УДК 004.7 (045)

Глазок О.М., к.т.н.,
Мироненко О.К.

МОНІТОРИНГ МЕРЕЖЕВОЇ АКТИВНОСТІ КОМП'ЮТЕРІВ НА ОСНОВІ АГЕНТНОЇ ТЕХНОЛОГІЇ

Національний авіаційний університет

Запропоновано використання технології програмних агентів для створення системи моніторингу мережевої активності комп'ютерів. Запропоновано можливі функції агентів, що можуть увійти до складу такої системи, а також виділено ряд алгоритмів, які необхідно розробити в процесі її створення.

Вступ

Протягом останніх років у розв'язанні проблеми захисту доступу до ресурсів корпоративної мережі сталися суттєві зміни. До розвитку відкритих комп'ютерних мереж безпеку інформаційних систем можна було з високим ступенем надійності забезпечити за допомогою традиційних захисних механізмів, таких, як ідентифікація і аутентифікація, розмежування доступу, шифрування і т.д. Як показали результати досліджень [1, 2], нині більшість компаній з корпоративними мережами використовують міжмережеві екрани, антивірусні програми, системи виявлення атак (*IDS*). Набувають поширення нові тренди в забезпеченні інформаційної безпеки. Комерційні компанії почали дедалі більше приділяти увагу системам моніторингу та журналювання роботи автоматизованих систем і подій безпеки (*Log management*), а також системам, які дозволяють проводити аналіз подій при розслідуванні інцидентів.

Огляд і аналіз існуючих рішень

Спостереження (перехоплення) трафіку може здійснюватися наступними способами [3]:

1. Звичайним «прослуховуванням» мережевого інтерфейсу. Цей метод ефективний при використанні в сегменті мережі концентраторів (хабів) замість комутаторів (світчів); в іншому випадку даний метод малоефективний, оскільки на сніффер потрапляють лише окремі фрейми.

2. Підключенням сніффера в розрив каналу. Сніффер (аналізатор трафіку) – мережевий аналізатор трафіку, програма або програмно-апаратний пристрій, призначений для перехоплення і (або) аналізу мережевого трафіку, призначеного для інших вузлів.

3. Відгалуженням (копіюванням) трафіку, що може виконуватися програмними або апаратними засобами, і подальшим спрямуванням його копії на сніффер.

4. Через атаку, яка призводить до перенаправлення трафіку на сніффер з подальшим поверненням. Відомі техніки виконання таких атак на канальному рівні (*MACspoofing*) та на мережевому рівні (*IPspoofing*).

Аналіз мережевого трафіку дозволяє досягти наступних цілей:

1. Виявити в мережі шкідливе і несанкціоноване ПЗ, наприклад, мережеві сканери, флудери, троянські програми, клієнти пірінгових мереж та інші. Цю задачу зазвичай розв'язують за допомогою спеціалізованих сніфферів – моніторів мережевої активності.

2. Перехопити будь-який незашифрований (а в деяких випадках і зашифрований) користувацький трафік з метою його аналізу.

3. Локалізувати несправність мережі або помилку конфігурації мережевих сервісів (сніффери найчастіше застосовуються системними адміністраторами саме для цієї мети).

4. Виявити паразитуючий, вірусний і кільцевий трафік, наявність якого збільшує завантаження мережевого обладнання та каналів зв'язку. Однак для розв'язання цієї задачі сніффери недостатньо ефективні; як правило, для цих цілей використовують збір різноманітної статистики серверами і активним мережним устаткуванням і її подальший аналіз.

Сучасні способи реалізації моніторингу трафіку комп'ютерної мережі наступні:

1. Моніторинг в режимі «*promis-cuous*».

Режим *promiscuous* – це особливий режим обладнання Ethernet, як правило – карт мережевих інтерфейсів (*NIC*), який дозволяє отримувати карті весь трафік мережі, навіть якщо цей трафік не адресований конкретно даній карті. (За замовчуванням мережева карта виділяє адресований їй трафік шляхом порівняння адреси призначення *Ethernet*-пакета і апаратної адреси пристрою (*MAC*-адреси), а весь не адресований їй трафік ігнорує.)

2. Перехоплення пакетів з використанням протоколу *ARP* (*ARP-spoofing*).

Цей спосіб використовує недоліки *ARP*-маршрутизації – монітор здійснює підміну *MAC*-адреси одержувача пакета, таким чином, перехоплюючи його.

3. Моніторинг з використанням хабів.

Для такого моніторингу може використовуватися будь-який комп'ютер, підключений до хабу, оскільки хаб передає прийняті/передані дані від маршрутизатора (роутера) на всі свої порти. Даний варіант дозволяє спостерігати за трафіком лише в підмережі, де знаходиться хаб.

4. Моніторинг з використанням комутаторів.

Керований (*managed*) комутатор з підтримкою дзеркалювання портів (*port mirroring*) дозволяє перенаправляти трафік з визначених портів на певний порт комутатора і, таким чином, здійснювати моніторинг сегменту мережі, де встановлений даний мережевий пристрій.

5. Програми-аналізатори (сніффери) мережевого трафіку – здійснюють перехоплення трафіку програмно-апаратним способом. Даний спосіб базується на використанні інших способів, перерахованих вище, та вимагає налаштування певної конфігурації мережі.

Відома значна кількість програм-сніфферів з різними можливостями, зокрема:

Для операційних систем лінійки *Microsoft Windows*: *CommView*, *SpyNet*, *Analyzer*, *IRIS*, *WinDUMP* (аналог *tcpdump for Unix*), *SniffitNT*, *ButtSniff*, *Wireshark*, *LanExplorer*, *Net Analyzer*.

Для операційних систем *Unix*, *Linux*: *linsniffer*, *linux_sniffer*, *Sniffit*, *HUNT*, *READSMB*, *tcpdump*, *Dsniff*, *Wireshark*, *Ksniffer*.

Постановка задачі

Моніторинг трафіку – важливе джерело інформації для ефективного управління мережею. Дані, отримані в результаті моніторингу трафіку, беруться до уваги при розподілі ресурсів, плануванні обчислювальних потужностей для виконання корпоративних додатків, виявленні та локалізації відмов, розв'язанні питань безпеки.

У мережах шинної топології, завдяки наявності єдиного спільного середовища розповсюдження даних, моніторинг трафіку був відносно простим завданням. Для стеження за всім трафіком до такої мережі достатньо підключити єдиний пристрій для реєстрації трафіку, або використати мережевий інтерфейс і відповідний програмний засіб на одному з існуючих вузлів.

В ході подальшого розвитку мереж передачі даних, зростаючі вимоги до пропускної здатності мережі і розвиток технологій комутації пакетів зумовили швидкий перехід від єдиного середовища передачі, спільно використовуваного усіма вузлами, до сегментованих топологій. При цьому загальний трафік вже неможливо «побачити» з однієї точки – для отримання повної картини потрібно виконувати моніторинг вхідного та вихідного трафіку окремо на кожному комп'ютері, що підключений до корпоративної мережі. Оскільки даний процес вимагає значних витрат обчислювальних потужностей персонального комп'ютера, то це може сповільнити роботу системи в цілому. Окрім проблеми додаткового обчислювального навантаження (*overhead*), постають і інші проблемні питання: надійне зберігання логів (файлів чи баз даних з результатами моніторингу), збирання цих даних для подальшого їх аналізу, поновлення баз даних заборонених (недопустимих) з'єднань, і все це з урахуванням можливості збоїв у роботі обладнання та виходу частин мережі з ладу.

Очевидно, що ефективність моніторингу з використанням аналізаторів мережевого трафіку залежить від топології досліджуваної мережі, її конфігурації, та від набору пристроїв, з яких мережа побудована. У разі виникнення потреби організації спостереження за трафіком у мережі деякої компанії доведеться додавати чи замінювати обладнання та переналаштовувати системи відповідно до нової топології. Додаткові переналаштування необхідно буде зробити після кожної зміни складу та топології мережі, які можуть статися через підключення, відключення, заміну комп'ютерів або мережевої апаратури, або через відмови окремих апаратних елементів. Таким чином, складність задачі організації та підтримки дослідження трафіку зростає разом із зростанням складності структури мереж.

Ось чому нині є актуальною розробка алгоритмів моніторингу для вирішен-ня вище перерахованих проблем і вибір для цього технології, яка вимагала б яко-мога менших витрат ресурсів комп'ютера, на якому реалізовано спостереження за трафіком.

Одним з актуальних наукових завдань є розробка алгоритмів для програмного засобу моніторингу мережевої активності персональних комп'ютерів, який би міг працювати у мережі будь-якої топології та дозволяв забезпечити постійний моніторинг трафіку мережі, навіть у випадках збоїв обладнання і при неможливості втручання адміністратора. Такий програмний засіб має бути розподіленою інте-

лектуальною системою, яка може самостійно приймати рішення відповідно до ситуації. Цим вимогам відповідає технологія програмних агентів. Розробка системи моніторингу мережевого трафіку персональних комп'ютерів на основі агентної технології дозволить забезпечити більш надійну інформаційну безпеку корпоративної мережі організації.

Мета

Метою даної статті є розгляд можливості створення розподіленої інтелектуальної системи моніторингу мережевого трафіку на основі технології програмних агентів та визначення проблемних питань в цій сфері, що потребують розв'язання в ході проектування такої системи.

Основна частина

Системи взаємодіючих інтелектуальних агентів представляють собою один із напрямків в області штучного інтелекту та прикладного програмування, який активно розвивається. Про це свідчить значна кількість наукових праць, книг і публікацій, присвячених цій темі.

Однак, на даний момент, немає загальноприйнятого визначення поняття «Агент». Найбільш визнаним є визначення програмного агента як комп'ютерної програми або програмної системи, яка виконується асинхронно, відповідно до поведінки, закладеної в неї певною особою або організацією, і має такі особливості: автономність, взаємодія, мобільність, реактивність, активність, індивідуальність «бачення світу», комунікабельність і кооперативність, інтелектуальність поведінки [4, 5].

Кожен агент – це процес, який володіє достатнім для виконання своїх функцій обсягом інформації (знань) про підконтрольний йому об'єкт і можливістю обмінюватися цими знаннями з іншими агентами. З точки зору об'єктно-орієнто-ваного підходу агента можна розглядати як комплекс функцій в поєднанні з інтерфейсом, який має здатність отримувати і надсилати запити і відповіді на них.

Мультиагентною системою (англ. – *Multi-agent system, MAC*) називається розподілена система, до складу якої входить кілька взаємодіючих агентів, що можуть спілкуватися один з іншим, взаємодіяти, здійснювати обмін поточною інформацією [6]. Один з агентів системи може здійснити запит до іншого агента на передачу певних даних або виконання певних дій. *MAC*, що діє як єдиний суб'єкт, повинна характеризуватися деякою загальною для всіх субагентів метою і координацією між ними дій для досягнення цієї мети. При цьому загальна задача розбивається на кілька підзадач, які розподіляються між агентами [7].

Актуальність створення мульти-агентних систем в даний час обумовлюють наступні основні причини:

- існує тенденція до ускладнення організаційних структур компаній, що тягне за собою відповідне ускладнення систем передачі та обробки даних, а також задач їх моніторингу з урахуванням можливих відмов окремих апаратних складових мережі;

- задачі, що розв'язуються в сучасних інформаційних системах, створюють розподілені (в просторі та у функціональному плані) та неоднорідні навантаження на обчислювальні вузли і канали передачі даних.

Ці причини приводять до того, що централізоване управління в таких інформаційних системах стає неефективним, порівняно з розподіленим.

Як уже сказано, задача розподіленого моніторингу включає в себе ряд різних за своєю природою задач – спостереження за трафіком, запис результатів спостережень, збирання отриманої інформації та її обробка. Розглядаючи можливий склад мультиагентної системи моніторингу мережевого трафіка, можна запропонувати ввести до складу системи кілька типів агентів для розв'язання різних задач.

1. Інтерфейсний агент – приймає зовнішні команди від користувача або логічного сервера *MAC* і передає їх для виконання іншим агентам. Також відображає в інтерфейсі користувача необхідну йому інформацію щодо функціонування системи.

2. Агент-сніффер – сканує порти комп'ютера з метою виявлення вхідного та вихідного трафіку, здійснює його реєстрацію та визначення метрик.

3. Агент-збирач даних – реалізує алгоритм надійного збереження результатів моніторингу мережі. Збирає дані від усіх агентів-сніфферів та передає їх на комп'ютер адміністратора або логічний сервер *MAC*. У разі, якщо він недоступний – відправляє дані на зберігання в один з резервних банків даних і очікує, доки пункт призначення відновить свою роботу. Таким чином реалізується розподілене зберігання результатів моніторингу.

4. Агент клонування – здійснює копіювання всіх даних, необхідних для організації моніторингу (коду програмних агентів, конфігураційних файлів, тощо), на комп'ютер, який щойно приєднався до мережі і ще не має встановленого клієнтського програмного забезпечення системи моніторингу.

5. Агент-логічний сервер *MAC*: здійснює загальне керування іншими агентами та частинами системи та координацію їх роботи, а також надає адміністратору командний інтерфейс для керування системою.

6. Агент-контролер: слідкує, щоб на комп'ютері не здійснювалися заборонені з'єднання. Його алгоритм також може передбачати оновлення баз заборонених адрес через спеціалізовані сервіси в Інтернеті.

Розробка алгоритмів діяльності агентів, що складають систему, може вестися на основі одного з двох припущень. В першому варіанті (централізована система) можна вважати, що логічний сервер системи працює безвідмовно, не потребує дублювання, і, таким чином, він є постійним центром прийняття рішень та керування іншими компонентами системи. В другому варіанті необхідно виходити з того, що логічний сервер потребує дублювання, яке дозволить реалізувати безперервне виконання його функцій в умовах можливих відключень або відмов окремих обчислювальних вузлів та мережевого обладнання. В такому разі необхідно мати кілька копій програмного забезпечення агента-логічного сервера на вузлах мережі, або навіть копію на кожному вузлі, де працює система.

Для створення системи моніторингу мережевої активності на основі агентної технології необхідно розробити і реалізувати такі основні алгоритми:

1. Алгоритм колективного прийняття рішень з вибору «керівництва системи». Цей алгоритм повинен активізуватися у випадку відключення або недоступності вузла, на якому до цього виконувався код логічного сервера, і дозволити компонентам, що лишилися у працездатному стані, обрати новий центр керування. Успішні розв'язання подібних задач відомі – наприклад, процедура автоматичного обрання контролера домену у мережі Windows-комп'ютерів.

2. Алгоритм сповіщення програмних агентів про існування та розміщення інших агентів, що дозволить їм у випадку відмови частини компонентів мережі перейти до виконання колективних дій з метою відновлення працездатності системи.

3. Алгоритм сповіщення програмних агентів логічним сервером про стан системи у такій формі, що у разі відключення вузла-носія логічного сервера новий обраний агентами логічний сервер зможе продовжити її функціонування.

4. Алгоритм тимчасового зберігання інформації агентами-збирачами даних у розміщеннях, з яких вони можуть бути пізніше отримані із заданою імовірністю в умовах можливих відключень.

Висновки

В статті розглянуто можливий склад мультиагентної системи для моніторингу мережевого трафіку. Виділено ряд алгоритмів, які необхідно розробити в процесі створення такої системи. Подальші дослідження можливі в напрямках безпосереднього створення та дослідження вказаних алгоритмів, оптимізації їх роботи з точки зору балансу між допустимими додатковими навантаженнями на обчислювальні та мережеві ресурси, з одного боку, і імовірнісними і швидкісними показниками роботи системи, з іншого, дослідження можливості масштабування системи.

Список літератури

1. Романов М. Безопасность корпоративных сетей: мониторинг, анализ, управление / М.Н.Романов //Storage News. – №1 (30). – 2007. – С. 21–25.

2. 2010/2011 Computer Crime and Security Survey /R.Richardson et al. – N.Y.: Computer Security Institute, 2011. – P. 42.

3. Столлингс В. Современные компьютерные сети /В.Столлингс. – СПб.: Питер, 2003. – 783 с.

4. Рассел С. Искусственный интеллект: современный подход. / С. Рассел, П. Норвиг. – М.: Вильямс, 2006. – 1408 с.

5. Городецкий В.И. Многоагентные системы (обзор). /Городецкий В.И., Грушинский М.С., Хабалов А.В. //Новости искусственного интеллекта. – 1998. – № 2. – С. 64–116.

6. Таненбаум Э. Распределенные системы: принципы и парадигмы /Таненбаум Э., Стеен М. – СПб.: Питер, 2005. – 878 с.

7. Швецов А.Н. Агентно-ориентиро-ванные системы: от формальных моделей к промышленным приложениям /Швецов А.Н. – Вологда: Вологодский ГТУ, 2008. – 101 с.