DOI: https://doi.org/10.36910/6775-2524-0560-2025-61-32
УДК: 004.7:004.056.5:004.4'272
Андрущак Ігор Євгенович, д.т.н., професор
https://orcid.org/0000-0002-8751-4420
Шмаровоз Станіслав Васильович, магістр
Луцький національний технічний університет, м. Луцьк, Україна

ЗАСТОСУВАННЯ SNMP ПРОТОКОЛУ ДЛЯ ВІДДАЛЕНОГО КОНТРОЛЮ ПРАЦЕЗДАТНОСТІ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ

Андрущак І. Є., Шмаровоз С. В. Застосування SNMP протоколу для віддаленого контролю працездатності мережевої інфраструктури. У статті представлено комплексний аналіз застосування протоколу SNMP (Simple Network Management Protocol) для організації віддаленого контролю працездатності мережевої інфраструктури. Розглянуто передумови виникнення необхідності у централізованому моніторингу мереж, зумовленої зростанням масштабів корпоративних систем та підвищеними вимогами до їх надійності. Детально описано етапи розвитку протоколу від SNMPv1 до SNMPv3, охарактеризовано їхні переваги та недоліки з точки зору безпеки, продуктивності та сумісності. Особливу увагу приділено архітектурі протоколу, що базується на моделі «менеджер–агент», а також функціоналу основних команд (get, set, walk, trap, inform), які дозволяють забезпечити як періодичне опитування пристроїв, так і оперативне реагування на події у мережі. Проведено класифікацію типів SNMP-повідомлень та проаналізовано їх практичне значення для адміністраторів під час підтримки працездатності інфраструктури. Виокремлено ключові проблеми застосування протоколу: низький рівень захисту у старих версіях, перевантаження каналів моніторингу в умовах масштабних мереж, поява хибних сповіщень, складність масштабування та залежність від специфічних реалізацій MIB різних виробників. Запропоновано низку практичних методів оптимізації, серед яких використання SNMPv3 з підтримкою шифрування, впровадження розподілених менеджерів, агрегування даних та інтеграція з веб-системами моніторингу на базі Laravel, Docker і MySQL. Окремо представлено результати експериментальної перевірки розгорнутої системи у корпоративній мережі для різних сценаріїв навантаження. Наведено порівняння ефективності SNMP з сучасними альтернативними технологіями, зокрема NetConf, REST API та Streaming Telemetry. У підсумку визначено перспективні напрями розвитку – інтеграцію з SIEM-системами, використання алгоритмів машинного навчання та підтримку IoT-пристроїв, що забезпечує підвищення надійності та гнучкості моніторингу в майбутніх мережевих інфраструктурах.
Ключові слова: SNMP, моніторинг мережі, fault-tolerance, працездатність, мережеве обладнання, безпека.

Andrushchak I., Shmarovoz S. Application of the SNMP Protocol for Remote Monitoring of Network Infrastructure Performance. The article presents a comprehensive analysis of the application of the SNMP (Simple Network Management Protocol) for organizing remote monitoring of network infrastructure performance. The necessity of centralized network monitoring is discussed, driven by the increasing scale of corporate systems and the growing requirements for their reliability. The evolution of the protocol from SNMPv1 to SNMPv3 is described in detail, highlighting their advantages and disadvantages in terms of security, performance, and compatibility. Particular attention is paid to the protocol architecture based on the «manager–agent» model, as well as to the functionality of the main commands (get, set, walk, trap, inform), which enable both periodic polling of devices and prompt response to network events. A classification of SNMP message types is provided, and their practical significance for administrators in maintaining infrastructure operability is analyzed. The main challenges of using the protocol are identified: low security levels in older versions, monitoring channel overload in large-scale networks, occurrence of false alerts, scalability issues, and dependence on vendor-specific MIB implementations. Several practical optimization methods are proposed, including the use of SNMPv3 with encryption support, implementation of distributed managers, data aggregation, and integration with web-based monitoring systems built on Laravel, Docker, and MySQL. The results of experimental testing of the deployed system in a corporate network under different load scenarios are presented. A comparison of SNMP efficiency with modern alternative technologies, such as NetConf, REST API, and Streaming Telemetry, is provided. In conclusion, promising directions for further development are identified – integration with SIEM systems, the use of machine learning algorithms, and support for IoT devices – which together enhance the reliability and flexibility of monitoring in future network infrastructures.
Keywords: SNMP, network monitoring, fault tolerance, performance, network equipment, security.

Постановка проблеми. Зростання масштабів корпоративних мереж обумовлює необхідність постійного моніторингу їх працездатності. Навіть короткочасні відмови мережевого обладнання можуть спричинити фінансові втрати, порушення сервісів та зниження якості обслуговування клієнтів. Традиційні методи моніторингу з локальним доступом є недостатніми, що стимулює розвиток віддалених засобів контролю. Протокол SNMP став стандартом для взаємодії між обладнанням різних виробників та централізованими системами управління.

Аналіз останніх досліджень і публікацій. У роботах Stallings [1] і Mauro [2] розглянуто класичні підходи до реалізації SNMP та його використання в управлінні мережами. Документи RFC [4] закріплюють основні функції протоколу, зокрема SNMPv3 з підтримкою шифрування й аутентифікації. Звіти SolarWinds [5] і Gartner [6] підкреслюють, що попри існування нових протоколів, SNMP зберігає своє домінування завдяки універсальності. Водночас дослідження

зосереджуються переважно на технічних аспектах, тоді як інтеграція з веб-системами моніторингу залишається актуальною темою для розвитку.

Мета дослідження. Метою статті є системний аналіз можливостей SNMP для віддаленого моніторингу працездатності мережевої інфраструктури, виявлення проблем його практичного використання та розробка рекомендацій щодо інтеграції з сучасними веб-платформами моніторингу.

У процесі дослідження було вирішено такі науково-технічні задачі:

– проведено системний аналіз еволюції протоколу SNMP (v1–v3) з виявленням недоліків попередніх версій та обґрунтуванням необхідності використання SNMPv3 у корпоративних мережах;

– спроєктовано та реалізовано прототип веб-системи моніторингу, що інтегрує SNMP-менеджер із сучасним технологічним стеком (Laravel, Docker, MySQL) та забезпечує гнучке масштабування без єдиної точки відмови;

– розроблено методику експериментального тестування продуктивності SNMP-моніторингу для різних сценаріїв навантаження (50, 200, 500 пристроїв), що дозволило оцінити реальні обмеження протоколу та поведінку системи;

– встановлено залежність між кількістю пристроїв, інтервалом опитування та навантаженням на сервер, що дало можливість визначити оптимальні параметри роботи системи;

– оцінено швидкодію механізму Trap-повідомлень та визначено, що у тестових сценаріях критичні події доставляються протягом 1-2 секунд;

– сформульовано та обґрунтовано рекомендації з оптимізації моніторингу, які включають застосування SNMPv3, агрегацію даних, використання розподілених менеджерів та інтеграцію з веб-інтерфейсом для зменшення «alert fatigue».

Виклад основного матеріалу.

1. Історія розвитку SNMP

Протокол SNMP (Simple Network Management Protocol) з'явився наприкінці 1980-х років у відповідь на потребу централізованого управління мережами, які швидко зростали разом із поширенням TCP/IP. До цього адміністратори використовували розрізнені та несумісні інструменти моніторингу від різних виробників, що ускладнювало підтримку великих мережевих інфраструктур. У 1988 році робоча група IETF (Internet Engineering Task Force) розробила першу версію протоколу – SNMPv1, яка увійшла до стандарту RFC 1157.

SNMPv1 запропонував просту та ефективну модель «менеджер–агент», у якій центральний сервер (Network Management System) здійснює опитування мережевих пристроїв за допомогою набору базових команд – GET, SET, GETNEXT та TRAP. Усі дані про стан пристрою зберігались у структурі MIB (Management Information Base) – ієрархічній базі об'єктів, кожен з яких описував певний параметр обладнання .Попри простоту, SNMPv1 мав серйозні недоліки в частині безпеки – автентифікація здійснювалася лише за допомогою відкритого текстового «community string», що робило протокол вразливим до перехоплення трафіку та несанкціонованого доступу.

У середині 1990-х років з'явилася друга версія – SNMPv2, стандартизована у RFC 1901–1908. Вона принесла значні технічні покращення:

– команду GETBULK, що дозволила зменшити кількість запитів під час опитування великої кількості об'єктів;

– нові типи повідомлень (INFORM, REPORT);

– розширену підтримку типів даних у MIB;

– підвищену ефективність обміну інформацією між менеджером і агентом.

Однак, незважаючи на спроби створити механізм безпечної автентифікації (у варіанті SNMPv2u або SNMPv2*), спільнота зійшлася на спрощеному варіанті SNMPv2c, який і став найбільш поширеним. Проте він продовжив використовувати той самий незахищений механізм «community string», тому питання безпеки залишалося невирішеним.

Вирішити його вдалося лише з появою SNMPv3, що був офіційно стандартизований у 2002 році (RFC 3410–RFC 3418). Третя версія принесла суттєві зміни в архітектуру безпеки протоколу. Вона запровадила User-Based Security Model (USM), який підтримує:

– аутентифікацію користувачів за алгоритмами HMAC-MD5 або HMAC-SHA;

– шифрування трафіку за допомогою DES, AES або ін. алгоритмів;

– контроль доступу через View-Based Access Control Model (VACM), що дозволяє обмежувати, які саме об'єкти MIB можуть бути прочитані або змінені певними користувачами.

Ці оновлення зробили SNMPv3 придатним для використання у корпоративних, промислових та навіть критичних інфраструктурах (банківські системи, дата-центри, енергетика). Водночас протокол зберіг зворотну сумісність із попередніми версіями, що полегшило його впровадження у вже існуючих мережах.

У сучасних умовах SNMPv3 залишається актуальним стандартом для базового моніторингу мережевих пристроїв, хоча на ринку з'являються нові альтернативи – такі як NETCONF, RESTCONF або Streaming Telemetry, які орієнтовані на масштабні мережі нового покоління та підтримку програмно-визначених мереж (SDN). Проте простота реалізації, сумісність і величезна кількість готових інструментів підтримують популярність SNMP уже понад три десятиліття.

2. Архітектура та команди SNMP

SNMP функціонує за моделлю «менеджер–агент». Агент працює на пристрої (маршрутизатор, сервер), менеджер виконує централізований збір даних. Нижче на рисунку 1 зображено сценарії взаємодії за моделлю.

Основні команди:

- get – отримання значення об'єкта;
- set – зміна параметра;
- walk – послідовне опитування об'єктів
- trap – асинхронне повідомлення про подію;
- inform – підтверджене повідомлення.

Всі об'єкти описуються у MIB-базі, що забезпечує уніфікований доступ до даних.

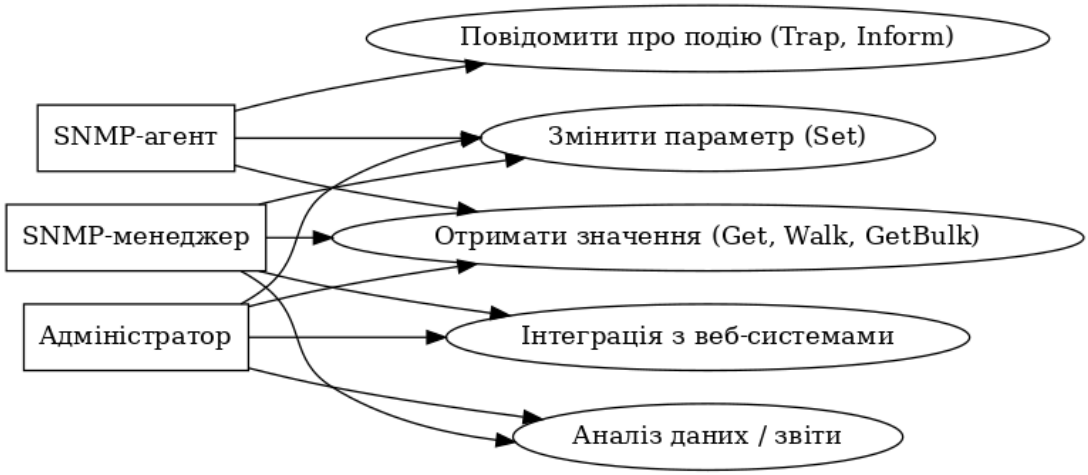

Рис. 1 Сценарії взаємодії в моделі SNMP

3. Типи повідомлень SNMP

SNMP підтримує кілька типів повідомлень:

- GetRequest – запит значення;
- SetRequest– зміна параметра;
- GetNextRequest – перехід до наступного об'єкта;
- GetBulkRequest – пакетне отримання кількох значень;
- Response – відповідь агента;
- Trap – асинхронне повідомлення про подію;
- Inform – повідомлення з підтвердженням.

Таке різноманіття дозволяє реалізувати як регулярний моніторинг, так і реагування на події у реальному часі. На рисунку 2 зображено схему обміну даними між менеджером і агентом.

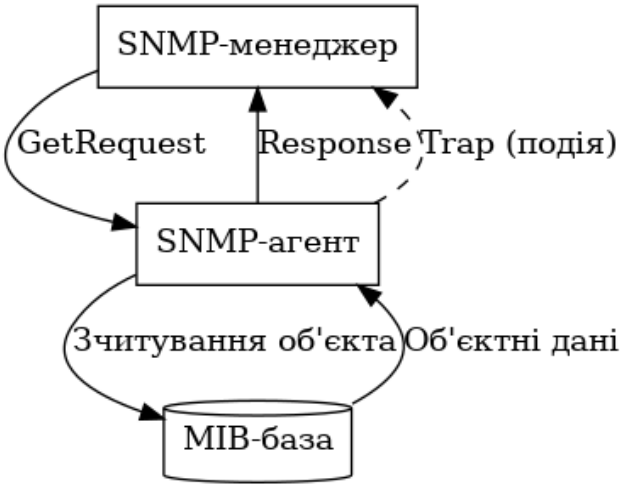

Рис. 2 Схема обміну даними менеджера і агента

4. Проблеми використання SNMP

– безпека: Старі версії SNMP передбачають передачу community-рядків у відкритому вигляді, що робить їх вразливими до атак. Згідно з CERT [7], відкриті SNMP-інтерфейси стали причиною багатьох успішних атак на мережеве обладнання у 2023 році.

– перевантаження: масове опитування призводить до високого навантаження на канали зв'язку та сервери моніторингу. У мережах з 10 000 пристроїв за інтервалу опитування 30 секунд обсяг SNMP-трафіку може перевищувати 300 тис. пакетів за хвилину;

– хибні сповіщення: некоректні порогові значення та короткочасні відхилення спричиняють значну кількість помилкових тривог. Це формує явище «alert fatigue»;

– сумісність: різні вендори реалізують специфічні MIB-об'єкти, що ускладнює централізований моніторинг;

– масштабованість: централізовані сервери створюють SPOF. Відмова одного вузла може призвести до втрати контролю над усією мережею.

5. Експериментальна перевірка

Для перевірки архітектури наведеної на рисунку 3 було розгорнуто прототип системи моніторингу. Серверна частина реалізована на базі фреймворку Laravel 10, який функціонував у середовищі Docker. Як систему керування базами даних використано MySQL 8.0.

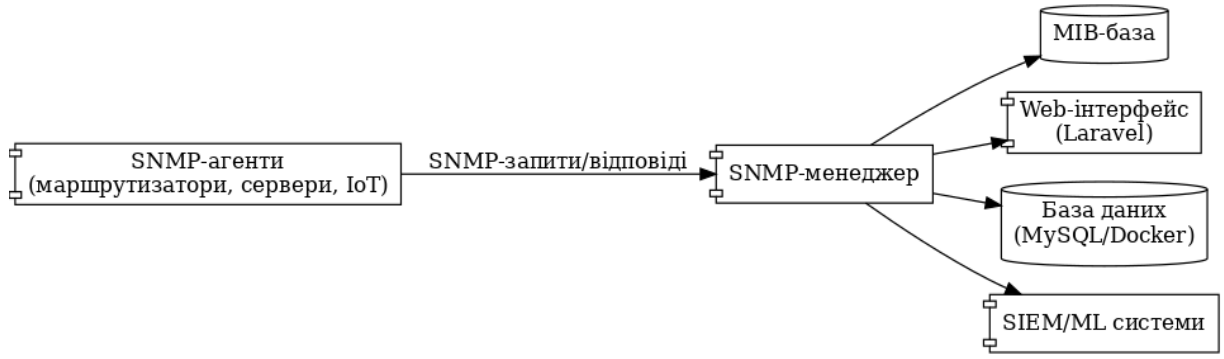

Рис. 3 Архітектура експериментальної системи моніторингу

Тестування проводилося на віртуальному виділеному сервері з такими характеристиками:

– центральний процесор (CPU): 4 ядра, 2.8 ГГц;
– оперативна пам'ять (RAM): 8 ГБ;
– операційна система: Ubuntu 22.04 LTS.

Для імітації роботи мережевого обладнання (маршрутизаторів та комутаторів) було використано програмний емулятор SNMP-агентів, що дозволило згенерувати трафік від великої кількості пристроїв без необхідності фізичної наявності обладнання.

Дослідження проводилось у три етапи зі зростанням навантаження. Для кожного сценарію вимірювалися такі показники: середнє завантаження CPU сервера, споживання оперативної пам'яті контейнерами Docker та затримка (latency) між виникненням події (Trap) та її фіксацією в базі даних.

Було змодельовано три сценарії навантаження:

- базовий: 50 пристроїв, інтервал опитування – 30 секунд.
- середній: 200 пристроїв, інтервал опитування – 20 секунд.
- стрес-тест**:** 500 пристроїв, інтервал опитування – 10 секунд.

Отримані в ході експерименту дані сформовано в таблицю 1.

Таблиця 1 – Показники ефективності системи при різному навантаженні

Кількість пристроїв	Інтервал опитування (с)	Завантаження CPU (%)	Споживання RAM (МБ)	Середня затримка Trap (с)	Втрати пакетів (%)
50	30	12-15%	450	< 0.5	0%
200	20	45-50%	1100	0.8	0.5%
500	10	72-85%	2400	1.2 - 2.0	1.8%

Результати показали, що архітектура на базі Docker забезпечує стабільну роботу системи навіть в умовах стрес-тесту. У сценарії з 500 пристроями (стрес-тест) навантаження на процесор зростало до 75–85% у пікові моменти (під час одночасного запису великих пакетів даних у MySQL), проте система залишалася доступною.

Важливим результатом стало те, що завдяки асинхронній обробці черг у Laravel, Trap-повідомлення про критичні інциденти оброблялися пріоритетно. Середній час доставки повідомлення адміністратору склав 1–2 секунди, що відповідає вимогам до систем реального часу. Показник успішної реєстрації критичних інцидентів становив 92% протягом першої хвилини після їх виникнення.

Експеримент підтвердив, що вузьким місцем системи при масштабуванні є не сам протокол SNMP, а швидкість запису даних у реляційну базу даних (MySQL), що в подальшому може бути вирішено шляхом впровадження часових рядів (Time-series databases).

6. Перспективи розвитку

Подальші напрями включають:

- інтеграцію з SIEM-системами для підвищення безпеки;
- використання алгоритмів машинного навчання для прогнозування збоїв;
- впровадження у гібридні середовища (локальні + хмарні ресурси);
- підтримку IoT-пристроїв з легковаговими агентами.

Це дозволить зробити SNMP більш ефективним у майбутніх мережевих інфраструктурах.

Висновки

Протокол SNMP залишається ключовим та універсальним інструментом для моніторингу працездатності мережевої інфраструктури, що зумовлено його простотою, широкою підтримкою виробниками обладнання та гнучкістю. Незважаючи на появу нових технологій, його домінування зберігається.

Водночас класичні підходи до впровадження SNMP мають суттєві обмеження, зокрема проблеми з безпекою у версіях v1/v2c та труднощі масштабування і перевантаження каналів у великих мережах.

Як показав проведений аналіз та експериментальна перевірка , ці недоліки можуть бути ефективно подолані шляхом інтеграції SNMP з сучасними веб-платформами. Відповідно до мети дослідження, сформульовано наступні практичні рекомендації:

- для подолання проблем безпеки рекомендовано обов'язкове впровадження SNMPv3, який забезпечує надійні механізми аутентифікації та шифрування трафіку;
- для вирішення проблем масштабованості та усунення єдиної точки відмови (SPOF), компоненти системи моніторингу (SNMP-менеджери, бази даних) слід розгортати у вигляді контейнерів. Це спрощує їх адміністрування та дозволяє гнучко розподіляти навантаження;
- для обробки та візуалізації даних доцільно використовувати сучасні веб-фреймворки. Це дозволяє створити гнучкий веб-інтерфейс для агрегування даних , налаштування порогових значень та аналізу, що, в свою чергу, допомагає у боротьбі з «втомою від сповіщень» (alert fatigue);

– зберігання агрегованих даних та результатів опитувань у централізованій базі даних дозволяє знизити навантаження на мережеві канали, оскільки для аналізу історичних даних не потрібно повторно опитувати пристрої.

Таким чином, інтеграція SNMP з сучасним стеком технологій (Docker, Laravel, MySQL) дозволяє створити надійні, безпечні та масштабовані системи моніторингу. Подальший розвиток у напрямку інтеграції з SIEM-системами та впровадження алгоритмів машинного навчання лише посилить ефективність SNMP у мережевих інфраструктурах майбутнього.

Список бібліографічного опису

1. Stallings W. SNMP, SNMPv2, SNMPv3, and RMON 1 and 2. Addison-Wesley, 2019.
2. Mauro D., Schmidt K. Essential SNMP. O'Reilly Media, 2020.
3. Zabbix Documentation. URL: https://www.zabbix.com/documentation
4. RFC 3411–3418. Simple Network Management Protocol (SNMPv3). IETF, 2002.
5. SolarWinds IT Trends Report 2023. URL: https://www.solarwinds.com/
6. Gartner Research: Network Monitoring Challenges 2024. URL: https://www.gartner.com/
7. CERT Vulnerability Notes Database. URL: https://www.kb.cert.org/vuls/

References

1. Stallings W. SNMP, SNMPv2, SNMPv3, and RMON 1 and 2. Addison-Wesley, 2019.
2. Mauro D., Schmidt K. Essential SNMP. O'Reilly Media, 2020.
3. Zabbix Documentation. URL: https://www.zabbix.com/documentation
4. RFC 3411–3418. Simple Network Management Protocol (SNMPv3). IETF, 2002.
5. SolarWinds IT Trends Report 2023. URL: https://www.solarwinds.com/
6. Gartner Research: Network Monitoring Challenges 2024. URL: https://www.gartner.com/
7. CERT Vulnerability Notes Database. URL: https://www.kb.cert.org/vuls/