УДК 004.457

Д. О. Данчук
М. Г. Тарновський

СИСТЕМА МОНІТОРИНГУ ТА АНАЛІЗУ МЕРЕЖЕВОГО ТРАФІКУ

Вінницький національний технічний університет

Анотація
У роботі досліджуються методи захоплення мережевого трафіку за допомогою Raw Socket та алгоритми розбору пакетів відповідно до рівнів моделі TCP/IP. Запропоновані підходи сприяють виявленню аномалій і забезпеченню безпеки мережі.
Ключові слова: моніторинг, аналіз, рівні мережевого трафіку, пакет, у реальному часі.

Abstract
The paper explores methods for capturing network traffic using Raw Socket and algorithms for packet parsing according to the TCP/IP model layers. The proposed approaches facilitate anomaly detection and enhance network security.
Keywords: monitoring, analysis, network traffic layers, packet, real-time.

Вступ

У сучасному світі, де інформація передається мережею з надзвичайною швидкістю, моніторинг та аналіз мережевого трафіку стають ключовими складовими забезпечення безпеки й оптимальної роботи IT-інфраструктури. Ця робота присвячена дослідженню способів захоплення пакетів у середовищі Windows із використанням Raw Socket, а також детальному аналізу їхнього вмісту на рівнях моделі TCP/IP. Зрозуміння структури мережевих пакетів і механізмів їхнього оброблення дозволить створити ефективний інструментарій для виявлення аномалій і контролю за станом мережі.

Метою роботи є вдосконалити, або створити нові алгоритми моніторингу та аналізу мережевого трафіку в режимі реального часу.

Основна частина

Захоплення пакетів у Windows реалізується за допомогою Raw Socket. Необхідно використовувати WSAStartup для ініціалізації Winsock, після чого створюють сокет з параметрами IPv4 і Raw, який спочатку IP-пакети. За потреби можна вказати фільтрацію за певним протоколом (TCP, UDP чи ICMP). Далі сокет прив'язують до локальної адреси, а читання описано через виклик Receive(buffer), який повертає розмір пакета, включно з IP-заголовком. За замовчуванням операція блокувальна, однак для асинхронної роботи можливий неблокувальний режим або BeginReceive/EndReceive [1].

Отримавши байтовий масив із IP-пакетом, переходять до розбору відповідно до рівнів моделі TCP/IP. Raw Socket у Windows починає з мережевого рівня, тобто кадри канального (Ethernet) рівня не надходять. В IP-заголовку запустіть довжину заголовка, загальну довжину пакета, ідентифікатор, прапорці фрагментації, TTL, протокол, контрольну суму, IP-адреси джерела й призначення, а також опції, якщо вони присутні. Далі залежно від поля «Протокол», розпочавши TCP, UDP чи ICMP [2, 3].

Якщо це TCP, у заголовку вказані порти джерела та призначені номери безперервності й підтвердження, довжина заголовка, прапорці (SYN, ACK, FIN тощо), розмір вікна, контрольна сума та, за потреби, Urgent Pointer і опції. Після TCP-заголовка розташовано корисне навантаження, яке містить дані відповідного рівня. Наприклад, для HTTP (порт 80) корисне навантаження складається з рядків ASCII — спочатку перший рядок методом і шляхом, потім заголовки до порожнього рядка, після чого читається тіло відповідно до Content-Length або Transfer-Encoding. Якщо порт 443, корисне навантаження зашифровано TLS/SSL, і без доступу до ключів розшифрувати його неможливо, але перші байти TLS-повідомлень (ClientHello, ServerHello тощо) дають загальну інформацію [3, 4].

У разі заголовок UDP складається з портів джерела й призначення, загальної довжини та контрольної суми. Корисне навантаження починається через вісім байтів від початку UDP-заголовка і

передається на аналіз прикладних протоколів (наприклад, DNS на порту 53, DHCP на 67/68, RTP). Для перших байт ICMP вкажіть тип і код повідомлення (наприклад, Echo Request/Reply), далі контрольна сума й інші поля залежать від типу. Аналіз на прикладному рівні в інтерпретації корисного навантаження відповідно до стандартів кожного протоколу. Для DNS на UDP-пакетах читають 12-байтовий заголовок із ідентифікатором, прапорцями та деякими записами, після чого обробляють секцію запитів і відповідей. SMTP (порт 25) — це текстовий потік команд і відповідей, який дозволяє відстежувати процес надсилання листів і вкладень. FTP (порт 21) також базується на текстовому каналі команди, а для передачі файлів вмикається окремо з'єднання (пасивний режим задається командою PASV, відповідно до якої потрібно парсити адресу та порт). Для вашого HTTPS (порт 443) даний зашифрований, тому без MITM або доступу до сертифікатів неможливо прочитати вміст, але заголовки TLS все можуть бути повідомлені версії [3, 4].

Використання Raw Socket у Windows дає можливість перехоплювати IP-пакети, розбирати їхні заголовки та передавати корисне навантаження на аналіз різних рівнів: IP, транспортного та прикладного. Підхід підходить для моніторингу трафіку самої системи, оскільки Raw Socket не передає кадри Ethernet і не бачить трафік між іншими пристроями в мережі. Для розширеного захоплення в усій локальній мережі слід використовувати WinPcap/Npcap або NDIS-драйвери.

Висновки

У результаті роботи показано, що використання Raw Socket у Windows дозволяє легко захоплювати IP-пакети та розбирати їх на рівнях TCP/IP. Розроблена методика десеріалізації заголовків IP, TCP, UDP та ICMP дозволяє відстежувати ключові параметри мережевих потоків і аналізувати відповідний трафік (HTTP, DNS, SMTP тощо). Запропоновані підходи інструментально прості в реалізації й можуть стати основою для побудови системи виявлення аномалій і контролю мережевої безпеки. У подальшому доцільно розширити захоплення всього локального трафіку й реалізації повноцінного графічного приладу.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Socket Class [Електронний ресурс] – Режим доступу до ресурсу: https://learn.microsoft.com/en-us/dotnet/api/system.net.sockets.socket?view=net-8.0.

2. Лекція 11. Комутація пакетів OSI [Електронний ресурс] – Режим доступу до ресурсу: https://e-tk.lntu.edu.ua/mod/page/view.php?id=3561&forceview=1.

3. Основи та застосування протоколів TCP/IP, повний путівник OSI [Електронний ресурс] – Режим доступу до ресурсу: https://hackyourmom.com/kibervijna/osnovy-ta-zastosuvannya-protokoliv-tcp-ip-povnyj-putivnyk/.

4. Розробка та реалізація мережних протоколів: Навчальний посібник [Електронний ресурс]: навч. посіб. для студ. спеціальності 121 «Інженерія програмного забезпечення» та 126 «Інформаційні системи та технології», спеціалізації «Інженерія програмного забезпечення інформаційно управляючих систем» та «Інформаційне забезпечення робототехнічних систем»/ Б. Ю. Жураківський, І. О. Зенів; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 11,5 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2020. – 462 с.

Данчук Дмитро Олександрович — студент групи 2СП-21б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: danchuk2002@gmail.com.

Науковий керівник: ***Тарновський Микола Геннадійович*** — кандидат технічних наук, доцент кафедри обчислювальної техніки, Вінницького національного технічного університету, м. Вінниця, e-mail: ntarn@vntu.edu.ua.

Danchuk Dmytro O. — student of group 2SP-21b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: danchuk2002@gmail.com.

Supervisor: ***Tarnovskyi Mykola H.*** — Ph.D., Associate Professor of the Computer Engineering Department of the Vinnytsia National Technical University, Vinnytsia, e-mail: ntarn@vntu.edu.ua.