УДК 004.77:004.424 **В.В. Василенко**

МАСШТАБОВАНІСТЬ SDN НА ОСНОВІ РЕКОНФІГУРАЦІЇ МОДЕЛІ МЕРЕЖІ З УРАХУВАННЯМ БЕЗПЕКИ І QOS ОБМЕЖЕНЬ

Досліджено питання масштабованості при використанні графа атак на основі аналізу моделі безпеки в програмному і віртуалізованому мережному середовищі. Запропоновано граф атак на основі аналізу рамок безпеки для відстеження вразливостей в кожній віртуальній машині та залежність вразливостей серед віртуальних машин.

Ключові слова: : хмарні технології, безпека даних, віртуалізація, трафік даних, атака, граф атак, гіпервізор, SLA, SDN.

Вступ і постановка завдання. Сьогодення вимагає від компаній малого і середнього бізнесу користування хмарними технологіями, щоб отримати швидкий доступ до кращих бізнес-додатків і покращити ресурси інфраструктури. Саме занепокоєння безпекою хмари перешкоджають міграції клієнтів до даного рішення. У традиційних центрах обробки даних, де системні адміністратори мають повний контроль над хост-машинами, вразливість може бути виявлена і виправлена системним адміністратором в централізованому порядку. Проте, латання дір в безпеці, в центрах обробки даних хмари, де користувачі хмари, як правило, мають привілей управляти встановленим програмним забезпеченням на їх керованих віртуальних машинах, не може працювати ефективно і може порушити «Угоду про рівень обслуговування» (SLA). Крім того, користувачі можуть встановити на хмару вразливе програмне забезпечення або навіть шкідливий код на їх віртуальну машину, що істотно сприяє поломкам в безпеці хмари. Завдання полягає в тому, щоб створити ефективну систему виявлення і реагування для точного виявлення атак і зведення до мінімуму наслідків порушення безпеки для користувачів хмари.

З появою Software-Defined Networking (SDN), хмара-провайдер здатна ефективно виявляти маршрутні підозрілі і шкідливі потоки від користувачів в своїх віртуальних мережах [1] і зручно створювати додатки безпеки в своїх віртуальних мережах [2] для виявлення і реагування на шкідливому трафіку. Проте, механізм динамічної маршрутизації (виконується провайдером хмари) не повинен впливати на «угоду про рівень обслуговування» для користувачів хмари. Тому актуальним є дослідження, як SDN може бути застосований в контексті хмарних обчислень, щоб динамічно змінювати базову мережну інфраструктуру з метою підтримки потреб в області безпеки, врівноважуючи необхідні QoS для користувачів хмари.

В даній роботі пропонується масштабувати структуру для відстеження вразливостей в кожній віртуальній машині і залежність уразливості між віртуальними машинами у віртуальній мережі, використовуючи граф атак аналітичної моделі. Граф атак є інструментом моделювання, щоб проілюструвати всі можливі шляхи багатоступінчастої, мульти-хост-атаки, які мають вирішальне значення для розуміння загроз і застосування відповідних контрзаходів. Однак, добре відомий стан проблем графа атак [3] робить його не в змозі змоделювати сценарії атак в великих масштабах мережі, такі як великі центри обробки даних. Пропонується обмежити розмір графа атак в керованому масштабі з використанням SDN функції динамічної реконфігурації мережі, зберігаючи при цьому мінімальну кореляцію між графами атак.

У цьому контексті ефективним є використання програмних функцій мережі SDN, щоб ізолювати, встановлювати карантин і перевіряти трафік відправки з вразливих служб. Для досягти цієї мети, використовують ковзаючі методи оборони, стримування та усунення атак без переривання регулярного мережного трафіку.

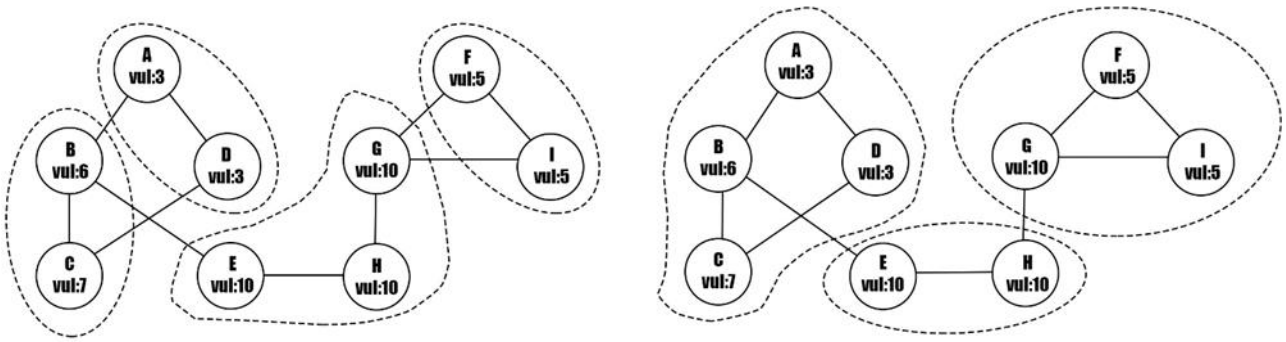
Постановка завдання. Граф атак і дерево атак повинні вирішувати питання масштабованості на різних етапах: формування, подання, оцінки і модифікації [4]. Розмір мережі може збільшуватися (наприклад, зростання кількості мобільних пристроїв), але це не є рішенням для вирішення масштабованості оцінки безпеки. Граф атак і дерево атак досі

страждають від проблеми масштабованості. Дерево атак може бути оцінене тільки таким чином, якщо воно буде ефективно згенероване, але до сих пір немає ефективного способу генерації дерева атак. Отже, як і раніше потрібен більш надійний метод оцінки, заснований на графах і методах генерації.

Використання графа атак моделює можливі моделі поведінки і вразливості атакуючого в мережі [5]. Тим не менше, немає ніякої попередньої роботи, яка використовує граф атак в дуже динамічному віртуалізованому і перебудованому мережному середовищі, і розглядала б застосування оптимальних контрзаходів. Крім того, проблеми масштабованості в режимі реального часу стають істотними в такому динамічному середовищі. Велику кількість віртуальних машин можна розмістити в хмарі у центрі хмари даних навколишнього середовища. Оскільки граф атак зберігає кореляції між угрозвми, і складність атаки групи зростає в геометричній прогресії. Для вирішення цієї проблеми пропонується модель реконфігурації мережі, щоб зменшити масштаби графа атак шляхом поділу контрольованої мережі на безпечні зони. Даний кластерний підхід зосереджений на одній комбінації наступних напрямків кластеризації підходів:

• угрупування віртуальних машин з подібними уразливостями (див. рис. 1 (а));

• під час використання груп взаємопов'язаних віртуальних машин з найменшими мережами і додатками підключення до інших груп (див. рис.1 (б)).



а) б)

Рис. 1. Угруповання віртуальних машин з (а) уразливостями та (б) можливості підключення кожного вузла

Аналіз останніх досліджень і публікацій. Сучасний стан формування методів аналізу та синтезу хмарних технологій нерозривно пов’язано з роботами таких вчених як O. Sheyner, P. Ammann, X. Ou, L. Wang, A. Рой, H. Poolsappasit.. В них розглядається сучасний науково-методичний апарат аналізу й порівняльного оцінювання маштабованосты SDN, проте комплексного дослідження даної проблеми на основі реконфігурації моделі мережі з урахуванням безпеки і QOS обмежень не проводиться. Тому, враховуючи реалії сьогодення, дане питання потребує більш глибокого вивчення.

Актуальність та мета статті. Обчислювальна потужність і зберігання в мережі є віртуалізованим для спільного використання в системі IaaS. Ця важлива технологія робить абстрактні інфраструктури та ресурси доступні для користувачів в якості ізольованих віртуальних машин (VM) та віртуальних мереж (VNs). Тим не менш, вона також підвищує вразливість і можливість атак в системі, так як всі користувачі в хмарі поділяють ці ресурси з іншими користувачами або навіть нападниками. Механізм захисту необхідний для забезпечення суворої ізоляції, опосередкованого спільного використання та безпечного зв'язку між віртуальними машинами.

Виходячи з вище зазначеного, мета статті та її основний зміст полягають у дослідженні питання як SDN може бути застосований в контексті хмарних обчислень, щоб динамічно змінювати базову мережну інфраструктуру з метою підтримки потреб в області безпеки, врівноважуючи необхідні QoS для користувачів хмари.

Виклад основного матеріалу. Почнемо з проектування системи. Представимо першу системну архітектуру запропонованого дизайну, та докладний опис його компонентів.

Пропонована структура показана на рис. 2. Є чотири етапи: побудова графа атак, граф атак, кластеризація, оцінка безпеки і реконфігурація.

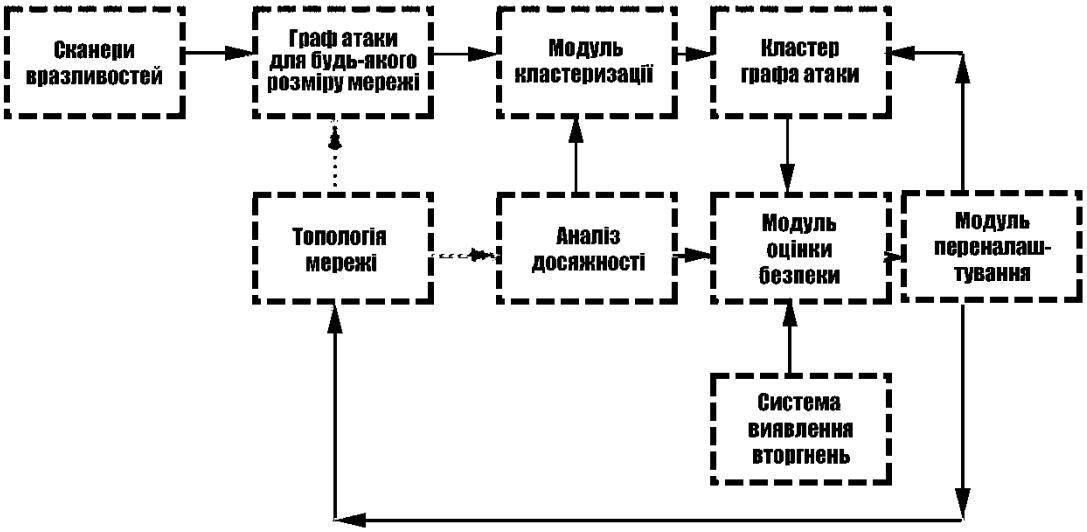


Рис. 2. Конфігурація моделі

Побудова графа атак відповідає за створення графа атак для довільного розміру контрольованої мережі. Граф атак кластерізації відповідає за настройки кожного графа атак, щоб вони були керовані шляхом застосування SDN на основі динамічного підходу реконфігурації при збереженні оригінальних мережевих служб в кожному вузлі. Даний кластерний підхід здатний зменшити розмір графа атак і залежність уразливості серед графів атак шляхом налаштування мережних служб, перенаправляючи мережний шлях до більш безпечного шляху, або реконфігурації топології мережі за допомогою архітектури SDN. Граф кластеризації атаки вимагає щоб топологія мережі та інформація були доступні, щоб налаштувати розмір контрольованої мережі. Індекс безпеки, пов'язаний з модулем оцінки безпеки, також є важливими тригером для активації настройки графа атак і контролювання мережею.

Модуль оцінки безпеки забезпечує показники, пов'язані з безпекою для модуля графа кластерної атаки, для внесення корективів графа атак і реконфігурації топології мережі. Модуль оцінки містить наступні показники для кожного кластера в контрольованій мережі:

- Ni: Розмір (номер вузла) кластера.
- AGI: розмір графа атак для кластера.
- GSII: глобальний індекс безпеки кластера.
- Riski: ймовірність ризику атаки графа для кластера.
- AvgBasei: експоненціальне середнє значення базових вразливостей в кластері.

Модуль оцінки безпеки також отримує повідомлення від системи виявлення вторгнень (IDS) як тільки аномальна подія або рух виявляється в IDS. Це попередження викличе динамічну стратегію реорганізації в модуль реконфігурації для вживання заходів.

Стратегія реконфігурації для захисту системи з використанням програмних мережних функцій програмного забезпечення. Пропонується активна система захисту і профілактики, заснована на уразливості і оприлюдненні інформації в графі атак, яка будує алгоритм, щоб зробити шлях атаки тяжким. Реконфігурація може бути зробленою за допомогою мережевого контролера, що може змінити: передачу потоку, перенаправити поток, відобразити поток, перезаписати MAC-адреси, змінити IP-адреси, відкинути пакети, брандмауера або фільтра і блокувати порт.

Стратегія реконфігурації залежить від програмних мережних функцій програмного забезпечення, певної мережі для забезпечення мережного трафіку за рахунок зміни топології

мережі і мережних ресурсів використання. Стратегія може бути заснована на інформації про уразливість графа атак і попередження від NIDS. Мета стратегії полягає в тому, щоб зробити шлях атаки в графі атаки більш важким для атакуючого. Уразливості і аналізатор атаки на рис. 3 створює стратегію реконфігурації, грунтуючись на подіях в мережі з кількох серверів управління і моніторингу, (наприклад: NIDS, моніторингу пропускної здатності, SNMP, NetFlow і SFlow) і інформації з бази даних графа атак; стратегію контрзаходів. Конфігурація двигуна це синтаксичний аналізатор для перекладу політики і стратегії у визначенні мови високого рівня в інструкції низького рівня. Робота реконфігурації здійснюється за допомогою мережевого контролера, дотримуючись інструкцій з реконфігурації двигуна. Примітивні операції реконфігурації включають: перенаправлення потоку, відображення потоку, переписування MAC-адресу, зміни IP-адреси, падіння пакетів, відкидання пакетів, обмеження трафіку, брандмауера або фільтра і блокування портів. Стратегію реконфігурації можна розділити на дві категорії: статична реконфігурація і динамічна реконфігурації.

Статична реконфігурація на основі поточних вразливостей в системі застосовується для переконфігурації трафіка або віртуальної мережі. Кожен експлуатуючий вузол в AG містить інформацію про уразливість. Статична реконфігурація буде шукати критичний шлях в AG, який є найпростішим шляхом для атакуючого і найбільш незахищеним шляхом у віртуальній мережі. Обраний підхід реконфігурації заснований на категорії атаки і типу вразливостей. Ця інформація може бути легко витягнута з бази даних NVD. Після застосування стратегії реконфігурації кумулятивна вірогідність ризику цільового вузла в AG буде знижена, а індекс безпеки поточної мережі також буде знижений. Індекс безпеки розраховується з числа різних шляхів і загальної довжини шляхів в AG.

Попередження завищених потреб системи засноване на динамічній реконфігурації. Допускається, що сигнали, завищені NIDS, є основними оповіщеннями. Підвищене попередження означає, що одна з основних вразливостей експлуатується. Для того, щоб захистити систему, потрібно застосувати статичну реконфігурацію, яка застосовується для захисту системи від компрометації. Замість того, щоб шукати критичний шлях в AG, система повинна відповідати попередженням вузла в AG і застосовувати реконфігурацію, використовуючи той же підхід в статичній реконфігурації.

Ще один момент в цьому дослідженні – оцінка стратегії реконфігурації. Знижена ймовірність ризику цільового вузла в AG покаже значення індексу безпеки поточної мережі, яка включає в себе кількість шляхів атаки і загальну вагу всіх доріжок, що впливають на число нормальних послуг і затримку відповіді на звичайний трафік.

Контрзаходом називається дія або ряд дій, які присікають атаки, в яких він може змінити мережні конфігурації і політику трафіку. Далі розглянемо розгортання пристрою захисту (наприклад, IPS), який вводить послідовність дій для забезпечення безпеки в хмарних середовищах віртуальних мереж. Коли атака або програмне забезпечення були виявлені, один (або набір) ефективних контрзаходів повинен бути обраний. Необхідно врахувати атрибути, такі як вартість, час розгортання, а також потенціал для зниження продуктивності або доступності системних ресурсів. За допомогою атаки графа моделювання поведінки нападників і вибору контрзаходів були добре вивчені [6]. Загалом, є багато контрзаходів, які можуть бути застосовані до системи хмара, в залежності від наявних пристроїв безпеки, які можуть бути застосовані.

Кілька загальних віртуальних мереж на основі контрзаходів перераховані в таблиці 1. Стратегії реконфігурації мережі включають в себе кілька рівнів дій з боку рівня-2, в верхніх рівнях. В рівні-2, віртуальні мости (в тому числі тунелі, які можуть бути встановлені між двома мостами) і віртуальні локальні мережі є основними компонентами в системі хмари віртуальної мережі для підключення двох віртуальних машин. Віртуальний міст є об'єктом, який надає віртуальні інтерфейси (VIFs). Віртуальні машини на різних ізольованих мостах на рівні 2. Відеосюжети на той же віртуальний міст, але з різними тегами VLAN не можуть взаємодіяти один з одним безпосередньо. На основі цього рівня 2 ізоляції, модуль реконфігурації може розгорнути зміни конфігурації мережі рівня 2 для ізоляції підозрілих віртуальних машин. В результаті, цей контрзахід роз'єднує шлях атаки в графі атак і змушує атакуючого

досліджувати альтернативний шлях атаки. Рівень-3 це інший спосіб реконфігурації, щоб від'єднати шлях атаки. За допомогою мережевого контролера таблиця витрат на кожному перемикачі OpenFlow (наприклад, як програмне забезпечення і фізичні комутатори) може бути модифікована, щоб змінити топологію мережі. Аналогічним чином, заходи протидії верхнього рівня такі, як порт зміни / блокування, протоколів фільтрації додатків, DPI і т.д., можуть бути розгорнуті.

Таблиця 1

Можливі типи контрзаходів

Рівні	**Контрзахід**
рівень-2	зміна mac-адресу
рівень -2	конфігурації комутатора
рівень-2 або 3	перенаправлення трафіку
рівень-2 або 3	ізоляція трафіку
рівень-3	зміна ip-адреси
рівень-3	зміна топології мережі
рівень-4	зміна / порт блок
додаток	аналіз пакетів (dpi)
додаток	патч програмного забезпечення
додаток	карантин
додаток/система	диверсифікація програмного забезпечення
додаток/система	оновлення програмного забезпечення
система	введення в віртуальні машини
система	створення правил фільтрації
...	...

Слід зазначити, що використання мережі реконфігурації в нижньому рівні має перевагу, тому що додатки верхнього рівня будуть відчувати мінімальний вплив. Такий підхід можливий тільки при використанні підходу програмного забезпечення для автоматизації перемикання конфігурації в динамічному мережному середовищі. Контрзаходи (такі як ізоляція трафіку) можуть бути реалізовані шляхом використання трафіку технічних можливостей OpenFlow комутаторів для обмеження потужності і переналаштування віртуальної мережі для підозрілого потоку. Коли підозрілу активність, таку як сканування портів виявлено, важливо визначити чи є у зловмисна активність, чи ні. Наприклад, зловмисники можуть навмисно приховувати свою поведінку сканування для запобігання мережевих IDS від визначення їх дії. У цій ситуації, змінюючи конфігурацію мережі, треба змусити атакуючого виконувати більше досліджень і, в свою чергу, змусити показати свою активність.

Більшість графів атак прийняті евристичним методом і до сих пір не існує алгоритму, який може оцінити всі можливі стани в поліноміальній складності. Крім того, є дуже обмежена кількість досліджень, проведених на графі атак динамічного оновлення.

У даній статті розроблена розподілена генерація графа атак і підхід до оцінки. Блок-схема способу показана на рис.3 .

Перший крок полягає у зборі інформації, що зроблено аналізатором атаки системи. Він збирає інформацію мережного підключення з мережним контролером SDN, показує стан установки програми та послуги в кожному пристрої за допомогою сканерів (наприклад, Nmap або уразливості сканера), а також відображає правила політики брандмауера з IPS або пристроїв брандмауера.

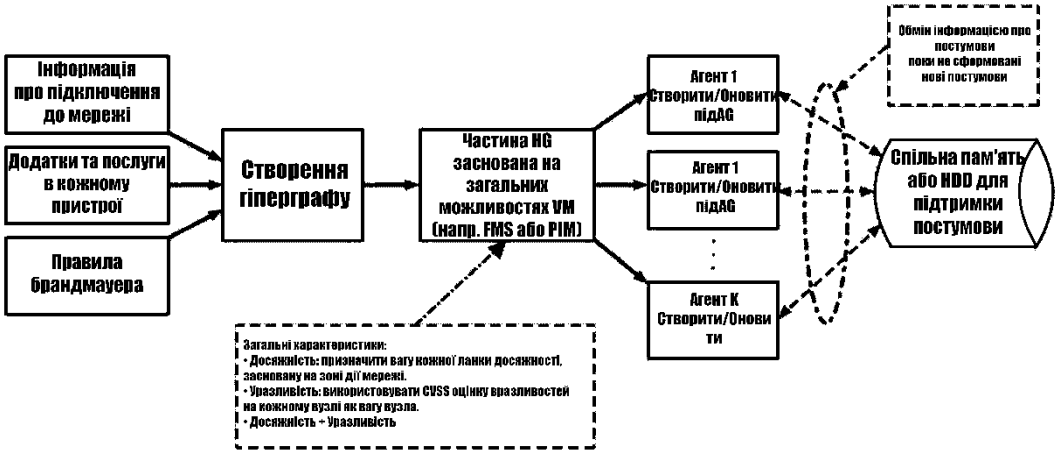


Рис. 3. Розподілені атаки побудови потоку графа

Наступним кроком аналізатор атаки створює досяжності гіперграфа (HG) на основі зібраної інформації з попереднього кроку. Гіпер-ребро в HG представляє стан служби, здатність між двома або більше віртуальними машинами або пристроями. Одне гіпер-ребро може бути підключене до того ж порту на декількох різних віртуальних машинах.

Третім кроком є фаза розділу гіперграфа. Модуль розділу HG в аналізаторі атак застосовує обраний алгоритм розділу HG для розбиття гіперграфу на кілька дрібніших кластерів. Алгоритм розділів знайде міні-розрізи в HG і відокремить HG на безліч K, менше HG з мінімальними розрізами між кластерами. Розділ гіперграфа є добре відомою NP-важкою задачею. Пропонується використання існуючих рішень для вирішення цієї проблеми, таких як FMS і PLM.

Коли граф атак застосовує алгоритм розділу, він може призначити вагу кожного вузла і краю в гіперграфі.

На підставі кількості кластерів після виконання алгоритму розділу HG, аналізатор атак розгортає однакову кількість агентів побудови графів атак у віртуальній мережі і поширює інформацію про суб-гіперграфу від відповідного кластера до кожного агента для побудови графіка суб-атаки. Кожен агент починає з пошуку необхідної інформації в суб-гіперграфі. З цією інформацією кожен агент починає створювати суб-граф атак. При створенні графа атак кожен агент буде генерувати нові правила виведення в конструкцію двигуна графа атак. Новий пост-стан представляє інформацію або привілеї для посилення атакуючого. Зловмисник може використовувати цю вигоду, щоб скористуватися іншими можливостями уразливості в системі. Тобто потрібно стежити за всіма новими пост-умовами, отриманими від кожного агента і дозволити кожному агенту, отримати нові пост-умови від інших агентів.

Використовується в даному випадку загальна пам'ять і розподілений набір даних (RDD) для підтримки цих нових пост-умов. Коли з'являється новий пост-стан, агент буде записувати інформацію в спільно використовувану пам'ять. Коли агент закінчив, він буде читати нові пост-умови із загальної пам'яті і викличе функцію оновлення графа атак на основі цих нових пост-умов.

Функція поновлення намагатиметься відповідати новій пост-умові з кожною попередньою умовою в поточному суб-AG. Якщо є збіг і змога використовувати певну вразливість в вузлі, агент буде оновлювати суб-AG. Якщо зміст загальної пам'яті порожній і немає будь-якого нового пост-стану генерованого усіма агентами, кожен агент буде ставити себе в пасивному режимі і зупинить алгоритм створення.

Розроблений веб-інструмент для запуску розподіленої атаки графу, дозволяє провести експеримент. Таблиця 2 показує час роботи 8 різних тестів з різними розмірами мережі (кількість вузлів) і різним числом агентів.

Таблиця 2

Час роботи розподіленої атаки

вузли	ребра	AG-вузли	AG-ребра	1 агент	2 агент	3 агент
11	52	39	53	2.3	4.2	6.4
32	152	55	82	4.2	6.3	8.3
74	369	102	224	7.2	10.2	13.5
124	1280	234	522	13.3	19.2	20.1
165	2321	398	790	28.4	26.3	25.9
199	3420	792	1323	38.2	27.3	26.1
230	5892	1342	2319	63.2	48.5	40.3
360	6203	2021	3080	83.9	63.2	55.4

Коли кількість вузлів в гіперграфі мала, продуктивність одного агента краще, ніж продуктивність декількох агентів, тому що відбуваються накладні витрати шляхом обміну пост-умов інформації між агентами і поновлення графа атак. Коли розмір мережі збільшується до 165, продуктивність генерації графа атак з використанням декількох агентів краще, ніж продуктивність одного і того ж процесу в якості єдиного агента. Коли число вузлів збільшене, більше агентів, тому і краща продуктивність.

Таким чином, можна зробити висновок з результатів експерименту, що в той час як число вузлів збільшено в гіперграфі, продуктивність з більшою кількістю агентів краща, ніж продуктивність з меншим числом агентів. Перевага розподіленого алгоритму стає очевидною у великій мережі.

Результати, висновки і рекомендації

Представлений граф атак на основі аналізу рамок безпеки/уразливості для відстеження вразливостей в кожній віртуальній машині (VM) і залежність уразливості серед віртуальних машин запропонований для вирішення проблеми масштабованості з використанням графів атак у великому центрі даних. З цією метою розроблено SDN на основі моделі реконфігурації. Представлена модель враховує безпеку і QoS для кінцевих користувачів. Це підвищує безпеку віртуального мережевого середовища з мінімальним рівнем втручання до нормального трафіку користувача. У статті досягнуто цієї мети, використовуючи ковзаючі методи оборони цільових, які змушують нападників постійно дбати лише про свої цілі, стримуванням та усуненням атак без переривання нормального мережевого трафіку. Це дозволить уникнути постійних загроз зі сторони нападника. Результатами запропонованої системи є:

• на основі віртуальної мережі створюється, в режимі реального часу, модель реконфігурації мережі для SDN, щоб реагувати на будь-які аномальні події в мережі;

• в даній моделі стратегії реконфігурації роблять віртуальне мережне середовище кожного користувача більш безпечним, зберігаючи при цьому їх функціональність мережі так само, як і вимоги SLA;

• шляхом класифікації уразливості в мережі дана система здатна розділити мережу на кілька логічно виділених зон з різними властивостями безпеки, і звести до мінімуму залежність уразливості між зонами;

• запропонована система застосовує новий розподілений граф атак на основі аналітичної моделі для відстеження вразливостей в кожній зоні, обмежуючи зв'язність вразливостей між зонами.

Література

1. Барсков А. SDN: кому и зачем это надо? [Електронний ресурс] / А. Барсков // Журнал сетевых решений. – 2012. – № 12.Режим доступу: http://www.osp.ru/lan/2012/12/13033012/

2. A Framework for IP Based Virtual Private Networks [Электронный документ] / B. Gleeson, A. Lin, J. Heinanen. — http://www.ietf.org/rfc/rfc2764.txt

3. Олифер В. Г., Олифер Н. А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. — СПб.: Питер, 2001. — 672 с.

4. Gillam, Lee. Cloud Computing: Principles, Systems and Applications / Nick Antonopoulos, Lee Gillam. — L.: Springer, 2010. — 379 p. — (Computer Communications and Networks). — ISBN 9781849962407.

5 Ammann P., Wijesekera D., Kaushik S. Scalable Graph-Based Network Vulnerability Analysis // Proc. of the 9th ACM Conference on Computer and Communications Security, New York: ACM Press. – 2002. – P. 217–224.

6. C. Tankard, “Advanced persistent threats and how to monitor and deter them,” Network Security, vol. 2011, no. 8, pp. 16–19, Aug. 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1353485811700861

Надійшла 22.07.2016 р. Рецензент: д.т.н., проф. Шелест М. Є.