4 ثواني في حالة ICMP مساوية لقيمة time out الافتراضية للأداة ping [2].
- قيمة حد العتبة لعدد محاولات الاتصال الفاشلة بالرقم (150) في حالة TCP وUDP وبالرقم (300) في حالة ICMP بالاعتماد على المصدر [7].
- الفترة الزمنية لحساب عدد المحاولات الفاشلة ومقارنته بقيمة حد العتبة بـ 5 دقائق بالاعتماد على المصدر [7].



الشكل رقم (3). واجهة تنفيذ أداة المراقبة للملف Dump5 ، Dump4 ، Dump3 ، Dump2 ، Dump1

عند الضغط على الزر analyze تبدأ عملية القراءة من ملف الحزم المختار وتحليلها لغرض الكشف عن الحاسبات المحتمل إصابتها بالدودة حيث ستظهر معلومات عنها على واجهة البرنامج على شكل جدول كما مبين في الشكل رقم (3) والذي يتألف من الحقول الآتية :

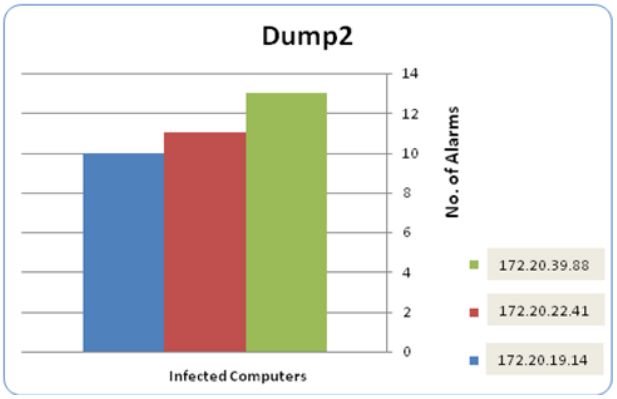
1. حقل No. : يمثل تسلسل الحاسبات المحتمل إصابتها بالدودة.
2. حقل Src_IP : يمثل عنوان IP الحاسبات المحتمل إصابتها بالدودة.

3. حقل Protocol : يمثل بروتوكول النقل TCP ، UDP ، ICMP.
4. حقل Dst_port : رقم منفذ الوجهة في حالة TCP و UDP.
5. حقل SYN/ICMPEcho : عدد طلبات الاتصال الأولى الصادرة من الحاسبة Src_IP (حزم TCPSYN في حالة بروتوكول TCP) أو (حزم ICMP Echo في حالة بروتوكول ICMP) والتي تمت قراءتها ومعالجتها وحسابها لحين اكتشاف الحاسبة Src_IP .
6. حقل SYNACK/IcmpEchoreply : عدد (حزم TCP SYN/ACK في حالة بروتوكول TCP) و (حزم ICMP Echo reply في حالة بروتوكول ICMP) الواردة إلى الحاسبة Src_IP والتي تمت قراءتها ومعالجتها وحسابها لحين اكتشاف الحاسبة Src_IP.
7. حقل ICMP Typ3/TCP RESET : عدد حزم ICMP لا يمكن بلوغ الوجهة وحزم TCP RESET الواردة إلى الحاسبة Src_IP والتي تمت قراءتها ومعالجتها وحسابها لحين اكتشاف الحاسبة Src_IP.
8. حقل No Answer : عدد طلبات الاتصال الأولى الصادرة من الحاسبة Src_IP (حزم TCP SYN في حالة بروتوكول TCP) أو (حزم ICMP Echo في حالة بروتوكول ICMP) والتي لم يتم الاستجابة أو الرد عليها ضمن فترة الـ timeout.
9. حقل Alarm Date : الختم الزمني Timestamp لحزمة الاتصال الفاشلة المسببة لحدوث الإنذار Alarm (ظهور معلومات عن الحاسبة Src_IP المحتمل إصابتها بالدودة).

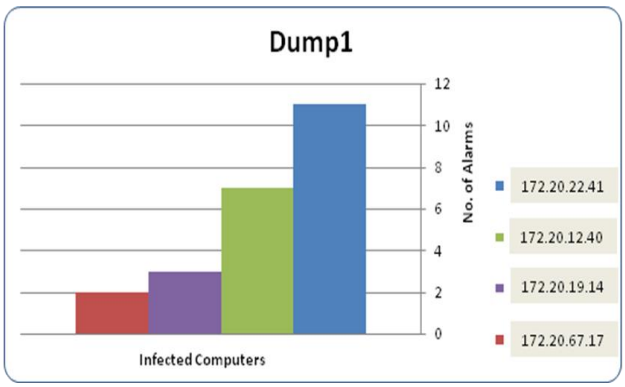
ملاحظة : قيمة حقل ICMP Typ3/TCP RESET + قيمة حقل No Answer = قيمة حد العتبة لعدد محاولات الاتصال الفاشلة (150) في حالة TCP وUDP و(300) في حالة ICMP.

النتائج المبينة في الشكل رقم (3) هي نتاج حساب إنذار واحد لكل حاسبة Src_IP المحتمل إصابتها بالدودة، ولغرض إعطاء نتائج أدق تم تنفيذ أداة المراقبة على سيل البيانات المخزون في الملفات المبينة في الجدول رقم (3) ولكل مرحلة من مراحل تجميع البيانات الثلاث ولكن مع حساب كافة الإنذارات المكررة والصادرة من الحاسبة Src_IP المحتمل إصابتها بالدودة علماً أن الإنذار الواحد وكما سبق ذكره يعني 150 محاولة اتصال فاشلة في حالة TCP و UDP و300 محاولة فاشلة في حالة ICMP ،كانت نتائج التنفيذ وبعد تحويلها إلى :

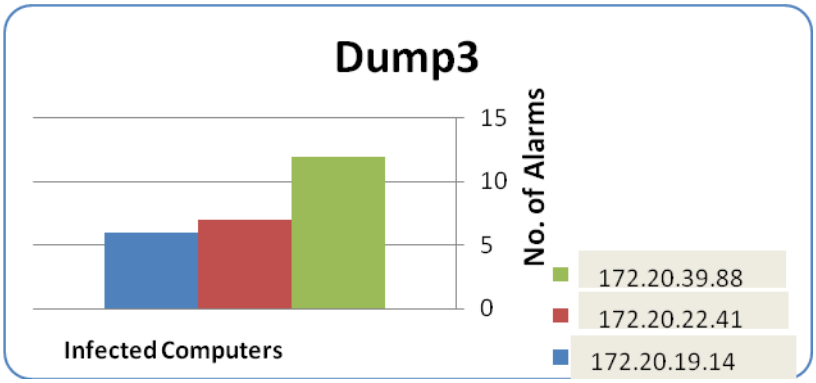
أ. **مخطط عمودي** يبين الحاسبات المكتشفة وعدد الإنذارات الصادرة منها، ظهرت لنا النتائج المبينة في الأشكال (4)، (5)، (6)، (7)، (8) وكالآتي:



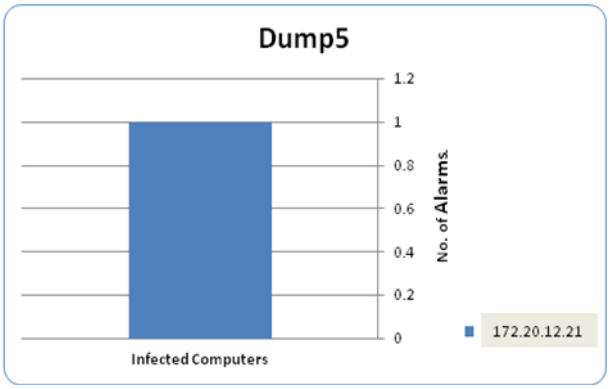
الشكل (4). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump1



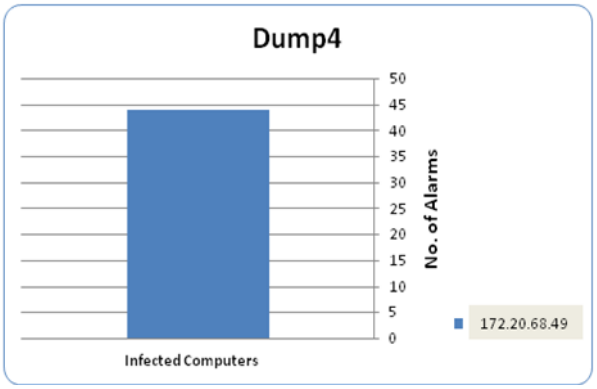
الشكل (5). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump2



الشكل (6). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump3



الشكل (7). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump4



الشكل (8). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump5

كما أشرنا سابقاً تتألف المرحلة الأولى من مراحل تجميع البيانات الثلاث من الملفاتDump1 و Dump2 و Dump3 التي تم الحصول عليها بفترات زمنية متقطعة ولكن ضمن نفس اليوم لذلك سنقوم بمناقشة النتائج المبينة في الأشكال (4)، (5)، (6) معاً، فمثلاً نلاحظ في الشكل (4) التابع للملف Dump1 أن عدد الإنذارات الصادرة من الحاسبة 172.20.22.41 يبلغ (11) إنذاراً واستمرار إطلاقها للإنذارات في Dump2 والبالغ عددها (11) إنذاراً وDump3 والبالغ عددها (7) إنذارات حيث يمثل هذا مؤشراً قوياً على احتمالية إصابة هذه الحاسبة بالدودة.

أما بالنسبة للحاسبة 172.20.12.40 يبلغ عدد الإنذارات الصادرة منها (7) إنذارات كما نلاحظ فقط ظهورها في Dump1 والذي يشير أيضاً إلى احتمالية إصابة الحاسبة 172.20.12.40 بالدودة ونعلل سبب عدم ظهورها في Dump2 و Dump3 إلى انطفاء الحاسبة أو انقطاعها عن الشبكة.

أما بالنسبة للحاسبة 172.20.19.14 يبلغ عدد الإنذارات الصادرة منها (3) إنذارات فقط ومن خلال تحليل البيانات الملتقطة تبين أن فاعلية هذه الحاسبة قد بدأت في نهاية Dump1 واستمرت في Dump2 و Dump3 حيث أن عدد الإنذارات الصادرة منها بالنسبة إلى Dump2 كانت (10) إنذارات و (6) إنذارات بالنسبة إلى Dump3 لهذا يمكن اعتبار هذه الحاسبة مصابة بالدودة أيضاً.

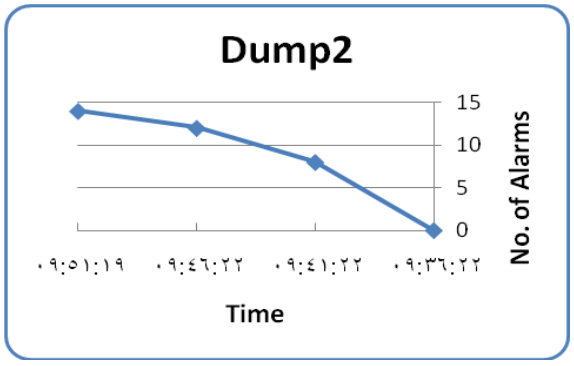
وأما بالنسبة للحاسبة 172.20.67.17 يبلغ عدد الإنذارات الصادرة منها إنذاران فقط كما نلاحظ فقط ظهورها في Dump1 والذي يشير إلى احتمالية ضئيلة لإصابتها بالدودة ونعلل سبب انخفاض عدد الإنذارات الصادرة منها إلى انطفاء الحاسبة أو انقطاعها عن الشبكة.

وأخيراً بالنسبة للحاسبة 172.20.39.88 نلاحظ أول ظهور لها كان في الشكل (5) التابع للملف Dump2 حيث يبلغ عدد الإنذارات الصادرة منها (13) إنذاراً واستمرار إطلاقها للإنذارات في Dump3 والبالغ عددها (12) إنذاراً ويمثل هذا مؤشراً قوياً على احتمالية إصابة هذه الحاسبة بالدودة.

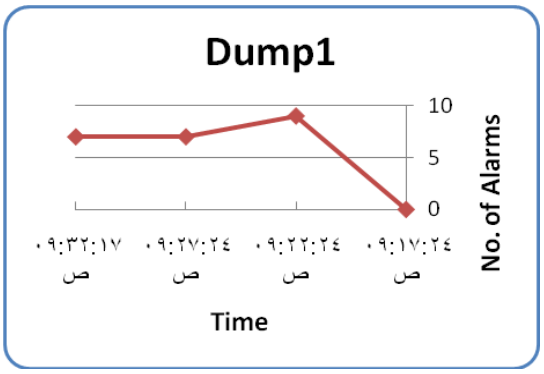
أما بالنسبة للشكل (7) التابع للملف Dump4 نلاحظ أن عدد الإنذارات الصادرة من الحاسبة 172.20.68.49 يبلغ (44) إنذاراً ويمثل هذا مؤشراً قوياً على احتمالية إصابة هذه الحاسبة بالدودة علماً أنها الحاسبة الوحيدة التي كانت تطلق إنذارات في ذلك اليوم.

وأما بالنسبة للشكل (8) التابع للملف Dump5 نلاحظ أن عدد الإنذارات الصادرة من الحاسبة 172.20.12.21 يبلغ إنذاراً واحداً فقط ويمثل هذا مؤشراً ضئيلاً على احتمالية إصابة هذه الحاسبة بالدودة علماً أنها الحاسبة الوحيدة التي كانت تطلق إنذارات في ذلك اليوم.

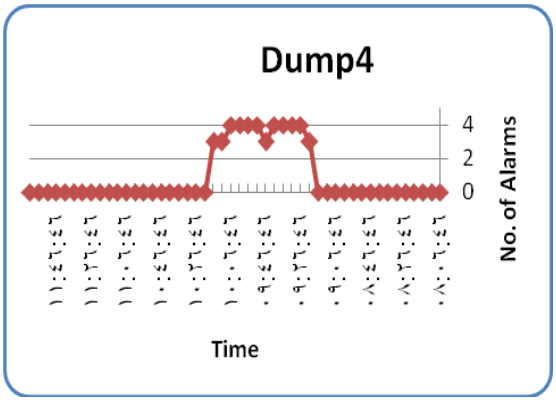
ب. رسم بياني مع تقسيم الزمن الملتقطة ضمنه سيل البيانات إلى فترات زمنية تساوي (5) دقائق تقريباً وحساب عدد الإنذارات الصادرة من الحاسبات بصورة عامة، ظهرت لنا النتائج المبينة في الأشكال (9)، (10)، (11)، (12)، (13) وكالآتي:



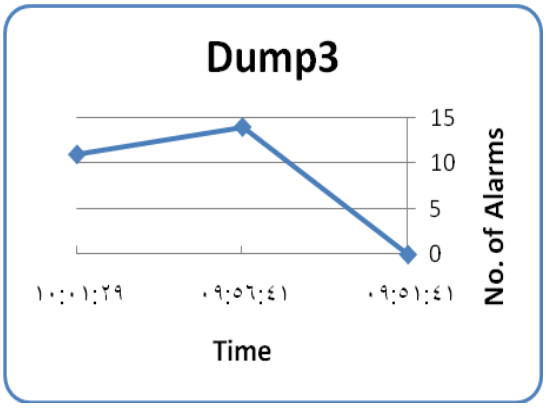
الشكل (9). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump1 مع تقسيم الزمن إلى فترات زمنية تساوي (5) دقائق تقريباً



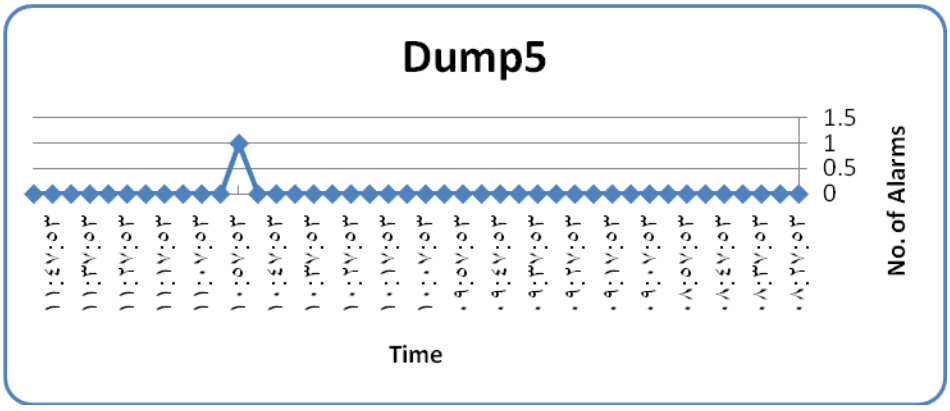
الشكل (10). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump2 مع تقسيم الزمن إلى فترات زمنية تساوي (5) دقائق تقريباً.



الشكل (11). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump3 مع تقسيم الزمن إلى فترات زمنية تساوي (5) دقائق تقريباً



الشكل (12). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump4 مع تقسيم الزمن إلى فترات زمنية تساوي (5) دقائق تقريباً



الشكل (13). يبين عدد الإنذارات الصادرة من الحاسبات المكتشفة في الملف Dump5 مع تقسيم الزمن إلى فترات زمنية تساوي (5) دقائق تقريباً

نلاحظ في الشكل(9) التابع للملف Dump1 أن عدد الإنذارات الصادرة في الخمس دقائق الأولى بلغ (9) إنذارات ثم انخفضت إلى (7) إنذارات في الخمس دقائق الثانية ومن ثم استقرت على هذه القيمة إلى نهاية فترة التنفيذ.

أما بالنسبة للشكل(10) التابع للملف Dump2 نلاحظ أن عدد الإنذارات الصادرة في الخمس دقائق الأولى بلغ (8) إنذارات ثم ارتفعت إلى (12) إنذاراً في الخمس دقائق الثانية واستمرت بالارتفاع بشكل طردي إلى نهاية فترة التنفيذ.

وأما بالنسبة للشكل (11) التابع للملف Dump3 نلاحظ أن عدد الإنذارات الصادرة في الخمس دقائق الأولى بلغ (14) إنذاراً ثم انخفضت إلى (11) إنذاراً في نهاية فترة التنفيذ.

نستنتج من الأشكال(9)، (10)، (11) أن هنالك سلوك شاذ للشبكة في تلك الفترة نتيجة إطلاق إنذارات بشكل ملحوظ وربما يعود هذا النشاط إلى انتشار ديدان في تلك الفترة.

أما بالنسبة للشكل (12) التابع للملف Dump4 نلاحظ أنه في الساعة الأولى تقريباً لم يكن هنالك أي سلوك شاذ للشبكة ولكن في الساعة الثانية ارتفع عدد الإنذارات إلى (3) إنذارات ومن ثم (4) إنذارات والتقلب بين هاتين القيمتين طيلة هذه الساعة ولكن بعد ذلك أصبح عدد الإنذارات صفراً مرة أخرى، إن هذه الإنذارات كانت تصدر من الحاسبة 172.20.39.88 فقط ويعود السبب وراء انخفاض عدد الإنذارات الصادرة منها إلى انطفاء الحاسبة أو انقطاعها عن الشبكة وهذا يعزز ما تم تحليله في النقطة (أ).

وأما بالنسبة للشكل (13) التابع للملف Dump5 نلاحظ وجود إنذار واحد فقط صادر من الحاسبة 172.20.12.21 في الزمن (10:53:44) ومن الواضح أن نشاط الشبكة في ذلك اليوم كان طبيعياً ولا يوجد أي نشاط ملحوظ للدودة في ذلك اليوم.

9. الاستنتاجات

من خلال تطبيق أداة المراقبة المصممة في هذا البحث لغرض الكشف عن الحاسبات المصابة بالدودة وعلى وفق النتائج التي تم الحصول عليها، تم التوصل إلى الاستنتاجات الآتية :

1. بعد دراسة طرائق الكشف عن الدودة الحاسوبية تم اختيار الخوارزمية المعتمدة لبناء أداة المراقبة وذلك لاعتمادها في الكثير من البحوث العلمية.
2. تحدث عملية الاتصال بين شبكة الجامعة وشبكة الانترنت عبر عدد قليل من المنافذ (53, 443, 80, 25, 110) وأما بقية المنافذ فتكون مغلقة. هذا يعني استحالة إصابة الشبكة بالديدان التي تستخدم المنافذ المغلقة مع احتمالية تعرضها للإصابة بمثل هذه الديدان فقط إذا تم ربط حاسبة مصابة من خارج شبكة الجامعة وفي هذه الحالة ستنتشر الإصابة داخل حدود شبكة الجامعة فقط. أو تعرضها للإصابة بنوع الديدان الذي يستخدم أكثر من ثغرة أمنية واحدة. في جميع الأحوال عملية المسح للحاسبات المصابة بالدودة ستكون مقتصرة فقط على المنافذ المفتوحة وهذا ما تم ملاحظته عند تنفيذ أداة المراقبة المصممة على سيل البيانات المخزون في الملفات المبينة في الجدول رقم (2) .
3. إن المبدل الرئيس الذي تم تنصيب أداة المراقبة عليه يمكننا فقط من مراقبة الاتصالات التي تحدث مابين شبكة الجامعة وشبكة الانترنت ولا يشتمل على الاتصالات مابين الأجهزة داخل شبكة الجامعة. يعني أن أداة المراقبة كانت تقوم فقط بمراقبة وتحليل وحساب محاولات الاتصال الفاشلة الواردة من شبكة الانترنت فقط، وهذا يدل على أن أداة المراقبة قادرة فقط على اكتشاف أجهزة الحاسوب المصابة بديدان المسح العشوائي والتي تكون نسبة عمليات المسح لها خارج شبكة الجامعة أعلى من عمليات المسح مابين أجهزة الجامعة.

10. الأعمال المستقبلية Future Works

إن ما تم التطرق إليه في هذا البحث من موضوعي أمنية الشبكات والبرامج الخبيثة وبالأخص الدودة

الالكترونية يمكن أن يكونا نقطتي انطلاق لأفكار وأعمال مستقبلية متنوعة. ولغرض الوصول إلى حلول أخرى مطورة وأكثر كفاءةً تم اقتراح الأعمال المستقبلية الآتية :

1. محاولة تطبيق أداة المراقبة المقترحة على بيانات أكثر من التي تم الحصول عليها للحصول على نتائج أفضل.
2. تطوير أداة المراقبة المصممة لتنفيذها في الزمن الحقيقي، علماً أن الأداة الحالية مهيأة تقريباً للعمل في الزمن الحقيقي.
3. تحسين أداة المراقبة لتكون أداة كشف واستجابة response في نفس الوقت. يمكن أن تكون الاستجابة عبارة عن غلق منفذ الشبكة على المبدل الرئيس الواقعة ضمنها الحاسبة المصابة بأقل تقدير لحين إصلاح الحاسبة ولمنع حدوث تفشي في الإصابة.
4. تحسين أداة المراقبة لتقوم بالكشف عن الديدان التي تستغل أكثر من ثغرة أمنية واحدة (أي التي تقوم بعملية مسح أكثر من منفذ وجهة).
5. ربط أداة المراقبة بقاعدة بيانات تحوي على تواقيع الديدان المعروفة لحد الآن .
6. تحسين أداة المراقبة لتقوم بالكشف عن الديدان التي تستخدم رقم منفذ مصدر ثابت ومنفذ وجهة مختلف كما في دودة " ويتي".
7. تحسين أداة المراقبة لتقوم بالكشف عن ديدان البريد الالكتروني وقائمة الاغتيال بالإضافة إلى تلك التي تقوم بعمليات مسح تسللي.

المصادر

[1] السرحاني ، محمد بن نصير محمد،(2004)، "مهارات التحقيق الجنائي الفني في جرائم الحاسوب"، رسالة ماجستير، قسم العلوم الشرطية، كلية الدراسات العليا، جامعة نايف العربية للعلوم الأمنية.

[2] عريان، عمار، (2003)، "المرجع الشامل في الشبكات"، شعاع للنشر والعلوم، حلب، سورية.

[3] مرجان، هيثم، (2001)، "دورة في كتاب JAVA2"، شعاع للنشر والعلوم، حلب، سورية.

[4] Ali Q. I. and Alabady S. A.J.,(2007)," Design and Implementation of a Secured Remotely Administrated Network", ACIT2007, Lattakia , Syria, pp.381-385.

[5] Chen S. and Tang Y., (2004),“Slowing Down Internet Worms,” Proc. 24th IEEE Int’l Conf. Distrib. Comp.. Sys..

[6] Fujii K.,(2009), "Jpcap Tutorial",
http://netresearch.ics.uci.edu/kfujii/jpcap/doc/index.html

[7] Göldi C. and Hiestand R.,(2005),"Scan Detection Based Identification of Worm-Infected Hosts". Master’s thesis, ETH Zurich.

[8] Hall P.,(2000),"Thinking in Java, 3rd ed.".http://www.BruceEckel.com

[9] Jung J., Paxon V., Berger W., and Balakrishnan H.,(2004), “Fast Port scan Detection Using Sequential Hypothesis Testing” ,Proc. IEEE Security and Privacy.

[10] Jung J., Schechter S. E., and Berger A. W.,(2004)," Fast Detection of Scanning Worm Infections", In Proceedings of RAID’2004.

[11] Kim H. and Karp B.,(2004), “Autograph: Toward Automated, Distributed Worm Signature Detection,” Proc. 13th USENIX Sec. Symp..

[12] Kohli P.,(2005),"Worms - survey and propagation", MS by Research - Computer Science and Engineering International Institute of Information Technology Hyderabad, India.

[13] Paul J. Deitel and Harvey M. Deitel,(2002),"Java How to Program: Fourth Edition", Prentice Hall.

[14] Shah K., Bohacek S. and Broido A.,(2004),"Feasibility of Detecting TCP-SYN Scanning at a Backbone Router", Proceeding of the American Control Conference.

[15] Stevens W. R., (1994), "TCP/IP Illustrated, Volume 1", The Protocols, 1st Edition, is a complete and detailed guide to the entire TCP/IP protocol suite, Addison Wesley Longman, Inc.

[16] Tang Y.,(2006),"defending against internet worms", Phi. D., university of Florida.

[17] Wang S. P.,(2003) ,"Java With Object-Oriented Programming ", Kent State university.

[18] Windump, http://www.winpcap.org/windump.

[19] Wu J.,Vangala S.,Gao L., and Kwiat K.,(2004), “An Effective Architecture And Algorithm For Detecting Worms With Various Scan Techniques,” in Proc. 1th Ann. Network and Distributed System Security Symposium (NDSS’04), San Diego, CA.

[20] Yang X. , Lu J., Zhu Y., Wang P.,(2006),"Simulation and Evaluation OF A New Algorithm of Worm Detection and Containment", Proceedings of the Seventh International Conference on Parallel and Distributed Computing, Applications and Technologies.