УДК 004.77:004.424

Василенко В.В., аспірант

РЕКОНФІГУРАЦІЇ МОДЕЛІ МЕРЕЖІ ДЛЯ МАСШТАБОВАНОСТІ SDN

Vasylenko V.V. Reconfiguration of network model for SDN scalability.

This article describes how SDN can be used in the context of cloud computing to dynamically change the basic network infrastructure to support the needs of security, balancing the required QoS for clouds users.

The question of scalability using graph attacks by analyzing the security model in the software and virtualized network environment. Introduced graph of attacks by analyzing the frames security / vulnerability for tracking of vulnerabilities for each virtual machine (VM) vulnerability and dependence among the virtual machines. To solve the problem of scalability using graphs of attacks in a large data center, is designed SDN based on the reconfiguration model. The model takes into account the security and QoS to end users. This increases the security of a virtual network environment with minimal interference to the normal user traffic. The article is achieved this goal by using methods of moving target defense, forcing the attackers always worry only about their goals, deterring and elimination of attacks without interrupting normal network traffic. This will avoid persistent threats from the side of the attacker.

Key words: Keywords: cloud, data security, virtualization, data traffic, attack, attack graph, hypervisor, SLA, SDN.

Василенко В.В. Реконфігурації моделі мережі для масштабованості SDN.

У даній статті описується, 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. Представлений граф атак на основі аналізу рамок безпеки/уразливості для відстеження вразливостей в кожній віртуальній машині (VM) і залежності уразливості серед віртуальних машин.

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

Василенко В.В. Реконфигурации модели сети для масштабируемости SDN.

В данной статье описывается, как SDN может быть применен в контексте облачных вычислений, чтобы динамически изменять базовую сетевую инфраструктуру с целью поддержки потребностей в области безопасности, уравновешивая необходимые QoS для пользователей облака. Представлен граф атак на основе анализа рамок безопасности / уязвимости для отслеживания уязвимостей в каждой виртуальной машине (VM) и зависимости уязвимости среди виртуальных машин.

Ключевые слова: облачные технологии, безопасность данных, виртуализация, трафик данных, атака, граф атак, гипервизор, SLA, SDN.

Вступ

Хмарні обчислення стали новою обчислювальною парадигмою для хостингу і надання послуг через Інтернет протягом останніх кількох років. Все більше компаній малого і середнього бізнесу почали користуватися хмарою, щоб отримати швидкий доступ до кращих бізнес-додатків і збільшити ресурси інфраструктури. Проте недавні дослідження показали, що занепокоєння безпекою хмари перешкоджають міграції клієнтів до даного рішення. Причиною коливань клієнта або відмовою переходу до віртуалізації, як ключової технології хмарних обчислень, є питання безпеки даних. Віртуалізації поверхні атаки для хмари, такі як гіпервізора вразливостей [1], втечі VM [2], викрадення VM [3, 4] і порушення ізоляції VM і віртуальної мережі [5, 6].

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 вразливостей атак і зведення до мінімуму наслідків порушення безпеки для користувачів хмари.

На щастя, з недавньою появою Software-Defined Networking (SDN), хмара-провайдер здатна ефективно виявляти маршрутні підозрілі і шкідливі потоки від користувачів в своїх віртуальних мережах [7] і зручно створювати додатки безпеки в них [1] 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 описати, 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 для того, щоб проілюструвати всі можливі шляхи багатоступінчастої, мульти-хост-атаки, які мають вирішальне значення для розуміння загроз і застосування відповідних контрзаходів. Проте, добре відомий стан проблем графа атак [8] робить його не в змозі змоделювати сценарії атак в великих масштабах мережі, таких 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. Намаганням досягнення цієї мети є використання ковзаючих методів оборони, стримування та усунення атак без переривання регулярного мережного трафіку [9].

Внесок даної системи представлений в такий спосіб:

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

• Пропонується модель пасивного відгуку мережних подій, щоб змінити конфігурацію мережі з використанням методів оборони рухомих мішеней.

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

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

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

Постановка завдання. Граф атак і дерево атак повинні вирішувати питання масштабованості на різних етапах: формування, подання, оцінки і модифікації [10]. Розмір мережі може збільшуватися (наприклад, зростає кількість 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. Таким чином, 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 [11]. Тим не менше, на сьогоднішній день немає наукових робіт, які використовують граф атак в дуже динамічному віртуалізованому і перебудованому мережному середовищі, і які розглядали 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 даних навколишнього середовища. Оскільки граф атак зберігає кореляції між загрозами, то й 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;

• під час використання груп взаємопов'язаних віртуальних машин з найменшими мережами і додатками підключення до інших груп.

Мета та задачі дослідження. Пропонується комплексне рішення для вирішення проблеми мережі і кластеризації графа атак для того, щоб обмежити розмір графа атак і зменшити залежність уразливості між віртуальними машинами, де VM угруповання засноване на типі уразливості і значимості в кожній віртуальній машині. Віртуальні машини з аналогічною уразливостю на тому ж рівні тяжкості, розглядаються як група безпеки. Після того, як угруповання засноване на уразливості, є можливість мати більш чітке уявлення про ситуацію в області безпеки системи і можливість дізнатися, які вразливі додатки впливають на безпеку системи. Обчислюється оцінка уразливості з базовим рахунком системи CVSS [12] для кожної віртуальної машини, за допомогою експоненційної середньої, для вимірювання всіх вразливостей кожної віртуальної машини (VM Індекс Безпеки), так що оцінка буде дорівнювати найсерйознішій уразливості системи. З тієї ж самої групи безпеки береться середня експонента уразливості від всіх віртуальних машин в одній групі (група індексів VM Security) і привласнюється їй агреговані метрики для вимірювання рівня безпеки цієї групи.

Крім того, угрупування VM на основі безпеки, можна розділити мережу по зв'язності і досяжності віртуальних машин. Розмір атакуючого графа впливає не тільки на уразливості в системі, а й на підключення до мережі кожної VM. Якщо можна згрупувати віртуальні машини по зв'язному графу і розбити мережі відповідно до найменшого зв'язку кластера, то можна обмежити розмір графа атак, а також звести до мінімуму залежність уразливості серед графів атак. Завдання розбиття графа полягає в розділенні вершин в ряді груп заданого розміру таким чином, щоб число ребер, що лежать між групами, були мінімальні. Число ребер, які працюють між кластерами називаються розмір відрізка [13]. Мережа кластеризації є однією з основних задач у багатьох областях науки і техніки. Ці методи мають тенденцію групувати мережі таким чином, що безліч ребер знаходяться всередині кожного кластера і кілька ребер між кластерами [14]. Багато алгоритмів були запропоновані практиками в різних областях, включаючи інформатику та фізику. Успішні приклади Мін-Макс відрізка [15] і нормованого відрізка [16], а також алгоритми, засновані на модульності [17, 18].

Пропонується багатоцільовий підхід оптимізації для отримання оптимальних кластерів графа атак, враховуючи обидва чинники безпеки груп VM і досяжності між групами. Подібна вразливість групи здатна зменшити надмірність вузла шляху в графі атак так, щоб розмір графа атак міг бути зменшений шляхом видалення надмірності. Найменші з'єднання між групами можуть зменшити залежність в обох досяжностях і уразливостях зв'язку, щоб кластером графа атак було легше управляти і регулювати.

1. Моделі графа атак і загрози мережі

Для того, щоб краще описати проблему і пропоноване рішення, визначаємо кілька моделей.

Модель загроз. У даному прикладі припустимо, що гіпервізор є надійним і захищеним, тобто гіпервізор правильно ізолює ресурси та умови для працюючих віртуальних машин. Низькорівнева оболонка захищена від будь-яких експлойтів запущених зловмисником і механізм виявлення встановлений гіпервізором є невидимим для атакуючого. Припустимо також, що користувачі мають можливість встановлювати уразливе програмне забезпечення і виконувати будь-які шкідливі програми або шкідливий код в їх VM. Системний адміністратор не може виправити програмне забезпечення або видалити шкідливий код без користувальницької угоди. Однак, CSP дозволяє блокувати трафік виданий такими процесами.

Мережеві моделі. Для того, щоб розділити мережі для отримання прийнятного розміру контрольованої мережі, що підлягають моніторингу, створюється мережева модель для віртуальної мережі в хмарній системі. Мережева модель не тільки здатна обмежити розмір графів атак, але також може забезпечити модель оцінки стратегії реконфігурації.

Моделюємо мережу як орієнтований граф G = (V, E), де V це безліч хостів, E це безліч посилань. Припустимо, що є потік з джерелом S і призначення D (S, D ∈ V), а тривалість потоку можна розділити на кілька інтервалів. Мета реконфігурації мережі – знайти маршрут між S і D, який задовольняє наступні обмеження для кожного інтервалу тривалості:

• обмеження пропускної здатності: новий маршрут не повинен включати ці посилання або вузли, які і без того перевантажені, або ті вузли або посилання, які не мають вимоги до пропускної здатності потоку;

• QoS обмеження: мутовані маршрути повинні підтримувати необхідну якість, такі як обмеження затримки або кількість перельотів;

• обмеження безпеки: індекс безпеки нового обраного маршруту, повинен бути оптимальний або неоптимальний шлях;

• обмеження витрат: якщо існує кілька маршрутів, вартість реконфігурації нового маршруту повинна бути не менше.

Модель графа атак. Граф атак є модельована парадигма, яка має вирішальне значення для розуміння загроз і вирішення їх, та підбору відповідних контрзаходів [40]. Граф атак корисний для виявлення потенційних загроз, можливих атак і відомих вразливостей в системі хмара. Оскільки граф атак надає інформацію про всі відомі вразливості в системі та інформацію про зв'язності, отримується ціла картина ситуацій в сфері безпеки системи, де є можливість спрогнозувати можливі загрози і атаки шляхом зіставлення виявлених подій або заходів.

Граф атак складається з вершин, що представляють умови і дії. Умови є конфігурації системи або привілеї доступу, які повинні бути істинними для того, щоб використовувати будь-яку вразливість. Дії – ті кроки, які зловмисник виконує для того, щоб поставити під загрозу віртуальну машину. Дії зловмисника залежать від наявності одного або декількох умов. Ребра в графі атаки це з'єднання вузлів, і безліч ребер, які позначають шлях від початкового вузла до кінцевого вузла атаки.

Визначимо граф атак AG = (AV, AE) де AV = NC ∪ ND ∪NR і

AE = Epre ∪ Epost. AV це безліч вершин, які включають в себе три типи вузлів: вузол кон'юнкції NC для подання використання, вузол диз'юнкції ND для позначення результату експлуатації і вузла NR корінь для показу початкової стадії сценарію атаки. AE позначає безліч спрямованих ребер.

e ∈ Epre ⊆ ND × NC показує, що ND повинен бути виконаний, щоб досягти NC. Ребро e ∈ Epost ⊆ NC × ND означає, що ND може бути отриманий, якщо NC виконано.

В роботі застосовано вимір ймовірності ризику на кожному графіку атаки на основі моделі Байєса ймовірнісної мережі для розрахунку ймовірності умовного ризику для кожного вузла уразливості і сукупної ймовірності ризику для цільового вузла в нападі графа. Імовірність ризику заснована на CVSS балах уразливості, знайденої в контрольованій мережі і віртуальних машинах. Порівняємо кожну атаку шляхом усереднення ймовірності кумулятивного ризику всіх цільових вузлів в контрольованій мережі. Цільові вузли можуть бути обрані випадковим чином з мережі або вибрані з віртуальних машин.

2. Проектування системи

Представимо першу системну архітектуру запропонованого дизайну та докладний опис його компонентів.

Загальна архітектура. Пропонована структура показана на рис. 1. Є чотири етапи: побудова графа атак, граф атак кластеризація, оцінка безпеки і реконфігурація.

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 того, 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.

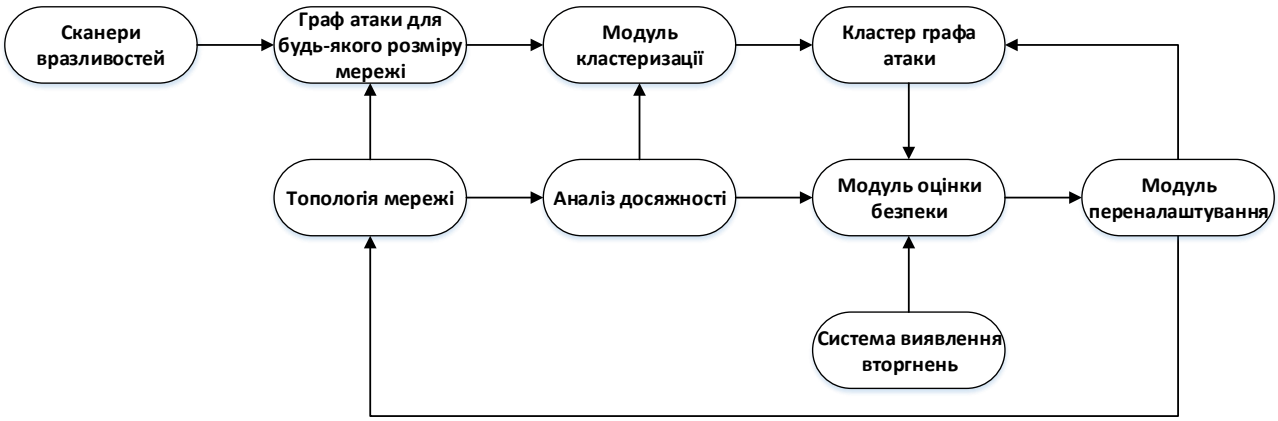

Рис. 1. Конфігурація моделі

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.

Стратегії реконфігурації. 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 важким. Реконфігурація може бути зробленою за допомогою мережевого контролера, що може змінити: передачу потоку, перенаправити потік, відобразити потік, 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. 2 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, стратегія 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-адреси, 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 реконфігурація.

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.

Динамічна реконфігурація заснована на попереджені завищених потреб системи. Можна припустити, що сигнали, підняті NIDS, є основними оповіщення. 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.

Оцінка стратегії реконфігурації ще один нюанс в цьому дослідженні. 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.

SDN на основі контрзаходів і реконфігурації. 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

[9]. Загалом, є багато контрзаходів, які можуть бути застосовані до системи хмара, в залежності від наявних пристроїв безпеки, які можуть бути використані.

Кілька загальних віртуальних мереж на основі контрзаходів перераховані в табл. 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 ре конфігурації,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.

Табл. 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 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.

3. Побудова розподіленого графа атак

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. 2.

Рис. 2. Розподілені атаки побудови потоку графа

Збір інформації. Перший крок робить аналізатор атаки системи. Він збирає інформацію мережного підключення з мережним контролером SDN, збирає стан установки програми та послуги в кожному пристрої за допомогою сканерів (наприклад, Nmap або уразливості сканера), а також збирає правила політики брандмауера з IPS або пристроїв брандмауера.

Етап створення гіперграфа. 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.

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 відомим NP-важкою задачею. Ми використовуємо існуючі рішення для вирішення цієї проблеми, такі як FMS і PLM.

Коли граф атак застосовує алгоритм розділу, він може призначити вагу кожного вузла і краю в гіперграфі. Призначення ваги пов'язане з тим, як алгоритм розділить гіперграф. Три варіанти алгоритму, щоб призначити вагу:

- досяжності: призначити вагу для кожної ланки досяжності на основі зони дії мережі;
- уразливості: використання CVSS оцінки вразливостей на кожному вузлі;
- розглянути доступність і вразливість.

Розподілені фази графа атак. 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 використовувати інші можливі уразливості в системі. Таким чином, необхідно 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 закінчує, 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.

Висновки

В даній статті досліджено питання масштабованості при використанні графа атак на основі аналізу моделі безпеки в програмному і віртуалізованому мережному середовищі. Представлений граф атак на основі аналізу рамок безпеки/уразливості для відстеження вразливостей в кожній віртуальній машині (VM) і залежність уразливості серед віртуальних машин 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, що підвищує безпеку віртуального мережевого середовища з мінімальним рівнем втручання до нормального трафіку користувача, використовуючи ковзаючі методи оборони цільових, які змушують нападників постійно дбати лише про свої цілі, стримуванням та усуненням атак без переривання нормального мережевого трафіку. Самий такий спосіб дозволить уникнути постійних загроз зі сторони нападника.

Література

1. D. Perez-Botero, J. Szefer, and R. B. Lee. Characterizing hypervisor vulnerabilities in cloud computing servers. [Електронний ресурс] / D. Perez-Botero, J. Szefer, and R. B. Lee.// in Proceedings of the 2013 International Workshop on Security in Cloud Computing, ser. Cloud Computing '13. New York, NY, USA: ACM. – 2013. – pp. 3–10. – Режим доступу : http://doi.acm.org/10.1145/2484402.2484406

2. K. Owens. Securing virtual compute infrastructure in the cloud. [Електронний ресурс] / K. Owens // Savvis Communications Corp, White paper. – 2009. – Режим доступу : http://viewer.media.bitpipe.com/1018468865 999/1296679360 880/Securing-Virtual-Compute-Infrastructure-in-the-Cloud.pdf

3. A. Koto, H. Yamada, K. Ohmura, and K. Kono. Towards unobtrusive VM live migration for cloud computing platforms. [Електронний ресурс] / A. Koto, H. Yamada, K. Ohmura, and K. Kono // in Proceedings of the Asia-Pacific Workshop on Systems, ser. APSYS '12. New York, NY, USA: ACM – 2012. – pp. 7:1–7:6. – Режим доступу : http://doi.acm.org/10.1145/2349896.2349903

4. E. Kotsovinos. Virtualization: Blessing or curse? [Електронний ресурс] / E. Kotsovinos // Commun. ACM, vol. 54, no. 1, pp. 61–65, Jan. 2011.. Режим доступу: http://doi.acm.org/10.1145/1866739.

5. Y.-L. Huang, B. Chen, M.-W. Shih, and C.-Y. Lai. Security impacts of virtualization on a network testbed. in 2012 IEEE Sixth International Conference on Software Security and Reliability (SERE), Jun. 2012, pp. 71–77.

6. T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. [Електронний ресурс] / T. Ristenpart, E. Tromer, H. Shacham, and S. Savage // . in Proceedings of the 16th ACM conference on Computer and communications security, ser. CCS '09. New York, NY, USA: ACM, 2009, pp. 199–212. – Режим доступу : http://doi.acm.org.ezproxy1.lib.asu.edu/10.1145/1653662.1653687

7. M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat. Hedera: Dynamic flow scheduling for data center networks. [Електронний ресурс] / M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat // in NSDI, vol. 10, 2010, pp. 19–19.. – Режим доступу : https://www.usenix.org/legacy/event/nsdi10/tech/ full papers/al-fares.pdf

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

9. C. Tankard. Advanced persistent threats and how to monitor and deter them. [Електронний ресурс] / C. Tankard // . Network Security, vol. 2011, no. 8, pp. 16–19, Aug. 2011.– Режим доступу : http://www.sciencedirect.com/science/article/pii/S1353485811700861

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

11. O. Sheyner, J. Haines, S. Jha, R. Lippmann, and J. M.Wing, "Automated generation and analysis of attack graphs," in 2002 IEEE Symposium on Security and Privacy, 2002. Proceedings. IEEE, 2002, pp. 273– 284.

12. P. Mell, K. Scarfone, and S. Romanosky, "Common vulnerability scoring system (CVSS)," http://www.first.org/cvss/cvss-guide.html, May 2010.

13. S. Fortunato Community detection in graphs. [Електронний ресурс] / . S. Fortunato // Physics Reports, vol. 486, no. 3–5, pp. 75–174, Feb. 2010. – Режим доступу : http://www.sciencedirect.com/science/ article/pii/S0370157309002841

14. DanforthM. Models for Threat Assessment in Networks. – Режим доступу : http://www.cs.ucdavis.edu/research/tech- reports/2006/CSE-2006-13.pdf

15 C. Ding, X. He, H. Zha, M. Gu, and H. Simon, "A min-max cut algorithm for graph partitioning and data clustering," in ICDM 2001, Proceedings IEEE International Conference on Data Mining, 2001, 2001, pp. 107–114.

16. J. Shi and J. Malik, "Normalized cuts and image segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 888–905, Aug. 2000.

17. R. Guimer`a and L. A. Nunes Amaral. Functional cartography of complex metabolic networks. [Електронний ресурс] / R. Guimer`a and L. A. Nunes Amaral //. Nature, vol. 433, no. 7028, pp. 895–900, Feb. 2005 – Режим доступу: http://www.nature.com/nature/journal/v433/n7028/abs/nature03288.html

Автор статті

Василенко Володимир Вікторович - аспірант кафедри Комп'ютерних наук та інформаційних технологій, Державний університет телекомунікацій, Київ, Україна. Тел.: +38 063 717 67 94.
E-mail: oknelisavvova172@gmail.com

Authors of the article

Vasylenko Volodymyr Viktorovych – post-graduate student of Department of Computer Science and information technology, State University of Telecommunications, Kyiv, Ukraine Tel .: + 38 063 717 67 94.
E-mail: oknelisavvova172@gmail.com

Дата надходження в редакцію: 11.08.2016 р. Рецензент: д.т.н., проф. В.В. Вишнівський