УДК 519.254

В.І. Михайлів

ДВНЗ «Криворізький національний університет»

ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДЛЯ ЗАХИСТУ ДАНИХ У ХМАРНИХ ОБЧИСЛЮВАЛЬНИХ СИСТЕМАХ

Розглянуто основні експериментальні дослідження ефективності інформаційної технології захисту хмарних обчислень на основі тестів, що враховують точність, повноту і швидкодію системи захисту на загрози безпеки. Представлено і проаналізовано результати тестових запусків дослідницької системи для проведення мережевих атак на хмарні обчислювальні системи. Розроблено технологію захисту даних, що виявляє небезпечну мережеву активність та спрямована на визначення різних типів мережевих атак, які будуть впливати на цілісність ресурсів та послуг у середовищі хмарних обчислень.

Ключові слова: *хмарні обчислювальні системи, захист даних, мережева атака, експериментальні дослідження.*

Рассмотрены основные экспериментальные исследования эффективности информационной технологии защиты облачных вычислений на основе тестов, учитывающих точность, полноту и быстродействие системы защиты на угрозы безопасности. Представлены и проанализированы результаты тестовых запусков исследовательской системы для проведения сетевых атак на облачные вычислительные системы. Разработана технология защиты данных, которая обнаруживает опасную сетевую активность и направлена на определение различных типов сетевых атак, которые будут влиять на целостность ресурсов и услуг в среде облачных вычислений.

Ключевые слова: *облачные вычислительные системы, защита данных, сетевая атака, экспериментальные исследования.*

The basic experimental research on the effectiveness of information security technology for cloud computing based on tests is considered, taking into account the accuracy, completeness and speed of the protection system for security threats. Presented and analyzed the results of test runs research system for network attacks on cloud computing. The technology of data protection is developed, that detects dangerous network activity and aims to identify the different types of network attacks, which will affect the integrity of the services and resources in a cloud computing environment.

Keywords: *cloud computing, data protection, network attack, experimental research***.**

Постановка проблеми. Задача визначення аномальної мережевої активності є актуальною і полягає в розробці нейромережевої технології моніторингу трафіка, що використовує апарат штучних нейронних мереж з метою підвищення рівня інформаційної безпеки хмарних обчислювальних систем, а також експериментальної реалізації програмних модулів системи захисту. Сучасні вимоги до інформаційної безпеки у хмарних обчислювальних системах передбачають створення спеціальних програмних сервісів для виявлення мережевих атак і забезпечення безпеки комунікацій та захисту даних у критично важливих інформаційних ресурсах**.** Незважаючи на наявний прогрес у створенні програмних засобів інформаційної безпеки, необхідно зазначити нагальну необхідність у захисті хмарних обчислень, що мають специфічні особливості в порівнянні зі звичайними обчисленнями. Питання вимагає ретельного опрацювання особливостей хмарних обчислень з метою виявлення вимог до систем їх інформаційного захисту.

Аналіз останніх досліджень і публікацій. В даний час розробляється велика кількість різних технологій захисту хмарних обчислювальних систем [1; 7—9]. До критеріїв оцінки їх якості можна віднести здатність виявляти нові атаки, точність і швидкість роботи, здатність працювати з великими обсягами навчальних вибірок. До їх недоліків можна віднести уразливість до нових атак, низьку точність і швидкість роботи. Одним з ефективних математичних апаратів для побудови систем безпеки є апарат штучних нейронних мереж [2; 7]. У зв’язку з необхідністю забезпечення інформаційної безпеки хмарних обчислень спостерігається підвищений інтерес до нейромережевих технологій, які показують досить високу повноту і точність класифікації мережевих атак. Необхідним етапом попередньої обробки мережевих пакетів є їх приведення до найбільш інформативної форми за рахунок видалення зайвих ознак і зменшення обсягу навчальної вибірки. У процесі експериментальних досліджень з розробки системи захисту було вивчено і використано інформацію, представлену у роботах [1—9].

Постановка завдання. Завдання полягає в розробці засобів підвищення безпеки хмарних обчислювальних систем. Для тестування розроблених засобів захисту було використано базу даних KDD-99 [2], що містить близько п’яти мільйонів записів про з’єднання. Кожен запис у цій базі являє собою образ мережевого з’єднання. З’єднання —

послідовність TCP пакетів за деякий кінцевий час, моменти початку і завершення якого чітко визначені, протягом якого дані передаються від IP-адреси джерела на IP-адресу отримувача (і у зворотному напрямку), використовуючи деякий певний протокол. Окремий запис складається із 100 байтів інформації, включає 41 параметр (ознаку) мережевого трафіка і промаркований як «атака» (дія, що призводить до порушення цілісності та конфіденційності інформації) чи «не атака» (норма, штатний режим роботи). Наприклад, перший параметр визначає тривалість з'єднання, другий — вказує використовуваний протокол, третій — цільову службу. Вони можуть бути розділені на кілька груп:

1. Ознаки, інформація про які може бути виділена із заголовка мережевого пакета (номери 1—9).
2. Ознаки, що виділяються з інформаційної частини пакета на підставі експертних знань, або як інформація про стан контрольованого вузла (10—22).
3. Ознаки, значення яких обчислюються на основі статистики за минулі дві секунди мережевої активності (номери 23—31).
4. Ознаки, значення яких обчислюються на основі статистики за останні 100 з'єднань (номера 32—41).

До мережевих ознак можуть бути віднесені ознаки, що входять у першу, третю та четверту групи. Ознаки першої групи можуть бути виділені з мережевого пакета, ознаки третьої і четвертої груп є результатом обробки статистики.

Розроблені засоби реалізують витяг немережевих ознак (ознаки другої групи) за принципом аналізу активності файлової системи, а також аналізу інформації статистики використання операційної системи. Для аналізу активності файлової системи застосовано спеціальні засоби розробника, що входять до складу операційної системи Linux з версією ядра не нижче 2.6, за допомогою яких можна відстежувати такі події як: доступ до певної директорії, створення файла, видалення файла, створення директорії, запуск програми і т . д.

Усього в базі KDD-99 представлено 22 типи атак. Атаки розбито на наступні групи:

1. DoS — відмова в обслуговуванні, характеризується генерацією великого обсягу трафіка, що призводить до перевантаження і блокування сервера.
2. U2R передбачає отримання зареєстрованим користувачем привілеїв локального суперкористувача (адміністратора).
3. R2L характеризується отриманням доступу незареєстрованого користувача до комп'ютера з боку віддаленої машини.

4. Probe полягає у скануванні портів з метою отримання конфіденційної інформації.

Виклад основного матеріалу дослідження. У ході моделювання проводилися наступні експерименти:

1. Видалення неінформативних ознак і стиснення простору ознак з використанням методу головних компонент і штучної нейронної мережі. Проведені експерименти показали, що, використовуючи апарат штучних нейронних мереж, можна значно стиснути простір ознак без значної втрати якісних характеристик, що позитивно позначиться на швидкості навчання штучних нейронних мереж і знизить витрати апаратних ресурсів. На рис. 1 представлено залежність кількості інформації від числа головних компонент.

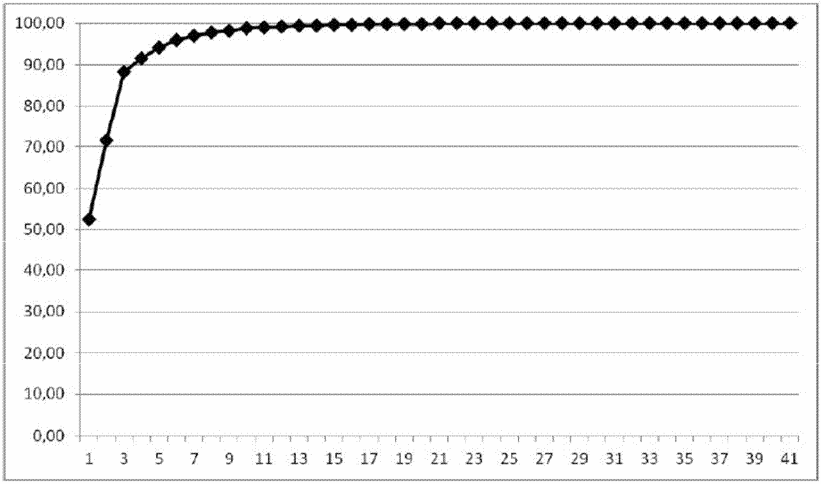

Рисунок 1 – Залежність кількості інформації від числа головних компонент

Як видно з рис. 1, 12 перших головних компонент містять 99 % інформації про мережевий трафік. У 30 компонентах, що залишилися, міститься тільки 1 % інформації, і з міркування доцільності їх можна виключити з аналізу.

Використання методу головних компонент виявило той факт, що для успішного аналізу мережевого трафіка досить використовувати 12 перших головних компонент, в яких міститься 99 % інформації про мережеві з'єднання, а не 41 параметр. Це дозволить суттєво прискорити як процес навчання нейромережевого детектора, так і процес аналізу мережевого трафіка. Для цього до виділених

параметрів з мережевого трафіка застосовуємо спочатку метод головних компонент, а потім подаємо отримані дані на вхід нейронної мережі.

При такому підході кількість n вхідних нейронів використовуваної нейронної мережі в якості детектора дорівнює 12. Подавана інформація – дванадцять перших головних компонент, подається на прихований шар детектора, де і відбувається його визначення до класу мережевої атаки або до класу легітимного з'єднання.

Порівняльний аналіз результатів виявлення мережевих атак із застосуванням методу головних компонент і без нього представлено в табл. 1.

Як видно з отриманих результатів, якість виявлення вдалося значно збільшити завдяки застосуванню методу головних компонент до параметрів мережевого трафіка. Так, приріст в якості виявлення в середньому для DoS-атак склав 1,6 %, для Probe-атак склав 26,0 %, для R2L-атак – 49,3 %, для U2R-атак – 40,9 %.

Слід зазначити, що відсоток виникнення помилкового виявлення становить менше 1,7 %.

Також за рахунок того, що для аналізу мережевого трафіка тепер використовується не 41 параметр, а 12 головних компонент, вдалося значно підвищити швидкодію системи в цілому, що є важливим критерієм для систем захисту даних.

Таблиця 1

Порівняльний аналіз результатів виявлення мережевих атак

Група атак	Клас атаки	З методом головних компонент, %	Без методу головних компонент, %	Поліпшення, %
DoS	Back	99,5	99,5	0
	Land	100,0	90,5	9,5
	Neptune	100,0	100,0	0
	Pod	98,1	98,1	0
	Smurf	100,0	100,0	0
	Teardrop	100,0	100,0	0
	Середнє по атаках	99,6	98,0	1,6

Продовження таблиці 1

Група атак	Клас атаки	З методом головних компонент, %	Без методу головних компонент, %	Поліпшення, %
Probe	Ipsweep	65,2	7,1	58,1
	Nmap	100,0	54,5	45,5
	Portsweep	99,9	99,6	0,3
	Satan	99,3	99,3	0
	Середнє по атаках	91,1	65,1	26
R2L	Ftp_write	100,0	25,0	75,0
	Guess_passwd	94,3	0	94,3
	Imap	83,3	50,0	33,3
	Multihop	57,1	28,6	28,5
	Phf	100,0	100,0	0
	Spy	100,0	0	100,0
	Warezclient	65,0	32,0	33,0
	Warezmaster	90,0	80,0	10,0
	Середнє по атаках	86,2	36,9	49,3
U2R	Buffer_overflow	83,3	63,3	20,0
	Loadmodule	100,0	0	100,0
	Perl	33,3	0	33,3
	Rootkit	30,0	20,0	10,0
	Середнє по атаках	61,7	20,8	40,9

2. Навчання штучної нейронної мережі проводилося на навчальній вибірці великого обсягу. Всі записи, що зберігаються в базі, були розділені на дві приблизно рівні за потужністю непересічних підмножини – дані з першої використовувалися для навчання штучної нейронної мережі, а дані з другої подавалися на розпізнавання. Використовувалася штучна нейронна мережа типу двошаровий персептрон.

3. Тестування повноти і точності розпізнавання мережевих атак [2]. Точність – число правильних визначень записів класу X ділиться на число фактів вказівки на клас X. Повнота — число правильних визначень записів класу X ділиться на реальне число записів класу X. При цьому пакети, впевненість у класі атаки для яких занадто низька, теж можуть вважатися аномальними, що дозволяє говорити, що нарівні із сигнатурним методом виявлення атаки використовується і аномальний.

Класи атак в базі KDD-99 мають неоднорідне представництво, що ускладнює побудову системи розпізнавання. Тому окремі класи атак були додатково розширені новими записами за допомогою розробленого програмного забезпечення вилучення мережевих ознак у форматі KDD-99. Лише 10 з 22 класів атак володіють достатньою кількістю еталонів. Малий обсяг навчальної вибірки для деяких класів атак обумовлює доцільність проведення поетапної обробки даних. У ході досліджень було проведено ряд експериментів, суть яких полягала у визначенні самого факту атаки, без уточнення її класу. Експерименти проводилися з використанням відстані Евкліда-Махаланобіса [3], мережі Кохонена та штучної нейронної мережі прямого поширення. В експерименті на репрезентативній вибірці з безлічі класів атак були видалені класи, що володіють малою кількістю прецедентів. Результати тестування відображено в табл. 2. Тестування показує досить високі показники якості розпізнавання.

Як можна зазначити (табл. 2), жоден з виділених класифікаторів не дає 100% точності по всіх класах, що спонукає до об'єднання класифікаторів в комітет. Отримані результати підтверджують, що якість класифікації безпосередньо залежить від кількості еталонів окремих класів у навчальній вибірці. При малому числі еталонів помилки виникають, в тому числі, і з високою впевненістю класифікатора.

Таблиця 2

Точність і повнота визначення мережевих атак (репрезентативна вибірка)

Група атак	Клас атак	Мережа прямого поширення		Кохонен		Метрика Евкліда–Махаланобіса	
		Повнота	Точність	Повнота	Точність	Повнота	Точність
Normal	норма	0.9896	0.9991	0.9961	0.9958	0.9689	0.9946
DoS	neptune	0.9999	1.0000	0.9999	1.0000	0.9705	0.9996
	smurf	0.9997	1.0000	0.9994	1.0000	0.9986	0.9999
	pod	1.0000	0.7308	0.9549	0.0992	0.9771	0.9275
	teardrop	1.0000	0.9919	0.9959	0.8484	0.9918	0.9939
	back	0.9973	0.7421	0.3212	0.7797	0.9510	1.0000
Probe	portsweep	0.9985	0.9763	0.9862	0.9902	0.9185	0.3441
	ipsweep	0.9930	0.9184	0.9795	0.9966	0.9431	0.9191
	satan	0.9945	0.9537	0.9846	0.9692	0.9138	0.5096
	nmap	0.9896	0.7655	0.9024	0.9001	0.4465	0.0547
r2l	warezclient	0.9941	0.1319	0.6843	0.5826	0.9490	0.0679

Навчання апарату розпізнавання атак відбувалося на 620152 прецедентах: 11733, 129610 і 478809 для трьох класів: «норми», DoS і NMap відповідно. Характеристики класифікаторів (для комп'ютера на базі Intel Xeon E5410 @ 2.33 GHz) наведено в табл. 3.

Таблиця 3

Час навчання і характеристики класифікаторів

Параметри	Мережа прямого поширення, 100 нейронів першого шару, 3 – другого	Кохонен, 2143 нейрони	Відстань Евкліда–Махаланобіса	Метод опорних векторів, 1014 опорних векторів
Час навчання, сек	310	830	280	14000
Швидкість розпізнавання, прец./сек	51000	9000	192000	59000

З табл. 3 видно, що мережа Кохонена має низьку швидкість розпізнавання, тому вона була виключена з класифікатора. Для використання навчання необхідне проведення підготовчих операцій у вигляді збору інформативних ознак з реальних потоків мережевого трафіка, що містять інформацію про прецеденти мережевих атак заданих класів. Підготовка навчальної і тестової вибірок відбувається з використанням самого модуля нейромережевого моніторингу, шляхом запуску останнього в режимі навчання та організації типових атак із заздалегідь визначених мережевих вузлів.

Тестування модуля нейромережевого моніторингу проводилося в локальній мережі, атакований комп'ютер під управлінням ОС Linux зі встановленим комплектом моніторингу, атакуючий комп'ютер під управлінням ОС Windows. Для тестування якості розпізнавання кожного з класифікованих класів проведено по 5 запусків. До складу розробленого дослідного стенду увійшли:

1. Кластерний обчислювальний пристрій з не менш ніж двома вузлами, що має мережеве обладнання та програмне забезпечення, необхідне для організації хмарних обчислень (рис. 2, де Frontend — керуюча машина, HV — гіпервізор, VM — віртуальна машина, а Node 1, Node 2, ... — обчислювальні вузли).

2. Тестова програма, яка проводить деякі хмарні обчислення на кластерному обчислювальному пристрої і вражається атаками. В якості тестової програми використовувалася програма хмарних обчислень, побудована на базі хмарної обчислювальної платформи сімейства «OpenShift» [4].

3. Мережа з персональних комп'ютерів, підключених до мережі зі встановленим програмним забезпеченням (генератором) для організації мережевих атак видів «сканування портів» та DDoS.

В якості основного засобу захисту, встановлюваного у критичних вузлах, передбачається розроблювана система захисту даних хмарних систем з підключеним модулем моніторингу аномальної мережевої активності на основі штучних нейронних мереж, яка включає в себе наступні компоненти:

1. Нейромережеві аналізатори мережевого трафіка.

2. Аналізатори на основі порівняння довгострокових і короткострокових мережевих профілів.

3. Аналізатори на основі поліноміальної відстані Евкліда–Махаланобіса.

4. Аналізатори на основі методу опорних компонент.

Для організації роботи віртуальних машин було обрано систему OpenVZ [5]. OpenVZ — це реалізація технології віртуалізації на рівні операційної системи, яка базується на ядрі Linux. В якості образів для віртуальних машин використовувався заздалегідь підготовлений варіант CentOS 7x86 зі встановленими необхідними версіями бібліотек для системи захисту. У гостьовій системі налаштовано один мережевий інтерфейс для взаємодії із зовнішньою мережею і один віртуальний — для організації доступу до віртуальних машин. Кожній віртуальній машині виділено IP-адресу. Всі віртуальні машини перебувають в одному адресному просторі. Для забезпечення доступу до мережі з віртуальних машин у системі увімкнено переадресацію. Для забезпечення доступу до віртуальних машин по ssh ззовні організовано перенаправлення портів, починаючи з 23, на віртуальні машини (23, 24 і 25). Орієнтовну схему зображено на рис. 3. У розробленій мережевій моделі є доступ до обчислювального вузла, а також організовано доступ до всіх віртуальних машин. Це не обмежує використання мережі всередині машини, але і не дозволяє отримати доступ до віртуальної машини ззовні інакше як через один, заздалегідь визначений, порт, що підвищує захищеність від мережевих атак.

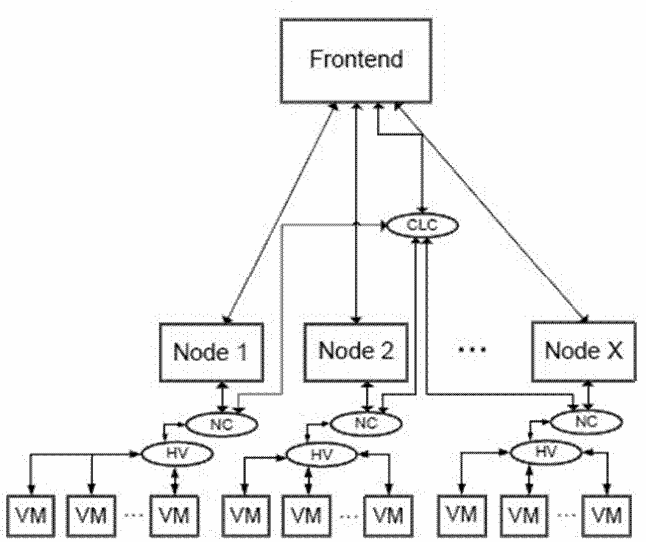

Рисунок 2 – Архітектура типової системи хмарних обчислень

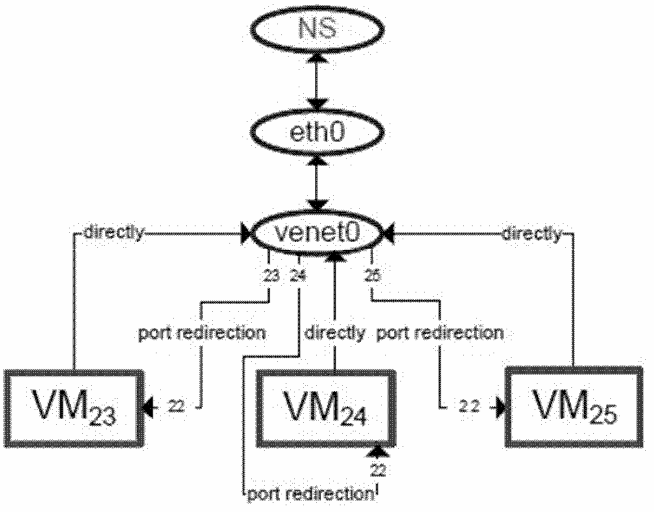

Рисунок 3 – Схема функціонування доступу до віртуальних машин

Таким чином, методика виявлення і запобігання мережевих атак на системи хмарних обчислень повинна враховувати зазначені потенційні

джерела мережевих атак. Специфіка захисту обумовлена, насамперед, вибраною системою для організації хмарних обчислень, а також операційною системою і гіпервізором. Для захисту від атак на керуючу машину, що має вихід в інтернет, на керуючій машині необхідно встановити компоненти захисту як від зовнішніх, так і внутрішніх атак. Для визначення та запобігання розподілених атак на віртуальну машину з іншої віртуальної машини хмари потрібна взаємодія між компонентами захисту на різних вузлах (віртуальних машинах). Атаки на віртуальну машину в межах одного обчислювального вузла можна визначити і зупинити різними способами:

1. Встановити систему захисту на самі віртуальні машини.
2. Встановити систему на вузли кластера для захисту з'єднання типу «міст» (bridge), через яке відбувається мережевий обмін працюючих віртуальних машин.
3. Встановити систему захисту на вузли кластера, організувавши доступ до необхідних лог-файлів віртуальних машин.
4. Встановити систему захисту на вузли кластера і розмістити у віртуальних машинах модуль, який буде проводити відправку інформації про мережеві пакети системі захисту на вузлу (даний варіант має місце, якщо з якихось причин немає можливості аналізувати мережевий трафік моста або обмін між віртуальними машинами проводиться не через міст).

Останній варіант вимагає доступу з віртуальної машини до вузла кластера, що підвищує рівень ризику і розширює спектр можливих атак і цілей для них. Атаки на віртуальну машину з іншої віртуальної машини хмари на іншому обчислювальному вузлі, по відношенню до атакуючих машин, можна блокувати або згаданим способом, або заборонити зв'язуватися віртуальним машинам на різних обчислювальних вузлах за допомогою відповідних мережевих налаштувань. Атаки на віртуальну машину ззовні, при наявності доступу до віртуальних машин, можна виявляти як на керуючій машині, так і на вузлах кластера, але запобігати краще на керуючому вузлі кластера шляхом оповіщення системи захисту, що знаходиться на ньому. Атаки на обчислювальні вузли ззовні віддаленим зловмисником можна виявляти як на керуючій машині, так і на модулях системи захисту, встановленої безпосередньо на обчислювальних вузлах. Атаки на обчислювальні вузли з віртуальної машини відбиваються лише захистом на рівні з'єднань типу «міст», вони можливі тільки у випадку видимості вузлів кластера для віртуальних машин. Найкращим захистом буде відділення адресного

простору вузлів кластера від віртуальних машин. Зазначені місця впровадження системи захисту показано на рис. 4, де NS — мережевий сенсор, AM — модулі аналізу і реакції, CM — модуль управління компонентами і SM — модуль зберігання.

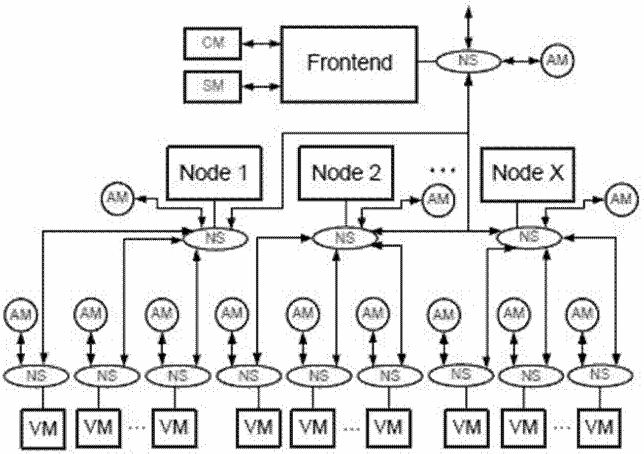

Рисунок 4 – Схема розміщення підсистем мережевого захисту

Для визначення ефективності запропонованої методики виявлення мережевих атак на системи хмарних обчислень було проведено ряд мережевих атак на експериментальний стенд. Усього було проведено шість мережевих атак, спрямованих на порушення інформаційної безпеки як керуючої машини хмари, так і віртуальних машин у її складі. Керуючій машині була привласнена приватна IP-адреса 192.168.71.88 і внутрішня IP-адреса, доступна для віртуальних машин — 10.0.0.1, яким були присвоєні адреси 10.0.0.2, 10.0.0.3 і 10.0.0.4 відповідно.

Для тестування якості розпізнавання класу «норма» використано термінал віддаленого доступу putty. Мережеві пакети класу «норма» являють собою трафік, створюваний терміналом у ході роботи віддаленого користувача з цільовою машиною, що виконує типові команди ОС. Середня повнота класифікації класу «норма» становить 100 %.

Основним напрямком атак було сканування портів (атаки класу «nmap»), для чого використовувалася утиліта Zenmap. Загальний вигляд основного вікна утиліти Zenmap наведено на рис. 5.

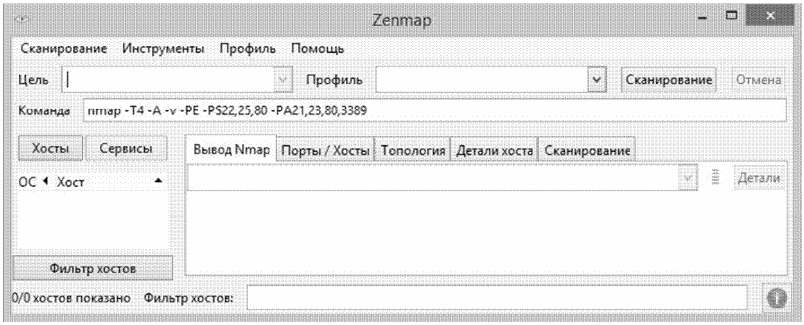

Рисунок 5 – Загальний вигляд вікна утиліти Zenmap

Дана утиліта являє собою графічний інтерфейс для програми nmap, що проводить сканування портів на віддаленій машині. Частота звернення до портів у разі атак всередині хмари була обмежена за допомогою параметра max-rate до 4000 в секунду. Дане обмеження викликано зниженням ефективності аналізатора трафіка при більшій частоті появи пакетів, що потенційно може призводити до неправильної інтерпретації ситуації або до втрати даних. Середня повнота класифікації класу «nmap» — 98,71 %.

Додатково було проведено атаку типу DDoS на сто тридцять дев'ятий мережевий порт захищеної хмарної системи за допомогою мережі з 10 комп'ютерів програмою TCP/IP DOS Attacker. Загальний вигляд основного вікна утиліти TCP/IP DoS Attacker наведено на рис. 6.

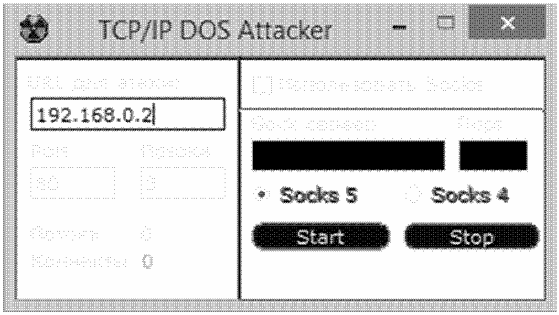

Рисунок 6 – Інтерфейс програми TCP/IP DOS Attacker

Дана утиліта організовує атаки класу Denial of Service, генеруючи великий потік мережевих пакетів на цільовий хост. Середня повнота класифікації класу «DoS» становить 99,3 %.

Висновки. Експериментальні дослідження з використанням лабораторного стенду показали високий рівень виявлення атак на основі запропонованих класифікаторів (близьке до нуля число помилок першого і другого роду у комітету класифікаторів). Також представлено застосування методу головних компонент для скорочення розміру даних для аналізу мережевого трафіка, що дозволило поліпшити якість виявлення мережевих атак на хмарні обчислювальні системи, а також підвищити швидкодію системи за рахунок скорочення аналізованих даних.

Отримані результати перевищують деякі показники, досягнуті в роботі [2] при нейромережевому аналізі даних тієї самої бази мережевих атак. Зокрема, частка помилкового виявлення атак у цій роботі, що слугувала прототипом досліджень, становить близько 12 %. Вдалося отримати також кращі показники за деякими типами атак по повноті і точності.

У разі, якщо достатнім є визначення факту атаки, механізм нейронних мереж показує високу ефективність і може бути рекомендований для використання в області моніторингу мережевих атак на хмарні обчислення.

Результати, отримані в ході експериментів, доводять правильність розміщення мережевих сенсорів і ефективність запропонованої моделі захисту системи хмарних обчислень. У той же час експеримент виявив недостатню ефективність зазначених сенсорів у плані застосування їх для аналізу трафіка між віртуальними машинами в межах одного вузла (через відсутність мережевих затримок всередині хмари). Це говорить про необхідність зниження числа одночасно спостережуваних мережевих сесій.

Бібліографічні посилання

1. **Sravan V.** Security techniques for protecting data in cloud computing / V. Sravan, K. Maddineni. — Karlskrona : Blekinge Institute of Technology, 2011. — 75 p.

2. **Wilson D.** Knowledge extraction from KDD’99 intrusion data using grammatical evolution / D. Wilson, D. Kaur // WSEAS Transactions on Information Science and Applications. — 2007. — № 4. — P. 237—244.

3. **De Maesschalck R.** The Mahalanobis distance / R. De Maesschalck, D. Jouan-Rimbaud // Chemometrics and Intelligent Laboratory Systems. — 2000. — № 1. — P. 1—18.

4. **Pousty S.** Getting started with OpenShift / S. Pousty, K. J. Miller. — Sebastopol : O'Reilly Media, 2014. — 105 p.

5. **Furman M.** OpenVZ essentials / Mark Furman. — Birmingham : Packt Publishing, 2014. — 110 p.

6. **Rhodes-Ousley M.** Information security (The complete reference) / Mark Rhodes-Ousley. — New York : Mcgraw-Hill Osborne Media, 2013. — 897 p.

7. **Li Z.** A neural network based distributed intrusion detection system on cloud platform / Zhe Li. — Toledo : The University of Toledo, 2013. — 72 p.

8. **Hesham A. I. M.** Cloud computing security, an intrusion detection system for cloud computing systems / Abdelazim Ismail Mohamed Hesham. — Pisa : University of Pisa, 2013. — 241 p.

9. **Modi C.** A survey of intrusion detection techniques in cloud / C. Modi, D. Patel, H. Patel // Journal of Network and Computer Applications. — 2013. — № 36. — P. 42—57.

Надійшла до редколегії 27.09.15