https://doi.org/10.31649/1997-9266-2021-155-2-92-97

УДК 004.77

І. П. Малініч[1]
В. І. Месюра[1]
І. Р. Арсенюк[1]

АНАЛІЗ ВИКОРИСТАННЯ ТРАФІКУ ПРИ СКАНУВАННІ КОМП'ЮТЕРНИХ МЕРЕЖ РІЗНИМИ ВЕРСІЯМИ NMAP

[1]Вінницький національний технічний університет

На сьогодні існує велика кількість інструментів для моніторингу та діагностики мережі, які можуть використовуватись з різними цілями, як уповноваженими на це особами, так і іншими особами, що можуть здійснювати несанкціонований доступ до мережі. Одним з них є мережевий сканер Nmap. Мережевий сканер Nmap — це досить потужний додаток мережевого сканування з відкритим поточним кодом. Наразі він є майже незамінним інструментом для діагностики роботи мереж, виявлення помилкових мережевих конфігурацій, а також допомагає під час пошуку вразливостей всередині мережі. Мережевий сканер Nmap постійно вдосконалюється його авторами та незалежними розробниками. Тому з плином часу цей додаток розширює свій функціонал та поліпшує вже існуючі інструменти. Однак це суттєво впливає на його поведінку в мережі.

Для виявлення сканувальної активності у мережі зазвичай використовуються сніфери, які фіксують пакети, що через них проходять. На сьогодні за їхньою допомогою та допоміжного програмного забезпечення для виявлення вторгнень можливо виконувати ідентифікацію хостів, які здійснюють сканування. Однак залишаються актуальними задачі ідентифікації програмного забезпечення, що сканує мережу, та його версій. У статті розглядається розв'язання задачі на прикладі мережевого сканера Nmap.

Розглянуто використання трафіку різними версіями мережевого сканера Nmap під час виконання різних фаз сканування. Відмінності у роботі програми виникають в результаті оновлення поточного коду цього додатку та його скриптів сканування. У більшості випадків код у межах однієї мажоритарної версії не зазнає суттєвих змін. Однак навіть незначні зміни можуть вплинути на процес ідентифікації програмного забезпечення в процесі аналізу мережевого трафіку. Проаналізовано роботу таких версій Nmap: 7.01, 7.60, 7.80.

Ключові слова: Nmap, мережевий трафік, сніфер, сканування мереж, система виявлення вторгнень.

Вступ

Актуальність. Аналіз трафіку на наявність сканувальної активності є актуальним завданням у роботі сучасних корпоративних мереж. Своєчасне виявлення паразитного трафіку у комп'ютерних мережах та можливості здійснення несанкціонованого доступу є запорукою беззбійної роботи мережі. Важливим зокрема є дослідження особливостей поведінки сканувальних програм.

Зв'язок з важливими практичними завданнями. Визначення ключових відмінностей мережевої активності різних версій сканувального програмного забезпечення для комп'ютерних мереж є важливим для дослідження змін поведінки подібного програмного забезпечення зі зміною версій. Виявлення сканувального трафіку за допомогою систем виявлення вторгнень з фіксованими правилами не завжди дозволяє точно визначити, за допомогою якого програмного забезпечення воно здійснювалось. Характеризація подібних змін дозволить удосконалити алгоритми виявлення сканувальної активності у мережі.

Аналіз останніх досліджень та публікацій. Робота з програмою Nmap найдетальніше викладена у книзі [1], написаній розробником цієї програми Гордоном Ліоном. Книга містить повну документацію з Nmap. Основну увагу зосереджено на тих розділах книги, які є актуальними для цього дослідження, зокрема розділи 3—5, де описуються основні режими роботи програми, ключі командного рядка для їх виклику та опис алгоритмів їх роботи. У розділі 9 приділено увагу скриптовому рушію Nmap (Nmap Scripting Engine, або просто NSE), на базі якого працюють модулі Nmap. В розділі також наведено процес написання скрипту на мові програмування Lua, а також приділено увагу паралелізму виконання скриптів під час різних етапів сканування. Розділ 10 цієї книги

описує процес виявлення сканування програмою Nmap за допомогою сніферів та засобів мережевих екранів.

В іншій роботі [2] наведено аналіз пакетів сканування портів з використанням теорії доказів Демпстера–Шафера. Серед практичних аспектів увагу приділено класифікації видів сканувань на такі чотири категорії: вертикальне, горизонтальне, стробове та блокове. Під вертикальним скануванням автор має на увазі сканування всіх портів окремої машини; під горизонтальним — сканування одного порту на всіх комп'ютерах мережі. Стробове сканування передбачає сканування деяких портів окремих машин, а блокове сканування — всіх машин у мережі. Аналіз показав, що теорія Демпстера–Шафера може бути використана для виявлення сканувальних атак.

У статті [3] виконано сканування різних інтернет-серверів методами Ping sweep, TCP sweep та Null-сканування, описано їх особливості, подано результати сканування різних ресурсів та їх порівняння. Порівняння можливостей пасивного та активного сканування наводиться у роботі [4]. Приділяється увага їх реалізації у різних видах сканерів, зокрема Nmap. Огляд видів TCP та UDP сканування здійснюється у тезах [5]. Розглядаються такі підвиди TCP-сканування: TCP connect scan, TCP SYN, TCP ACK, TCP NULL та UDP-сканування. Налаштування захоплення та аналізу пакетів сканувального трафіку за допомогою інструментів iptables та Snort наведено у матеріалах роботи [6]. Увагу приділено налаштуванню мережевого екрану iptables.

Процес виявлення операційної системи на хості сканування описано у статті [7]. Приділено увагу процесу формування та використання бази даних відбитків різних операційних систем, їх ідентифікації у процесі сканування, а також відмінностям процесу ідентифікації при скануванні машин з різними операційними системами. Особливості логування пакетів засобами iptables у віртуалізованих середовищах наведені у роботі [8]. У роботі [9] описуються різні режими роботи Nmap за використання різних опцій командного рядка, зокрема порівнюються різні види горизонтального сканування за допомогою TCP ping та ICMP ping. Робота окремих опцій командного рядка описана у матеріалах [10].

Невирішені аспекти проблеми. Автори розглянутих досліджень та публікацій приділили багато уваги як самому процесу сканування комп'ютерних мереж, так і порівнянню роботи різних систем сканування. Однак тому, як змінюється сканувальна мережева активність зі зміною версій програмного забезпечення увага не приділялась

Метою роботи є розроблення методу для виявлення відмінностей процесу сканування у різних версіях Nmap та визначення у яких фазах сканування вони виникають. Для розробки цього методу необхідно виконати збір метаданих мережевих пакетів при скануванні комп'ютерних мереж різними версіями Nmap та провести їх аналіз. Необхідність виконання такого аналізу виникла в ході дослідження мережевого трафіку, який виникає під час використання різного програмного забезпечення для сканування мереж [11], оскільки для продовження дослідження необхідно знати чи варто виділяти міноритарні версії модульних мережевих сканерів у окремі класи. Виявлення відмінностей у різних фазах сканування в різних версій Nmap, а також порівняння кількісних показників кожного сканування є важливим для подальшого вивчення мережевої активності інших програмних пакетів та скриптів для сканування мережі.

Постановка задачі

Дослідження відмінностей у мережевій активності різних версій Nmap потребує виконання різних версій у тестовому середовищі, а також збору даних про пакети, що проходять під час сканування. Спочатку необхідно визначити, які дані про пакети будуть збиратись. Для цього необхідно записати мережеві пакети в дамп, після чого визначити їх кількість, розмір, а також встановити належність до протоколу ICMP або протоколів транспортного рівня. Іншим завданням є визначення належності пакетів до конкретних фаз сканування.

Також слід здійснити вибір досліджуваних версій Nmap. Вибір мажоритарної версії Nmap передбачає визначення версії цього ПЗ, міноритарні підверсії якої зможуть мати найбільше змін. Архітектура такої програми передбачає, що її поведінка сильно залежить від NSE-скриптів. 7-му версію вибрано завдяки тому, що в ній найбільше доступних скриптів різних версій, а також тому, що вона є актуальною на момент написання статті. Вибір міноритарних версій зроблено завдяки їх поширенню у репозиторіях LTS-версій дистрибутиву ОС GNU/Linux Ubuntu [12] — 7.01, 7.60 та 7.80.

Над отриманими даними слід провести аналіз схожості пакетів засобами мови програмування Python. Для цього необхідно визначити важливі для аналізу параметри заголовків пакетів, і на основі їх задати критерії схожості для побудови масивів даних з пакетами, які не підпадатимуть під їх дію.

Також необхідно визначити причину виникнення подібних пакетів та визначити, яким чином вони в цілому впливають на процес сканування.

Випробувальне середовище

Методика побудови стенду вже згадувалась під час вивчення сканування портів [2]. Однак конфігурація, в якій скануючий хост та цілі знаходяться в тому самому широкомовному домені суттєво змінює алгоритм сканування, задіюючи при цьому сканування на другому рівні моделі OSI [1]. Тому між підмережами має знаходитись віртуальний маршрутизатор (рис. 1) для відтворення сценарію віддаленого сканування, завдяки чому L2-сканування використовуватись не буде.

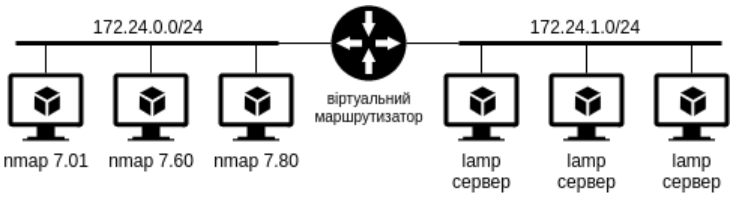

Рис. 1. Схема віртуальної мережі

У віртуальних машинах цілей сканування розгорнуто образ Ubuntu 20.04 LTS з LAMP-стеком та FTP-сервером. Використано такі пакети: Apache 2.4.41, MySQL 8.0.22, PHP 7.4 та ProFTPd 1.3.6. Завдання передбачає збір інформації про заголовки пакетів, для чого буде використовуватись програмна бібліотека libpcap. Для визначення належності пакетів до конкретної фази сканування у режимі реального часу використано можливості мережевого екрану з відкритим поточним кодом NetFilter.

Параметри сканування мають бути задані таким чином, щоби процес сканування мав водночас і схожі параметри з тими, що найчастіше використовуються ботами та популярними сервісами онлайн-сканування, а також використовували б найширший набір скриптів Nmap. Для сканування вибрано такі прапорці:

– "T4" — це «агресивний» шаблон таймінгів, який розрахований на надійний канал передачі даних та низьку затримку між сканувальними ітераціями;

– "A" — вмикає виявлення ОС, виявлення версій, розширені скрипти сканування та трасування;

– "oX" — зберігає звіт у файлі XML для подальшого парсингу в процесі обробки даних;

– "v" — дозволяє вивести детальнішу інформацію про процес сканування.

Збір даних

Після запуску та роботи випробувального середовища зібрано три окремі набори даних. Кожен набір даних містить дамп перехоплених пакетів, логи Nmap та iptables. На базі інформації про пакети згенеровано файли зі статистикою використання мережі. Вміст файлів статистики подано у табл. 1. У таблиці наводиться статистика надісланих та отриманих пакетів різних версій Nmap. Сценарій сканування за замовчуванням Nmap передбачає переважно сканування TCP-сервісів. Згідно з журналом фіксації пакетів сканування UDP-портів відбувається на етапі сканування служб. Воно здійснюється скриптами, які сканують сервіси, що працюють на базі обох протоколів — TCP та UDP, такі як DNS. З огляду на відмінність у тривалості виконання та у кількості пакетів варто виконати глибший аналіз заголовків та вмісту пакетів.

За допомогою функціоналу раніше створених програмних бібліотек [11] зібрані дампи пакетів конвертовано у набори метаданих про заголовки пакетів та розбито по фазам сканування: горизонтального — сканування на доступність хостів мережі тестового стенду 172.24.1.0/24, стробового — сканування портів відомих служб кожного виявленого хосту, а також сканування сервісів, під час якого визначались версії програмного забезпечення служб, виявлених під час стробового сканування.

Таблиця 1

Статистика використання мережі під час сканування

Версія програми	7.01	7.60	7.80
Тривалість виконання (сек)	111	247	32
Передано байтів	937862	951182	750892
Отримано байтів	937158	950870	750788
Передано пакетів	15366	17491	9307
Отримано пакетів	12715	12567	6771
Кількість переданих ICMP-пакетів	1166	1224	1060
Кількість переданих TCP-пакетів	14115	16159	8222
Кількість переданих UDP-пакетів	85	108	25
Кількість отриманих ICMP-пакетів	247	308	81
Кількість отриманих TCP-пакетів	12468	12259	6690
Кількість отриманих UDP-пакетів	0	0	0

Обробка та аналіз зібраних даних

Під час повторних запусків процесу сканування виявлено, що деякі параметри заголовків пакетів не є сталими, та змінюються випадковим чином. Серед них виділено такі параметри: порт походження TCP та UDP, порядковий номер (sequence number) TCP та номер підтвердження (acknowledgement number) TCP, завдяки чому вони були проігноровані. Виділено такі поля, які залишаються незмінними: ідентифікатор транспортного протоколу, IP-адреса походження, IP-адреса призначення, порт призначення (за наявності), довжина пакета, TCP-прапорці пакета (за наявності) та хеш-сума блоку даних пакета. Дублюючі елементи прибрано.

Виявлено, що кожна версія Nmap під час стробового сканування сканує однаковий набір портів. На рис. 2 зображено діаграму використання номерів портів під час роботи всіх трьох версій Nmap. Як видно на діаграмі, найщільніше відбувається сканування загальновідомих портів служб. Всі номери портів зазначені у скриптах стробового сканування Nmap. За одночасної роботи різних скриптів стробового сканування можливі повторні підключення того самого порту. Для подальшого аналізу виконано вибірку лише надісланих пакетів.

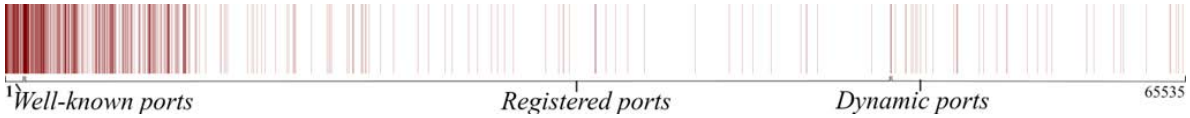

Рис. 2. Діаграма використання портів при стробовому скануванні в Nmap

Запропоновано метод виявлення унікальних пакетів у наборах метаданих кожної версії. Метод полягає у вибірці неоднакових пакетів для кожної версії та подальшому визначенню схожих пакетів, які мають варіативні елементи всередині блоку даних. Для здійснення початкової вибірки кожен набір метаданих потрібно представити у вигляді множин, з яких виділяється множина унікальних пакетів таким чином:

$$U_n = P_n \setminus \left(P_1 \cap P_2 \cap P_3\right), \tag{1}$$

де U_n — множина унікальних пакетів сканування конкретної фази; n — версія, для якої відповідно відбувається визначення унікальних пакетів; P_n, P_1, P_2, P_3 — множини елементів наборів даних про пакети відповідної фази сканування для кожної версії програми.

Так відбувається виключення з множини U пакетів, що мають варіативні елементи всередині блоку даних. Блок-схема алгоритму, який це здійснює показана на рис. 3.

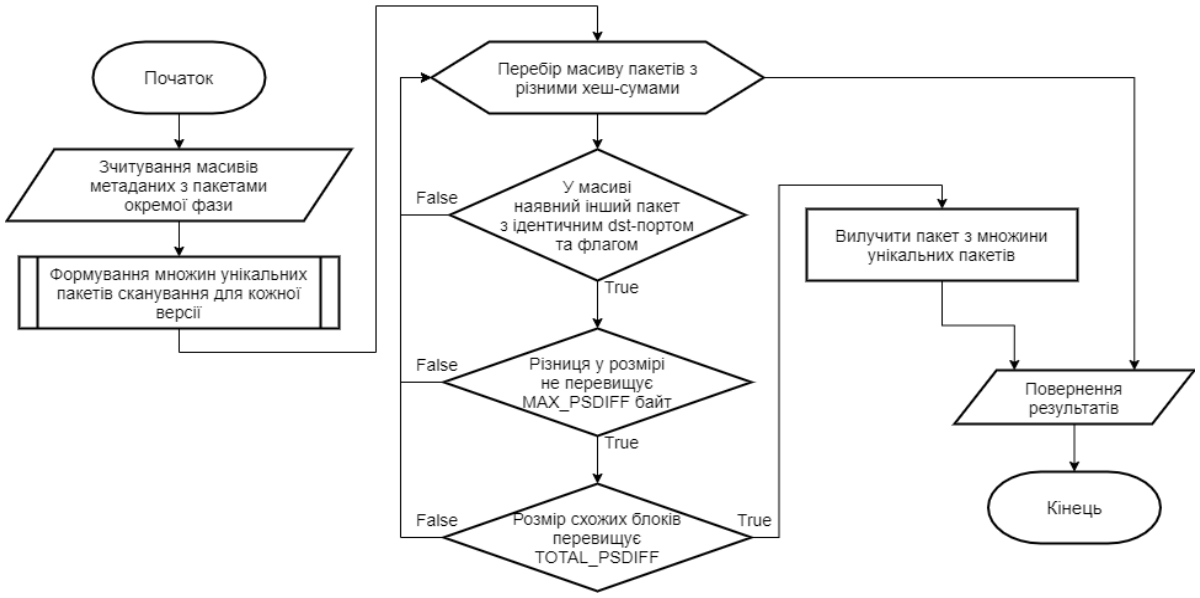

Рис. 3. Блок-схема алгоритму виключення пакетів, що мають ідентичні прапорці та спільні елементи блоку даних

У процесі виключення пакетів відбувається перебір пакетів з множини U, під час чого здійснюється пошук пакетів зі схожими портами призначення та прапорцями TCP (якщо такі є). Для кожного протоколу визначаються критерії подібності пакетів, такі як граничні значення різниці розміру пакетів з варіативними елементами блоку даних та розмір спільних блоків. Пакети, які мають такі

ознаки подібності вилучаються з множини *U*. Також у множині унікальних елементів відсутні повторні елементи, оскільки вони видаляються при обробці вхідних даних. В результаті застосування методу відбувається побудова масивів з унікальних пакетів для кожної фази кожної версії окремо.

Таблиця 2

Кількість унікальних пакетів по фазам

Фаза сканування	Версія		
	7.01	7.60	7.80
Горизонтальна	3	0	3
Стробова	0	1000	0
Сервісів	10	19	10

Отримана в результаті аналізу кількість унікальних пакетів наведена у табл. 2. Відмінність роботи вибраних міноритарних версій виявлено на першому етапі (горизонтального сканування): у них різні алгоритми визначення активних хостів. Версія 7.60 під час сканування визначає віртуальний маршрутизатор як активний хост, та намагається його сканувати у всіх подальших етапах сканування, завдяки чому під час стробового сканування кількість унікальних надісланих пакетів зростає на 1000. Різниця унікальних надісланих пакетів для сервісів зумовлена тим, що деякі протоколи можуть мати варіативні елементи у запитах. Як результат, мажоритарна версія 7 мережевого сканера Nmap не потребує визначення окремих класів міноритарних версій для ідентифікації системами аналізу трафіку.

Висновки

Розглянуто використання трафіку різними версіями мережевого сканера Nmap під час виконання різних фаз сканування. Проаналізовано роботу таких версій Nmap: 7.01, 7.60, 7.80 на предмет наявності відмінних пакетів у різних фазах сканування, що показало необхідність створення спільних класів визначення сканувального ПЗ для міноритарних версій Nmap.

Розроблено метод виявлення унікальних пакетів у наборах метаданих різних версій Nmap, який дозволяє отримувати дані про відмінність процесу сканування у цих версіях та визначати у яких фазах сканування вони виникають. Розроблений метод у подальшому планується удосконалити для аналізу іншого програмного забезпечення, що здійснює сканування мереж для його подальшої класифікації.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

[1] Gordon Fyodor Lyon, *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning*, Sunnyvale, CA, USA: Insecure, 2009, 464 p.

[2] R. R. Singh, and D. S. Tomar, "Port scanning attack analysis with Dempster–Shafer evidence theory," *Int. J. Appl. Eng. Res*., vol. 12, no. 16, pp. 5900-5904, 2017.

[3] G. Bagyalakshmi et al., "Network vulnerability analysis on brain signal/image databases using Nmap and Wireshark tools," *IEEE Access*, vol. 6, pp. 57144-57151, 2018.

[4] В. Ю. Кива, і Ю. С. Дрозд, «Аналіз існуючих методів кібернетичної розвідки інформаційно-телекомунікаційних мереж,» *Збірник наукових праць Центру воєнно-стратегічних досліджень Національного університету оборони України імені Івана Черняховського*, № 3, с. 62-66, 2017.

[5] В. В. Довгий, і І. В. Небесний, *Алгоритми сканування портів у корпоративній комп'ютерній мережі*. Тернопіль, Україна: ТНЕУ, 2018. [Електронний ресурс]. Режим доступу: http://dspace.tneu.edu.ua/handle/316497/31957 .

[6] В. В. Довгий, *Алгоритми виявлення процедури сканування портів в корпоративній комп'ютерній мережі*. Тернопіль, Україна: ТНЕУ, 2018. [Електронний ресурс]. Режим доступу: http://dspace.tneu.edu.ua/handle/316497/32436 .

[7] J. P. S. Medeiros, A. M. Brito, and P. S. M. Pires, "A data mining based analysis of nmap operating system fingerprint database," in *Computational Intelligence in Security for Information Systems*, Springer, 2009, pp. 1-8.

[8] І. П. Малініч, і В. І. Месюра, «Ін'єктивний метод отримання даних користувацького досвіду в ігрових симуляторах комп'ютерних мереж,» *Вісник Вінницького політехнічного інституту*, № 5, с. 49-54, 2019.

[9] Mark Wolfgang, *Host Discovery with NMAP* 2015. [Online]. Available: https://havel.mojeservery.cz/wp-content/uploads/2015/10/nmap-discovery-howto-2002.pdf . Accessed on: December 17, 2020.

[10] *Host Discovery with NMAP*. [Online]. Available: https://medium.com/@minimalist.ascent/host-discovery-with-nmap-a3759e3d214f . Accessed on: December 17, 2020.

[11] І. Малініч, В. Месюра, і П. Малініч, *Проблеми створення середовищ для визначення типу сканувальної активності, що здійснюється при скануванні мереж*, Вінниця: ВНТУ, 2021.

[12] Nmap. Ubuntu packages [Online]. Available: https://packages.ubuntu.com/search?keywords=nmap. Accessed on: December 17, 2020.

Рекомендована кафедрою комп'ютерних наук ВНТУ

Стаття надійшла до редакції 31.03.2021

Малініч Ілля Павлович — асистент кафедри комп'ютерних наук, e-mail: malinich@vntu.edu.ua ;

Месюра Володимир Іванович — канд. техн. наук, доцент, професор кафедри комп'ютерних наук, e-mail: mesyura@vntu.edu.ua;

Арсенюк Ігор Ростиславович — канд. техн. наук, доцент кафедри комп'ютерних наук.

Вінницький національний технічний університет, Вінниця

I. P. Malinich[1]
V. I. Mesyura[1]
I. R. Arseniuk[1]

Analysis of Traffic Usage by Scanning Computer Networks with Different Versions of Nmap

[1]Vinnytsia National Technical University

Currently, there are a large number of tools for network monitoring and diagnostics which can be used for various purposes, both authorized persons and other persons who may have unauthorized access to the network. One of them is the Nmap network scanner. The Nmap is a powerful open source network scanning application. Currently it is almost indispensable tool for diagnosing network operation, detecting fault network configurations, and also helps in finding vulnerabilities within the network. The Nmap network scanner is being improved by its author and independent developers. Nowadays the team of this application expands its functionality and improves existing tools that significantly affects its network behavior.

Sniffers are usually used to detect scanning activity on the network, which captures packets passing through them. At this time, they and the intrusion detection software can be used to identify the hosts that perform the scan. However, the task of identifying the software that scans the network and its versions remain relevant. The article considers the solution of the task on the example of the network scanner Nmap.

The use of traffic by different versions of the Nmap during different scanning phases is considered. Differences in the operation of the program arise as a result of updating the current code of this application and its scanning scripts. In most cases, the code does not change significantly within one majority version. However, even minor changes can affect the software identification process when analyzing network traffic. The work of the following versions of Nmap is analyzed: 7.01, 7.60, 7.80.

Keywords: Nmap, network traffic, sniffer, network scanning, intrusion detection system.

Malinich Illia P. — Assistant Lecturer of the Chair of Computer Sciences, e-mail: malinich@vntu.edu.ua ;
Mesyura Volodymyr I. — Cand. Sc., Associate Professor, Professor of the Chair of Computer Sciences, e-mail: mesyura@vntu.edu.ua;
Arseniuk Ihor R. — Cand. Sc. (Eng.), Associate Professor of the Chair of Computer Sciences

И. П. Малинич[1]
В. И. Месюра[1]
И. Р. Арсенюк[1]

Анализ использования трафика при сканировании компьютерных сетей различными версиями Nmap

[1]Винницкий национальный технический университет

На сегодня существует большое количество инструментов для мониторинга и диагностики сети, которые могут использоваться с различными целями, как уполномоченными на это лицами, так и другими лицами, которые могут осуществлять несанкционированный доступ к сети. Одним из них является сетевой сканер Nmap. Сетевой сканер Nmap — это достаточно мощное приложение сетевого сканирования с открытым исходным кодом. На сегодня он является почти незаменимым инструментом для диагностики работы сетей и выявления ошибочных сетевых конфигураций, а также помогает при поиске уязвимостей внутри сети. Сетевой сканер Nmap постоянно совершенствуется его авторами и независимыми разработчиками. Поэтому с течением времени это приложение расширяет свой функционал и улучшает уже существующие инструменты. Однако это существенно влияет на его поведение в сети.

Для выявления сканирующей активности в сети обычно используются сниферы, которые фиксируют пакеты, проходящие через них. На сегодня с их помощью и вспомогательного программного обеспечения для обнаружения вторжений можно выполнять идентификацию хостов, которые осуществляют сканирование. Однако остаются актуальными задачи идентификации программного обеспечения, сканирующего сеть, и его версий. В статье рассматривается решение задачи на примере сетевого сканера Nmap.

Рассмотрено использование трафика различными версиями сетевого сканера Nmap при выполнении различных фаз сканирования. Различия в работе программы возникают в результате обновления текущего кода этого приложения и его скриптов сканирования. В большинстве случаев код в пределах одной мажоритарной версии не претерпит существенных изменений. Однако даже незначительные изменения могут повлиять на процесс идентификации программного обеспечения при анализе сетевого трафика. Проанализирована работа таких версий Nmap: 7.01, 7.60, 7.80.

Ключевые слова: Nmap, сетевой трафик, снифер, сканирования сетей, система обнаружения вторжений.

Малинич Илья Павлович — ассистент кафедры компьютерных наук, e-mail: malinich@vntu.edu.ua ;
Месюра Владимир Иванович — канд. техн. наук, доцент, профессор кафедры компьютерных наук, e-mail: mesyura@vntu.edu.ua;
Арсенюк Игорь Ростиславович — канд. техн. наук, доцент кафедры компьютерных наук