NETWORK AND APPLICATION SECURITY

УДК 004.056.5

АНДРІЙ ГОЛОВІН

ВИЯВЛЕННЯ DDOS-АТАК ПРИКЛАДНОГО РІВНЯ ШЛЯХОМ ВИКОРИСТАННЯ МОДЕЛІ MAPREDUCE

У статті розглянуто теоретичні основи *DDoS*-атак типу *HTTP GET flood* та запропоновано багатокритеріальний метод виявлення *DDoS*-атак в режимі реального часу на основі моделі розподілених обчислень *MapReduce*. З'ясовано, що запропонований метод на базі ПЗ *Apache Hadoop* можна застосовувати для виявлення *DDoS*-атак за прийнятний інтервал часу. Можливості горизонтального масштабування обчислювальних ресурсів дозволяють застосовувати метод для виявлення атак різної потужності. Представлені результати показують, що кластер на базі 10 вузлів виявляє джерела (зомбі хости) *DDoS*-атаки із трафіку 16 Гб менш ніж за 5 хвилин. При чому більшу частину часу (в найгіршому випадку 75%) займає процес захоплення трафіку і передачі лог-файлу в *Hadoo*р кластер.

Ключові слова: DDoS-атака, HTTP flood атака, MapReduce, Apache Hadoop

Постановка проблеми. Розподілена атака на відмову в обслуговуванні (*Distributed Denial of Service, DDoS*) є однією із найбільш небезпечних загроз у сфері захисту інформації у комп'ютерних мережах. *DDoS*-атака це напад на комп'ютерну систему з наміром зробити комп'ютерні ресурси недоступними до користувачів, для яких комп'ютерна система була призначена.

Кількість інцидентів і потужність *DDoS*-атак постійно зростають, у березні 2013 року була зареєстрована найпотужніша *DDoS*-атака в історії із піковою потужністю 300 Гбіт/с. В той момент мільйони користувачів мережі Інтернет відчули результати атаки, в основних центрах обміну трафіку (*Internet Exchange Point*) швидкість обміну даними впала у 3 рази, доступ до більшості сайтів був обмеженим.

Умовно *DDoS*-атаки можна поділити на два підвиди: мережевого та прикладного рівня. Декілька років тому найбільш дієвими були *DDoS*-атаки мережевого рівня, які мають за мету перевантажити канал, проте сьогодні вони не створюють серйозної небезпеки, завдяки розробленим методам боротьби з ними. При цьому, *DDoS*-атаки прикладного рівня продовжують бути ефективним інструментом для перевантаження інформаційних ресурсів. Під *DDoS*-атаками прикладного рівня розуміють атаки, спрямовані на мережеві сервіси, що функціонують на рівні додатків, згідно класифікації по еталонній моделі *OSI/ISO*. Зазвичай, їхня дія проявляється у вивантаженні великої кількості об'ємних файлів або переповненні ресурсу ресурсномісткими запитами. У випадку застосування протоколу *HTTP* у *DDoS*-атаках (*HTTP flood*) прикладного рівня, синтаксис *HTTP*-запитів та характеристики рівня трафіку нічим не відрізняються від легітимного трафіку, а тому їх виявлення у реальному часі є досить складною задачею. Вирішенням проблеми *DDoS*-атак прикладного рівня є раннє виявлення факту їх початку, ідентифікація джерел атаки, а також повідомлення адміністраторам зомбі-вузлів. В такому випадку атака може бути ефективно контрольована.

В статті розглянуто застосування моделі обробки даних *MapReduce* [1] для виявлення *DDoS*-атак прикладного рівня в реальному часі.

Аналіз останніх досліджень і публікацій. Останнім часом було запропоновано різні методики і способи зниження впливу *DDoS*-атак на мережеве середовище. Jie-Hao та Ming [2] використовували штучні нейронні мережі для виявлення *DDoS*-атаки. Автори порівнюють ефективність виявлення *DDoS*-атак за допомогою таких методів, як дерево рішень, аналіз ентропії, метод наївного Баєсівського класифікатору та метод штучних

нейронних мереж. Liu та Gu [3] використали нейронну мережу із квантизацією векторів в процесі навчання. Це контрольований варіант квантування, який може бути використаний для розпізнавання образів, мультикласової класифікації та для стиснення даних. Akilandeswari та Shalinie [4] представили класифікатор атак на базі ймовірнісної нейронної мережі. Siaterlis та Maglaris [5] проводили експерименти із багатошаровим перцептроном (*Multi-Layer Perceptron*), як алгоритмом прийняття рішення при класифікації трафіку.

Bharanidharan Shanmugam та Norbik Bashah Idris [6] запропонували використання гібридної системи виявлення вторгнень на основі апарату нечіткої логіки (*Fuzzy Logic Based Hybrid IDS*). Система складається із таких компонентів: аналізатор даних, модуль видобутку даних та модуль нечіткого виводу. Аналізатор даних досліджує трафік і здійснює агрегацію пакетів. Агрегована інформація надходить до модуля видобутку даних, який генерує правила фільтрації. Правила фільтрації та мережевий трафік надходять до модуля нечіткого виводу, який приймає рішення про наявність зловмисного впливу на систему. На практиці метод не може бути використаний для виявлення атак в реальному часі через високу обчислювальну складність та високий рівень помилкових спрацювань.

Мета і завдання статті. Метою статті є поглиблене вивчення, розвиток і вдосконалення методів виявлення *DDoS*-атак прикладного рівня, а також представлення методу застосування моделі обчислень *MapReduce* для ефективного виявлення *DDoS*-атак в реальному часі.

Виклад основного матеріалу. Атака типу *HTTP GET flood* використовується зловмисниками для нападу на веб-сервери та сервери веб-додатків. Атака являє собою сукупність начебто легітимних *GET* або *POST* запитів на сервер. Це спеціально розроблені запити для споживання значного обсягу ресурсів серверу. В результаті вони можуть призвести до стану відмови в обслуговуванні, при чому, без необхідності переповнення каналу великим обсягом трафіку. Такі запити у випадку розподіленої *DoS*-атаки відправляються із десятків тисяч заражених (зомбі) вузлів. На рис. 1 схематично зображено послідовність пакетів при *HTTP GET* запиті після встановлення *TCP* з'єднання:

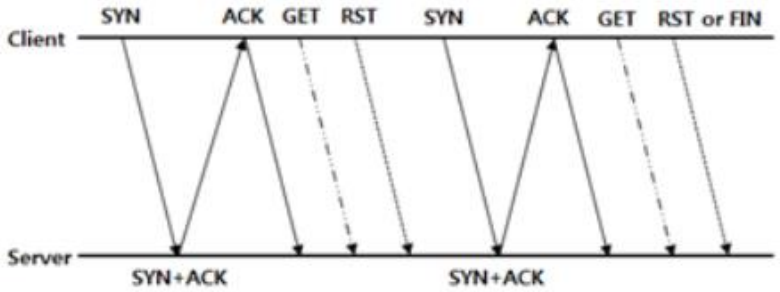

Рисунок 1− Послідовність пакетів при атаці типу HTTP GET

В процесі атаки зловмисник постійно відправляє запити, створюючи при цьому нові *TCP* з'єднання. Останнім часом [8] також стали розповсюдженими *HTTP GET flood* атаки в рамках одного *TCP* з'єднання (див. рис. 2). Цей тип атаки не можливо виявити методом оцінки кількості *SYN* запитів.

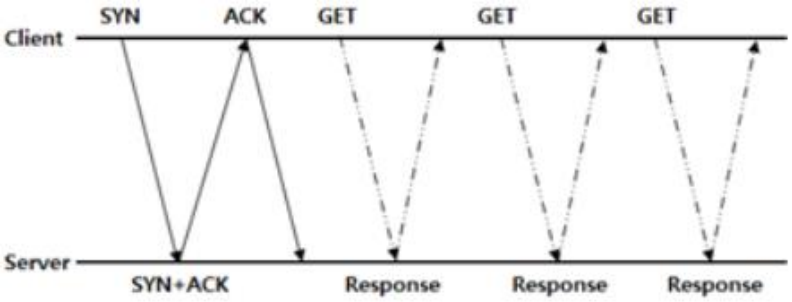

Рисунок 2 − Послідовність пакетів при атаці типу HTTP GET в рамках одного TCP з'єднання

HTTP flood атаки зараз є однією із найбільш передових загроз інформаційній безпеці, що безпосередньо не пов'язана із вразливостями програмного забезпечення. Для обладнання захисту відрізнити зловмисні *HTTP* запити від легітимних є надзвичайно складною задачею, невірні методи чи налаштування призводять до великої кількості помилкових спрацювань. Використання метрик, які ґрунтуються лише на оцінці інтенсивності запитів не є оптимальною методикою виявлення *DDoS*-атак типу *HTTP flood*, оскільки обсяг трафіку при цьому може бути нижчим за поріг спрацювання. Через це доцільно використовувати багатокритеріальний метод виявлення *DDoS*-атак, із метриками залежними від інтенсивності запитів, так і тими, що не залежать від цього показника.

Застосування моделі обчислень MapReduce для виявлення DDoS-атаки типу HTTP GET flood. *MapReduce* – модель проведення розподіленої паралельної обробки великих масивів даних з використанням кластерів (великої кількості обчислюваних блоків). Робота *MapReduce* складається із двох етапів: *Map* і *Reduce*.

На етапі *Map* виконується попередня обробка вхідних даних. Для цього один із обчислювальних елементів кластеру (головний вузол, *master node*) отримає вхідні дані для розрахунку і розподіляє дані серед робочих вузлів.

На етапі *Reduce* відбувається зворотна згортка попередньо оброблених даних. Головний вузол отримує відповіді від робочих вузлів і на їх основі формує результат – рішення задачі.

Перевага моделі *MapReduce* полягає в тому, що вона дозволяє паралельно і незалежно виконувати операції попередньої обробки і згортки, а також горизонтально масштабувати обчислювальну можливість кластеру. Операції попередньої обробки працюють незалежно один від одного і можуть проводитися паралельно (хоча на практиці це обмежено джерелом вхідних даних та/або кількістю використовуваних обчислювальних блоків). Аналогічно, група робочих вузлів можуть здійснювати згортку – для цього необхідно тільки щоб всі результати попередньої обробки з одним конкретним значенням ключа оброблялися одним робочим вузлом в один момент часу.

Багатокритеріальний метод виявлення DDoS-атак типу HTTP flood. Для роботи багатокритеріального методу виявлення *DDoS*-атак необхідно провести попередній аналіз і розрахунки: визначити показники (критерії) по яким буде ідентифікуватися наявність або відсутність атаки; побудувати модель для нормального трафіку в мережі; задати порогові значення для обраних показників.

В якості критеріїв оцінки були обрані наступні показники:

- рівень завантаження центрального процесору серверу;
- обсяг зайнятої оперативної пам'яті серверу;
- розмір пакету;
- поточний рівень трафіку (Mbit/s);
- розподіл значення адреси джерела запитів (*source ip*);
- *user-agent* в запиті;
- *URI* (ієрархічна частина та фрагменти *url* запиту);

Наявність атаки *HTTP GET flood* можливо охарактеризувати по кількості запитів із джерела в секунду (*GET Request per Second, GRPS*) [8]. Зрозуміло, що легітимний користувач постійно не здійснює велику кількість запитів до одного і того ж самого ресурсу, як це може робити вузол, підконтрольний зловмиснику (зомбі). Отже, за деякий проміжок часу Δt з *IP*-адреси джерела x_i надходить s_i запитів.

Введемо поняття ентропії *HTTP GET* запитів відносно джерела з яких вони надходять:

$$H(s) = \sum_{i=1}^{n} p(x_i) \log p_i ,$$

де $p(x_i)$ – імовірність надходження запиту s_i із джерела x_i:

$$p(x_i) = \frac{s_i}{\sum_i s_i},$$

В результаті отримаємо значення ентропії $H_1, H_2, .., H_t$ у послідовні проміжки часу. Необхідно визначити поріг *GRPS*, при якому запити із джерела x_i вважати зловмисними. Занизький поріг може призвести до появи похибок першого роду (помилкове спрацювання) при визначенні наявності атаки, а зависокий – відповідно до похибок другого роду (пропуск події). Враховуючи можливість наявності похибки із за неоднорідності мережевого середовища та відносну складність обчислення значення *GRPS*, у випадку неоднозначності показника порогового значення $H_{\max}$ необхідно враховувати оцінку інших критеріїв.

Запропонований процес виявлення *DDoS*-атак прикладного рівня складається із наступних етапів (див. рис. 3):

1. Захоплення трафіку і накопичення лог-файлу з даними по вхідних запитах на сервер.
2. Передача лог-файлу на обчислювальний кластер.
3. Визначення значення ентропії *GRPS* за допомогою моделі *MapReduce*.
4. Оцінка значень та прийняття рішення.

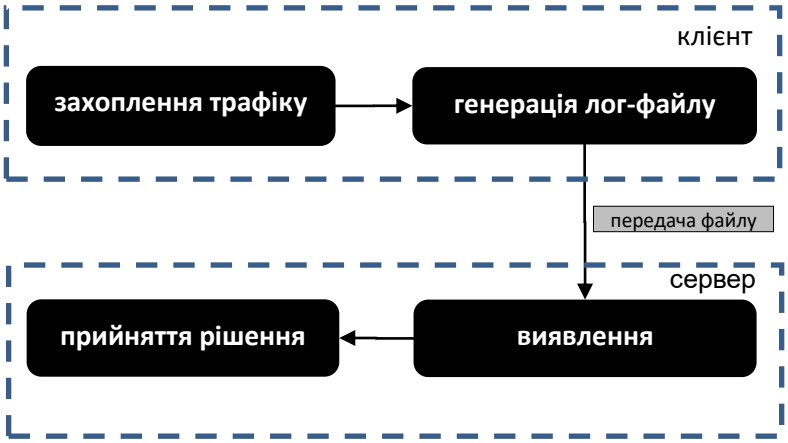

Рисунок 3 − Схема роботи методу виявлення HTTP GET flood

Лог-файл містить інформацію про запити, значення обраних для оцінки критеріїв, а також мітку часу, адресу відправника, адресу отримувача, протокол:

139875; 10.1.12.73; 10.1.12.101;
HTTP/1.1; 730; GET; /posts/23897/324/req?k=12876182746;
UA=Mozilla/5.0 (Macintosh; U; Intel Mac OS X; en-US) AppleWebKit/533.4 (KHTML, like Gecko) Chrome/5.0.375.86 Safari/533.4;
CPU=12; RAM=66; LOAD=3;

На етапі виявлення моделі *MapReduce* необхідно задати інтервал часу вимірювання (*time interval*), порогове значення кількості запитів на сервер (*threshold*) та порогове значення кількості запитів між окремим клієнтом і сервером (*peer threshhold*). Функція *Map* фільтрує інші типи запитів окрім *GET* та генерує значення ключа (*IP*-адреса джерела, *IP*-адреса призначення, маска мітки часу і інтервалу часу вимірювання). Алгоритм зображено на рис. 4:

Після перевищення заданого рівня кількості запитів на сервер (*threshold*) *MapReduce* знаходить запити із завищеним *peer treshold* та блокує відправника цих запитів.

Результати дослідження. Для генерації *HTTP GET* запитів при емуляції *DDoS*-атаки є ряд утиліт, які доступні у відкритому доступі *LOIC* [9], *Scapy* [10], *Mausezahn* [11], *Iperf* [12]. Для експерименту був обраний *Mausezahn* із за можливості генерувати потік запитів із підставною адресою відправника.

Для розподілених обчислень за моделлю *MapReduce* було використано відкрите програмне забезпечення *Apache Hadoop*. До складу *Hadoop* входить також реалізація розподіленої файлової системи *Hadoop Distributed Filesystem* (*HDFS*) для зберігання лог-файлів великих розмірів.

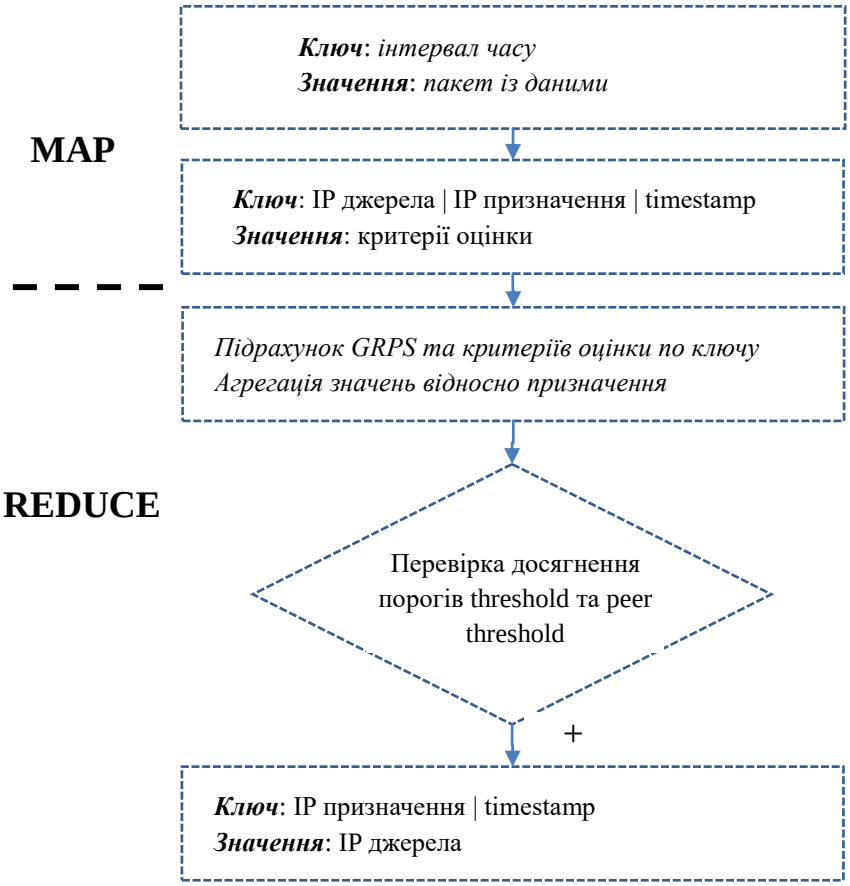

Рисунок 4 − Алгоритм виявлення атаки типу HTTP GET flood із застосуванням моделі MapReduce

Для визначення оптимальних налаштувань кластера *Hadoop* в експерименті послідовно використовувалось від 2 до 10 обчислювальних вузлів, лог-файли розміром від 10 до 1000 Мб та розміри блоку для розподілення обчислень від 32 до 128 Мб.

В результаті проведення експерименту було з'ясовано:

1. *Hadoop* кластер здатен аналізувати великі обсяги *HTTP GET* flood *DDoS*-атак із можливістю горизонтального масштабування обчислювальних ресурсів.
2. Атаку потужністю у декілька Гбіт/с можна виявити на протязі 3-5 хвилин.
3. Значення ентропії, що наближені з обох сторін до порогу треба перевіряти іншими критеріями оцінки (рівень завантаження центрального процесору, обсяг зайнятої оперативної пам'яті серверу).
4. Інтервал виявлення можна зменшити до десятків секунд (30-40) зменшивши при цьому розмір лог-файлу. Проте це вплине на кількість зловмисних джерел трафіку, які система буде здатна виявити.
5. На малих розмірах лог-файлу (<650Мб) кластер не виконує розподіл завдання серед вузлів, тому нарощування кількості обчислювальних вузлів для таких розмірів лог-файлу не має сенсу.
6. Найбільший виграш у швидкодії при нарощенні кількості обчислювальних блоків для 1000 Мб лог-файлу дає розмір блоку розподілених обчислень у 128 Мб. Це пояснюється тим, що для менших розмірів блоку зростають накладні витрати на розподіл вхідних даних між вузлами і зворотну згортку результатів.

Висновки. В статті розглянуто теоретичні основи *DDoS*-атак типу *HTTP GET flood* та запропоновано багатокритеріальний метод виявлення *DDoS*-атак в режимі реального часу на основі моделі розподілених обчислень *MapReduce*. З'ясовано, що запропонований метод на базі ПЗ *Apache Hadoop* можна застосовувати для виявлення *DDoS*-атак за прийнятний

інтервал часу. Можливості горизонтального масштабування обчислювальних ресурсів дозволяють застосовувати метод для виявлення атак різної потужності. Представлені результати показують, що кластер на базі 10 вузлів виявляє джерела (зомбі хости) *DDoS*-атаки із трафіку 16 Гб менш ніж за 5 хвилин. При чому більшу частину часу (в найгіршому випадку 75%) займає процес захоплення трафіку і передачі лог-файлу в *Hadoop* кластер. В подальшому доцільно провести оптимізацію процесу захоплення трафіку, формування та передачі лог-файлу до кластеру, а також враховувати в аналізаторі інші критерії оцінки для значення ентропії близького до порогового.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Lammel R. Google's mapreduce programming model revisited / R. Lammel // Science of Computer Programming. – January 2008. – No. 70 (1). – P. 1-30.

2. Jie-Hao C. DDoS defense system with test and neural network / C. Jie-Hao, Z. Ming, C. Feng-Jiao, Z. An-Di // Proceedings of the IEEE International Conference on Granular Computing (Hangzhou, China, 11-13 Aug. 2012). – Hangzhou, 2012 – P. 38-43.

3. Li J. DDoS attack detection based on neural network / J. Li, Y. Liu, L. Gu // Proceedings of the 2nd International Symposium on Aware Computing (Tainan, 1-4 November, 2010). – Tainan, 2010. – P. 196-199.

4. Akilandeswari V. Probabilistic neural network based attack traffic classification / V. Akilandeswari, S. Shalinie // Proceedings of the Fourth International Conference on Advanced Computing (Chennai, 13-15 Dec. 2012). – Chennai, 2012. – P. 1-8

5. Siaterlis C. Detecting incoming and outgoing DDoS attacks at the edge using a single set of network characteristics / C. Siaterlis, V. Maglaris // Proceedings of the 10th IEEE Symposium on Computers and Communications (Washington, 27-30 June 2005). Washington, 2005. – P. 469-475.

6. Shanmugam B. Improved Intrusion Detection System using Fuzzy Logic for Detecting Anamoly and Misuse type of Attacks / B. Shanmugam, N. Idris // International Conference of Soft Computing and Pattern Recognition (Malacca, 4-7 December 2009). – Malacca, 2009. – P. 212-217.

7. DDoS Definitions – DdoSPedia [Electronic resource] – Access mode : http://security.radware.com/knowledge-center/DDoSPedia/http-flood/. – Access data : July 2015. – The title of the screen.

8. Зінченко В. В. Виявлення DDoS-атак прикладного рівня / В. В. Зінченко, М. В. Зінченко // Міжнародна науково-технічна конференція «Радіотехнічні поля, сигнали, апарати та системи» (Київ, 16-22 березня 2015). – К., 2015. – С. 262-264.

9. LOIC (Low Orbit Ion Cannon) : A network stress testing application [Electronic resource] – Access mode : http://sourceforge.net/projects/loic/. – Access data : July 2015. – The title of the screen.

10. Scapy Project [Electronic resource] – Access mode : http://www.secdev.org/projects/scapy/ – Access data : August 2015. – The title of the screen.

11. Mausezahn [Electronic resource] – Access mode : http://www.perihel.at/sec/mz/ – Access data : August 2015. – The title of the screen.

12. Iperf : network performance measurement tool [Electronic resource] – Access mode : https://iperf.fr/ – Access data : August 2015. – The title of the screen.

Стаття надійшла до редакції 9 вересня 2015 року.

REFERENCE

1. Lammel, R. (2008), *Google's MapReduce programming model – revisited*, Science of Computer Programming, No. 70 (1), pp. 1-30.

2. Jie-Hao, C., Ming, Z., Feng-Jiao, C., An-Di, Z. (2012), *DDoS defense system with test and neural network*, Proceedings of the IEEE International Conference on Granular Computing (GrC), Hangzhou, China, pp. 38-43.

3. Li, J., Liu, Y., Gu, L. (2010), *DDoS attack detection based on neural network*, Proceedings of the 2nd International Symposium on Aware Computing (ISAC), Tainan, pp. 196-199.

4. Akilandeswari, V., Shalinie, S. (2012), *Probabilistic neural network based attack traffic classification*, Proceedings of the Fourth International Conference on Advanced Computing (ICoAC), Chennai, pp.1-8.

5. Siaterlis, C., Maglaris, V. (2005), *Detecting incoming and outgoing DDoS attacks at the edge using a single set of network characteristics*, Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC), Washington, pp. 469-475.

6. Shanmugam, B., Idris, N. (2009), *Improved Intrusion Detection System using Fuzzy Logic for Detecting Anamoly and Misuse type of Attacks*, Proceedings of the International Conference of Soft Computing and Pattern Recognition, Malacca, pp. 212-217.

7. *DDoS Definitions – DdoSPedia*, available at : http://security.radware.com/knowledge-center/DDoSPedia/http-flood (accessed 9 July 2015).

8. Zinchenko, V. V, Zinchenko, M. V (2012), *Viyavlennya ddos-atak prikladnogo rivnya* [Detection of application layer DDoS attacks], Mizhnarodna naukovo-tehnichna konferentsiya «RadIotehnichni polya, signali, aparati ta sistemi», Kyiv, pp. 262-264.

9. *LOIC (Low Orbit Ion Cannon) : A network stress testing application*, available at: http://sourceforge.net/projects/loic/ (accessed 9 July 2015).

10. *Scapy Project*, available at : http://www.secdev.org/projects/scapy/ (accessed 19 August 2015).

11. *Mausezahn*, available at : http://www.perihel.at/sec/mz/ (accessed 19 August 2015).

12. *Iperf : network performance measurement tool*, available at : https://iperf.fr/ (accessed 19 August 2015).

АНДРЕЙ ГОЛОВИН

ВИЯВЛЕНИЕ DDOS-АТАК ПРИКЛАДНОГО УРОВНЯ ПУТЕМ ИСПОЛЬЗОВАНИЯ МОДЕЛИ MAPREDUCE

В статье рассмотрены теоретические основы *DDoS*-атак типа *HTTP GET flood* и предложен многокритериальный метод обнаружения *DDoS*-атак в режиме реального времени на основе модели распределенных вычислений *MapReduce*. Установлено, что предложенный метод на базе ПО *Apache Hadoop* можно применять для выявления *DDoS*-атак за приемлемый промежуток времени. Возможности горизонтального масштабирования вычислительных ресурсов позволяют применять метод для обнаружения атак различной мощности. Представленные результаты показывают, что кластер на базе 10 узлов выявляет источники (зомби хосты) *DDoS*-атаки из трафика объемом 16 Гб менее чем за 5 минут. Причем большую часть времени (в худшем случае 75%) занимает процесс захвата трафика и передачи лог-файла на *Hadoop* кластер.

Ключевые слова: DDoS-атака, HTTP flood атака, MapReduce, Apache Hadoop

ANDRII HOLOVIN

DETECTING DDOS ATTACK USING MAPREDUCE OPERATIONS

Denial of Service (*DoS*) and Distributed DoS (*DDoS*) attacks are evolving continuously. These attacks make network resources unavailable for legitimate users which results in massive loss of data, resources and money. Recent distributed denial-of-service (*DDoS*) attacks have demonstrated horrible destructive power by paralyzing web servers within short time. As the volume of Internet traffic rapidly grows up, the current *DDoS* detection technologies have met a new challenge that should efficiently deal with a huge amount of traffic within the affordable response time. This work focuses on novel *DDoS* detection method based on *Hadoop* that implements a *HTTP GET* flooding detection algorithm in *MapReduce* on the distributed computing platform.

Keywords: DDoS Attack, HTTP Flooding Attack, MapReduce, Apache Hadoop.

Андрій Юрійович Головін, аспірант, Державний заклад «Інститут спеціального зв'язку та захисту інформації Національного технічного університету України «Київський політехнічний інститут», Київ, Україна.
E-mail: golovin.010@gmail.com.

Андрей Юрьевич Головин, аспирант, Государственное учреждение «Институт специальной связи и защиты информации Национального технического университета Украины «Киевский политехнический институт», Киев, Украина.

Andrii Holovin, postgraduate student, State institution «Institute of special communication and information security of National technical university of Ukraine «Kyiv polytechnic institute», Kyiv, Ukraine.