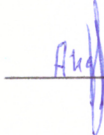


Міністерство освіти і науки України
Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки

ПОЯСНЮВАЛЬНА ЗАПИСКА

до випускної кваліфікаційної роботи магістра
на тему: «Методи та технології кібербезпекового
моніторингу мережі на базі Zabbix»

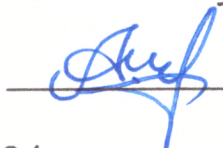
Виконав студент 2-го курсу групи КБм-23-1
спеціальності 125 «Кібербезпека та захист інформації»

 А.О. Анчис

Керівник доцент кафедри комп'ютерної інженерії та
кібербезпеки, кандидат педагогічних наук, доцент

 О.С. Головня

Рецензент доцент кафедри комп'ютерної інженерії та
кібербезпеки, кандидат технічних наук

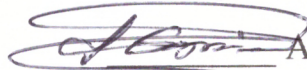
 О.О. Шелуха

Житомир – 2024

Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки
Спеціальність 125 «Кібербезпека та захист інформації»
Освітня програма «Кібербезпека»

ЗАТВЕРДЖУЮ

Завідувач
кафедри комп'ютерної
інженерії та кібербезпеки

 Андрій Єфіменко

17 вересня 2024 р.

ЗАВДАННЯ

на випускню кваліфікаційну роботу магістра

Студент: Анчис Андрій Олександрович

Тема роботи: **«Методи та технології кібербезпекового моніторингу мережі на базі Zabbix»**

затверджена Наказом університету від 17 вересня 2024 р. № 479/с

Термін здачі студентом закінченої роботи 23 грудня 2024 р.

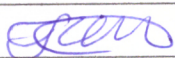
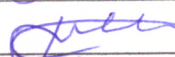
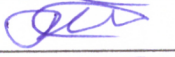
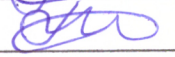
Вихідні дані роботи (зазначається предмет і об'єкт дослідження):

1. Топологія локальної мережі;
2. Система моніторингу Zabbix

Обладнання та програмне забезпечення:

1. Гіпервізор Oracle VirtualBox;
2. Мережевий емулятор GNS3;
3. ОС Ubuntu 24.04;
4. ОС Windows 10;
5. ОС Windows Server 2019.
6. Zabbix (v.7.0.6)

Консультанти з випускної кваліфікаційної роботи із зазначенням розділів, що їх стосуються

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	О.С. Головня	17.09.2024 	14.10.2024 
2	О.С. Головня	15.10.2024 	16.11.2024 
3	О.С. Головня	17.11.2024 	20.12.2024 

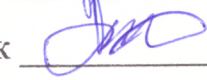
Студент  А.О. Анчис

Керівник  О.С. Головня

Календарний план

№ з/п	Назви етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Аналіз отриманого та уточнення завдання	17.09.2024	Виконано
2	Аналіз літературних джерел	25.09.2024	Виконано
3	Аналіз об'єкта дослідження	02.10.2024	Виконано
4	Постановка задачі дослідження	10.10.2024	Виконано
5	Розробка топології	18.10.2024	Виконано
6	Виконання базових налаштувань мережі	25.10.2024	Виконано
7	Налаштування системи кібербезпекового моніторингу Zabbix	05.11.2024	Виконано
8	Тестування і перевірка роботи мережі та системи кібербезпекового моніторингу	15.11.2024	Виконано
9	Написання пояснювальної записки	9.12.2024	Виконано
10	Оформлення презентації	16.12.2024	Виконано
11	Захист	23.12.2024	Виконано

Студент  А.О. Анчис

Керівник  О.С. Головня

АНОТАЦІЯ

Пояснювальна записка до випускної кваліфікаційної роботи на тему «Методи та технології кібербезпекового моніторингу мережі на базі Zabbix» складається зі вступу, трьох розділів, висновку, списку використаних джерел та додатків. Текстова частина викладена на 74 сторінках друкованого тексту. Пояснювальна записка має 6 сторінок додатків. Список використаних джерел містить 20 найменувань і займає 2 сторінки. В роботі наведено 77 рисунків та 4 таблиць. Загальний обсяг роботи – 80 сторінок.

Метою роботи є дослідження та практична реалізація методів та технології кібербезпекового моніторингу мережі на базі Zabbix.

В ході роботи було розглянуто теоретичні аспекти систем моніторингу, їхнє призначення, основні функціональні можливості та проведено аналіз відомих існуючих рішень моніторингу в ході яких визначено, що Zabbix є багатофункціональним рішенням, яке забезпечує моніторинг пристроїв, серверів і додатків із високою гнучкістю у налаштуванні. Особливу увагу приділено Zabbix, його структурі, архітектурі та основним компонентам. Виконано розробку топології комп'ютерної мережі з використанням трирівневої ієрархічної моделі та проведено розподіл адресації для пристроїв мережі. Спроектовано роботу системи кібербезпекового моніторингу Zabbix. Виконано підбір обладнання для реалізації проєкту мережі. Проведено налаштування мережі з використанням мережевого емулятора GNS3, що включало налагодження маршрутизатора, комутатора, Windows Server та кінцевих пристроїв. Виконано розгортання і налаштування системи кібербезпекового моніторингу Zabbix, встановлено агенти на кінцевих пристроях. Проведено тестування роботи мережі та системи Zabbix, що показало повну працездатність проєкту та виконання всіх поставлених завдань.

КЛЮЧОВІ СЛОВА: МОНІТОРИНГ, БЕЗПЕКА, ЗАХИСТ, ПОДІЇ, МЕРЕЖА, ТОПОЛОГІЯ, СПОВІЩЕННЯ.

ANNOTATION

The explanatory note for the final qualification work on the topic «Methods and Technologies of Cybersecurity Network Monitoring Based on Zabbix» consists of an introduction, three chapters, a conclusion, a list of references, and appendices. The text portion is presented on 74 pages of printed text. The explanatory note includes 6 pages of appendices. The list of references contains 20 entries and occupies 2 pages. The work includes 77 figures and 4 tables. The total volume of the work is 80 pages.

The purpose of the work is to study and practically implement methods and technologies of cybersecurity network monitoring based on Zabbix.

During the work, theoretical aspects of monitoring systems, their purpose, main functionalities, and an analysis of well-known existing monitoring solutions were considered. It was determined that Zabbix is a multifunctional solution providing monitoring of devices, servers, and applications with high configuration flexibility. Particular attention was paid to Zabbix, its structure, architecture, and main components. A computer network topology was developed using a three-tier hierarchical model, and addressing was allocated for network devices. The operation of the Zabbix cybersecurity monitoring system was designed. Equipment for network implementation was selected. The network was configured using the GNS3 network emulator, including the setup of a router, switch, Windows Server, and end devices. The deployment and configuration of the Zabbix cybersecurity monitoring system were carried out, and agents were installed on end devices. Testing of the network and the Zabbix system confirmed the full operability of the project and the fulfillment of all tasks.

KEYWORDS: MONITORING, SECURITY, PROTECTION, EVENTS, NETWORK, TOPOLOGY, NOTIFICATIONS.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ СИСТЕМ МОНІТОРИНГУ	10
1.1 Призначення та функціональні можливості систем моніторингу.....	10
1.2 Аналіз існуючих рішень систем моніторингу	15
1.3 Структура та функціональні можливості Zabbix	23
РОЗДІЛ 2 ПРОЄКТУВАННЯ СИСТЕМИ МОНІТОРИНГУ МЕРЕЖІ З ВИКОРИСТАННЯМ ZABBIX	28
2.1. Розробка топології комп'ютерної мережі.....	28
2.2 Організація роботи Zabbix.....	33
2.3 Вибір обладнання.....	35
РОЗДІЛ 3 ПОБУДОВА, НАЛАГОДЖЕННЯ ТА ТЕСТУВАННЯ РОБОТИ МЕРЕЖІ.....	39
3.1 Базове налаштування мережі	39
3.2 Встановлення системи моніторингу Zabbix	48
3.3 Налаштування і перевірка роботи системи моніторингу Zabbix.....	53
ВИСНОВКИ	72
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	73
ДОДАТКИ	75

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

DDoS – Distributed Denial of Service

IDS/IPS – Intrusion Detection System / Intrusion Prevention System

EDR – Endpoint Detection and Response

GDPR – General Data Protection Regulation

PCI DSS – Payment Card Industry Data Security Standard

SNMP – Simple Network Management Protocol

JMX – Java Management Extensions

IPMI – Intelligent Platform Management Interface

API – Application Programming Interface

SMS – Short Message Service

DevOps – Development and Operations

SQL – Structured Query Language

IIS – Internet Information Services

WMI – Windows Management Instrumentation

DRAM – Dynamic Random-Access Memory

LAN – Local Area Network

NAT – Network Address Translation

DNS – Domain Name System

PSK – Pre-Shared Key

ВСТУП

Актуальність. Моніторинг у сфері кібербезпеки набуває ключового значення в умовах постійного зростання кількості та складності кібератак. Із розвитком інформаційних технологій сучасні організації стикаються з новими викликами щодо захисту своїх ІТ-інфраструктур, що включають мережі, сервери, додатки та бази даних. Системи моніторингу стають невід'ємною частиною забезпечення кібербезпеки, оскільки дозволяють виявляти загрози на ранніх етапах, реагувати на аномалії та мінімізувати ризики витоків даних чи несанкціонованого доступу.

Ефективні рішення для моніторингу дозволяють в реальному часі відстежувати стан мережевих і системних компонентів, забезпечувати аудит безпеки, аналізувати журнали та виявляти ознаки вторгнень або зловмисних дій. Своєчасне виявлення потенційних загроз і реакція на них можуть запобігти суттєвим фінансовим втратам і репутаційним збиткам, які часто супроводжують кібератаки.

З огляду на постійно змінювані загрози та їхню складність, системи моніторингу, такі як Zabbix, SolarWinds, Datadog та інші, стають основними інструментами для організацій у боротьбі з кіберзагрозами. Вони дозволяють комплексно оцінювати стан безпеки, автоматизувати виявлення загроз та впроваджувати превентивні заходи, що забезпечує надійний захист ІТ-інфраструктур.

Метою кваліфікаційної роботи є використання методів та технологій кібербезпекового моніторингу мережі на базі рішення Zabbix для своєчасного виявлення, аналізу кіберзагроз для ІТ-інфраструктури організації та запобігання їм.

Мета передбачає виконання наступних завдань:

- провести аналіз сучасних систем моніторингу, визначивши їхні переваги, недоліки та можливості застосування;

- дослідити структуру, функціональні можливості та основні принципи роботи системи Zabbix у контексті моніторингу IT-інфраструктури;
- розробити та обґрунтувати підходи до налаштування Zabbix для моніторингу мережі та виявлення загроз безпеці;
- виконати практичне впровадження Zabbix для обраної мережевої топології, описати процес налаштування та оцінити ефективність системи.

Об'єктом дослідження є системний моніторинг як складова забезпечення безпеки в IT-інфраструктурі організацій.

Предметом дослідження є функціональні можливості та архітектура системи моніторингу Zabbix. Досліджується її здатність забезпечувати безперервний моніторинг IT-інфраструктури, виявляти потенційні загрози, аналізувати аномальні події та реагувати на інциденти. Окрему увагу приділено налаштуванню тригерів, системі сповіщень, інтеграції з іншими системами безпеки, а також використанню Zabbix для запобігання кіберзагрозам.

РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ СИСТЕМ МОНІТОРИНГУ

1.1 Призначення та функціональні можливості систем моніторингу

Системи моніторингу стали невід'ємною частиною кібербезпеки, забезпечуючи постійний контроль за станом IT-інфраструктури, відстежуючи ключові параметри роботи систем, мереж і сервісів. Основною метою є не лише забезпечення безперебійної роботи, але й запобігання загрозам, виявлення аномалій і своєчасне реагування на інциденти безпеки [1]. У сучасному світі, де кібератаки стають складнішими, системи моніторингу відіграють ключову роль у захисті даних і сервісів.

1. Моніторинг продуктивності та ресурсів.

Системи моніторингу продуктивності й ресурсів виконують контроль за технічними аспектами роботи IT-інфраструктури. До цих аспектів належать наступні:

- Відстеження використання процесорів на серверах і клієнтських пристроях. Якщо процесор перевантажений, це може спричинити уповільнення роботи системи. У разі виявлення таких проблем на ранніх етапах стає можливим попередити адміністратора про необхідність втручання.
- Відстеження оперативної пам'яті. Коли пам'ять використовується на критично високому рівні, це може вплинути на продуктивність або навіть призвести до збоїв. Система моніторингу генерує попередження, якщо доступні ресурси пам'яті досягають критичних значень.
- Відслідковується обсяг трафіку в мережі, включаючи вхідні та вихідні дані, пропускну здатність та помилки в роботі мережі. Системи моніторингу дозволяють виявляти "вузькі місця" у мережі, запобігаючи перевантаженням, які можуть стати причиною збоїв або низької продуктивності.
- Системи відстежують використання дискового простору на серверах і сховищах даних. Якщо обсяг вільного простору критично малий, це може спричинити перебої в роботі системи або її зупинку. Постійне відстеження цього дозволяє попереджувати подібні інциденти.

- Деякі системи моніторингу також відстежують температурний режим критичного обладнання. Перегрівання може призвести до виходу з ладу серверів або мережевого обладнання.

2. Моніторинг безпеки.

Моніторинг безпеки спрямований на виявлення потенційних загроз, аномалій або атак у реальному часі. Безпека досягається шляхом аналізу даних із різних джерел, таких як журнали(логи), мережевий трафік і поведінка користувачів [2]. Основні можливості, які надає моніторинг безпеки, є наступними:

- Системи моніторингу здатні виявляти підозрілі активності, такі як спроби несанкціонованого доступу, багаторазові невдалі входи або раптове збільшення використання системних ресурсів. Наприклад, спроба здійснення входу в систему з багатьох IP-адрес може свідчити про атаку брутфорс.

- Деякі системи використовують машинне навчання або поведінковий аналіз для виявлення нетипових дій. Наприклад, система може виявити, що користувач раптово почав використовувати значно більше ресурсів, що може свідчити про компрометацію акаунта.

- Можливе відстеження змін в конфігураціях безпеки, наприклад, відкриття портів, зміни в доступі до файлів або модифікації налаштувань брандмауера. Якщо такі зміни не були санкціоновані, це може свідчити про атаку.

- Аналіз трафіку дозволяє виявляти спроби проникнення через мережу або внутрішньомережеві загрози, як-то спроби сканування портів або DDoS-атаки. Система моніторингу може автоматично реагувати на такі інциденти, обмежуючи доступ або блокуючи підозрілі IP-адреси.

- Відстеження подій та інтеграція з такими системами як Firewall, IDS/IPS та EDR [3]. Це включає моніторинг стану цих систем, фіксацію спроб вторгнення, блокування підозрілих з'єднань і виявлення підозрілих дій, які можуть свідчити про компрометацію мережі чи її компонентів. Аналіз даних із

цих джерел у загальну систему моніторингу дозволяє забезпечити цілісний огляд безпекового стану інфраструктури.

3. Аналіз журналів.

Журнали або логи є одним із основних джерел. В логах фіксуються різноманітні події, що відбуваються в системі, включаючи дії користувачів, доступ до даних, зміни в конфігурації системи та інші операції, рис.1.1. Таким чином стає можливим аналіз цих журналів для виявлення загроз та аномалій.

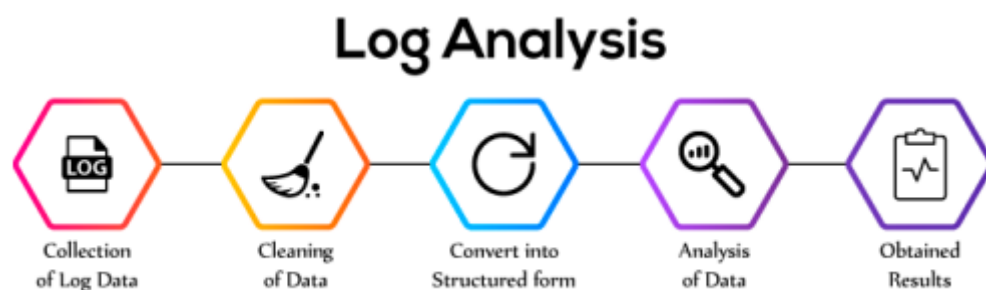


Рисунок 1.1 – Класичні етапи аналізу логів

Основні аспекти аналізу журналів є наступними:

- Збір журналів з багатьох джерел, таких як сервери, мережеві пристрої, програми та бази даних. Це дозволяє отримати загальну картину того, що відбувається в IT-інфраструктурі.
- Інструменти моніторингу дозволяють зіставляти події з різних джерел, щоб визначити взаємозв'язки між ними. Наприклад, виявлення спроби входу в систему може бути поєднано з аналізом мережевого трафіку для підтвердження можливості атаки.
- Завдяки автоматизації аналізу логів система може виявляти шаблони підозрілої поведінки або аномалії, які можуть свідчити про зловмисні дії.
- Деякі системи можуть автоматично виконувати дії у відповідь на події в журналах. Наприклад, якщо виявляється спроба підбору пароля, система може автоматично заблокувати IP-адресу, з якої здійснюється атака.

4. Оповіщення та реагування.

Функція оповіщення є однією з ключових для будь-якої системи моніторингу, оскільки вона дозволяє адміністратору або команді кібербезпеки

оперативно отримувати інформацію про критичні події. Основними характеристиками даного функціоналу є зокрема перелічені нижче:

- Надсилання сповіщень через різні канали — електронну пошту, SMS, месенджери. Це гарантує, що повідомлення буде доставлене вчасно, навіть якщо один з каналів зв'язку недоступний.
- Важливим є налаштування рівнів важливості подій, від інформаційних до критичних, що дозволяє зосередити увагу на найбільш важливих інцидентах.
- Існує можливість автоматичного реагування на певні події. Наприклад, при перевищенні критичного рівня використання ресурсів система може автоматично закрити непотрібні процеси або збільшити ресурси для критичних сервісів.

5. Аналіз тенденцій і звітність.

Окрім моніторингу в реальному часі, важливим аспектом систем моніторингу є їх здатність надавати аналітику на основі довгострокових даних. Ці можливості дозволяють виявляти тенденції у використанні ресурсів, продуктивності системи або безпеці. Аналіз тенденцій надає наступні можливості:

- Аналізуючи дані за певний період часу, можливо виявляти закономірності в роботі інфраструктури, що дозволяє планувати ресурси та запобігати проблемам.
- Виведення інформації того, як змінюється кількість і типи інцидентів безпеки після впровадження нових заходів або оновлень.
- Інструменти моніторингу можуть автоматично генерувати звіти з детальною інформацією про інфраструктуру, події, інциденти безпеки та використання ресурсів. Це корисно для звітування перед керівництвом, аудитами або для планування нових заходів безпеки.

- На основі аналізу минулих даних системи можуть прогнозувати можливі проблеми в майбутньому, наприклад, зростання навантаження на мережу або можливі вразливості.

В умовах сучасних кіберзагроз моніторинг є невід'ємною частиною захисту будь-якої організації. Його значення можна розглядати з кількох точок зору:

1. Моніторинг дозволяє виявляти аномалії та підозрілу активність до того, як вони стануть загрозами. Наприклад, виявлення нетипових патернів трафіку або незвичних дій користувачів може свідчити про підготовку до атаки.

2. Однією з найважливіших функцій є оповіщення про інциденти в реальному часі. Це дозволяє командам безпеки миттєво вживати заходів для мінімізації шкоди.

3. Системи моніторингу можуть бути налаштовані для виявлення нових загроз і вразливостей, що не були раніше відомі. Це критично важливо для захисту від атак нульового дня, коли шкідливий код використовує нові вразливості.

4. У багатьох індустріях існують вимоги щодо збереження безпеки даних і відповідності стандартам (наприклад, GDPR, PCI DSS). Системи моніторингу допомагають організаціям дотримуватися цих стандартів, забезпечуючи відповідність вимогам щодо збереження даних і реагування на інциденти.

Один із реальних прикладів використання системи моніторингу у кібербезпеці можна знайти в банківському секторі. Банки використовують системи моніторингу для контролю за транзакціями та виявлення підозрілої активності. Наприклад, якщо система фіксує, що користувач здійснює численні транзакції в різних частинах світу, це може вказувати на шахрайство. Завдяки цьому можна оперативно заблокувати акаунт, мінімізувавши втрати.

1.2 Аналіз існуючих рішень систем моніторингу

Системи моніторингу є ключовими інструментами забезпечення стабільної роботи інформаційних технологій, особливо у сфері кібербезпеки. В умовах швидкого розвитку цифрової інфраструктури та підвищення складності мережевих середовищ виникає необхідність в ефективному контролі за станом серверів, мереж, додатків та віртуальних платформ. Сучасні системи моніторингу здатні надавати реальний огляд інфраструктури, виявляти потенційні проблеми на ранніх етапах і мінімізувати ризики збоїв.

У контексті кібербезпеки системи моніторингу не лише сприяють своєчасному виявленню технічних несправностей, але й допомагають захистити від загроз та аномалій, що можуть свідчити про кібератаки. Ринок програмного забезпечення пропонує велику кількість рішень для моніторингу, кожне з яких має свої особливості, архітектуру, можливості та сфери застосування.

У цьому пункті буде проведено аналіз найпопулярніших рішень для моніторингу, таких як Zabbix, Nagios, Prometheus, SolarWinds та Datadog. Кожна з цих систем є потужним інструментом для відстеження стану інфраструктури, проте вони мають різні архітектурні підходи, переваги та недоліки, що важливо враховувати при виборі оптимального рішення для конкретних потреб організації.

Zabbix, рис.1.2, це потужна система моніторингу з відкритим кодом, яка дозволяє контролювати сервери, мережеві пристрої, додатки та віртуальні середовища [4]. Її архітектура підтримує як агентний, так і безагентний моніторинг. Zabbix може збирати дані через різні протоколи, такі як SNMP, JMX, IPMI, а також через власний Zabbix Agent. Основними її перевагами є гнучкість налаштувань, висока масштабованість та підтримка численних інтеграцій з іншими системами. Вона добре підходить для великих інфраструктур і дає можливість контролювати велику кількість пристроїв. Однак система може вимагати значних зусиль для налаштування, особливо у великих компаніях, і є ресурсомісткою при великому навантаженні.



Рисунок 1.2 – Система моніторингу Zabbix

Архітектура складається з кількох основних компонентів: Zabbix Server, який керує всією системою, Zabbix Proxy для масштабування і збору даних з віддалених вузлів, а також Zabbix Agent, який встановлюється на цільові сервери для збору метрик [5].

Zabbix має широкі можливості для забезпечення безпеки мережі, такі як: підтримка збирання метрик у реальному часі, створення тригерів і сценаріїв автоматизації, налаштування сповіщень, гнучка візуалізація даних через панель керування.

Переваги:

- Можливість контролювати широкий спектр пристроїв, незалежно від їх типу чи виробника.
- Можна легко розширювати, додаючи нові вузли за допомогою Zabbix Proxy.
- Підтримка інтеграції з іншими інструментами через API або веб-хуки.

Недоліки:

- Для нових користувачів Zabbix може бути складною у налаштуванні, особливо в великих інфраструктурах.
- При великій кількості пристроїв система може споживати значні ресурси.
- Zabbix активно використовується в великих компаніях та державних організаціях для моніторингу серверних ферм, мережевої інфраструктури та віртуальних середовищ.

Nagios, рис.1.3, одна з найстаріших та найпопулярніших систем моніторингу з відкритим кодом. Вона використовується для моніторингу

серверів, мережевих пристроїв і додатків, відома своєю модульною архітектурою та величезною бібліотекою плагінів [6]. В Nagios є можливість легко налаштовувати систему для різних видів обладнання та інфраструктур, а також її активна спільнота користувачів, що постійно розробляє нові плагіни. Однак, система має обмежені засоби візуалізації і потребує додаткових налаштувань для ефективного моніторингу складних середовищ.



Рисунок 1.3 – Система моніторингу Nagios

Nagios має ядро, яке збирає дані від агентів або плагінів, та інтерфейс, який візуалізує стан сервісів. Модульна структура дозволяє використовувати безліч плагінів для різноманітних типів пристроїв і сервісів.

Платформа включає такі функції як: моніторинг мережевих пристроїв, серверів, баз даних, додатків. Підтримка сповіщень про збої в роботі за допомогою електронної пошти або SMS [7].

Переваги:

- Велика бібліотека плагінів дозволяє розширити можливості системи для специфічних завдань.
- Nagios має велику спільноту користувачів, що активно розробляє нові плагіни та розширення.

Недоліки:

- Вбудовані засоби візуалізації досить обмежені, що може вимагати додаткових інструментів для аналітики.
- Для ефективного використання системи потрібно мати досвід налаштування та створення плагінів.

- Nagios використовується в великих підприємствах для моніторингу мережевих інфраструктур, серверних ферм, баз даних та різних ІТ-сервісів.

Prometheus, рис.1.4, це сучасна система моніторингу, розроблена для середовищ з динамічними архітектурами, таких як контейнери та хмарні інфраструктури [8]. Вона спеціалізується на зборі та аналізі часових рядів метрик. Prometheus використовує pull-модель для збору метрик та інтегрується з такими системами, як Kubernetes, що робить її ідеальним вибором для середовищ DevOps. Основною перевагою Prometheus є її гнучка мова запитів PromQL, яка дозволяє робити складні аналітичні запити для виявлення проблем і прогнозування стану системи. Однак вона менш зручна для традиційних серверних середовищ, і для сповіщень потрібно налаштовувати додатковий інструмент Alertmanager.



Рисунок 1.4 – Система моніторингу Prometheus

Prometheus працює з метриками, які зберігаються у вигляді часових рядів. Він використовує pull-підхід, коли система запитує метрики від вузлів внесених до мережі.

На платформі реалізовано автоматичне виявлення нових сервісів, потужна мова запитів PromQL для аналізу даних, інтеграція з іншими системами через експортери, що надає великі можливості у її використанні.

Переваги:

- Prometheus ідеально підходить для моніторингу динамічних хмарних і контейнерних середовищ.
- PromQL дозволяє робити складні аналітичні запити для отримання глибокої інформації про стан систем.

Недоліки:

- Prometheus найкраще працює в сучасних середовищах DevOps, але може бути менш зручним для моніторингу традиційних серверів.
- Для алертів необхідно використовувати Alertmanager або сторонні рішення.
- Prometheus є стандартом в середовищах, які активно використовують хмарні рішення та контейнери, особливо в середовищах DevOps та Kubernetes.

SolarWinds — це комерційна система моніторингу, відома своєю потужністю та глибиною аналізу інфраструктур [9]. Вона призначена для великих підприємств і урядових організацій, оскільки може відстежувати сервери, додатки та віртуальні середовища. Система використовує модульну архітектуру, що дозволяє додавати нові можливості через спеціалізовані модулі. SolarWinds має зручний інтерфейс і пропонує велику кількість інструментів для детального аналізу та сповіщень. Проте її недоліком є висока вартість та складність налаштування, що може потребувати значних зусиль на етапі впровадження.



Рисунок 1.5 – Система моніторингу SolarWinds

SolarWinds використовує модульну архітектуру, де кожен модуль відповідає за конкретні завдання (наприклад, моніторинг мережі, управління логами, моніторинг продуктивності додатків). Усі модулі інтегруються в єдиний інтерфейс для полегшення управління.

Платформа підтримує глибокий аналіз мережевого трафіку, моніторинг стану серверів, віртуалізованих середовищ та додатків, а також управління конфігураціями [10].

Переваги:

- SolarWinds забезпечує високий рівень технічної підтримки та регулярні оновлення.

- Усі аспекти IT-інфраструктури можуть бути моніторинговані в єдиному інтерфейсі.

Недоліки:

- SolarWinds є комерційною платформою, і її ціна може бути занадто високою для малих та середніх підприємств.

- Для максимальної ефективності SolarWinds потребує значних зусиль на етапі впровадження та налаштування.

- SolarWinds широко використовується великими корпораціями та урядовими організаціями для моніторингу великих і складних IT-інфраструктур.

Datadog, рис.1.6, це хмарна платформа для моніторингу, яка поєднує в собі моніторинг інфраструктури, додатків та логів у єдиному інтерфейсі [11]. Вона спеціально розроблена для сучасних DevOps-середовищ і має понад 400 інтеграцій з різними інструментами та платформами, що робить її дуже гнучкою та масштабованою. Datadog дозволяє автоматично виявляти нові сервіси та контейнери, що спрощує моніторинг динамічних хмарних середовищ. Основною перевагою є простота використання та глибока інтеграція з хмарними сервісами. Однак система є комерційною, і її використання може бути дорогим для малих і середніх підприємств.



Рисунок 1.6 – Система моніторингу Datadog

Datadog використовує агентну архітектуру, де агенти встановлюються на різні пристрої та сервіси, збираючи дані про їх продуктивність. Ці дані передаються до хмарної платформи Datadog, де вони аналізуються і виводяться в інтуїтивному інтерфейсі [12].

Платформа пропонує моніторинг продуктивності серверів, контейнерів, мережевих пристроїв, додатків і баз даних. Крім того, Datadog надає можливості аналізу логів, АРМ (моніторинг продуктивності додатків), відстеження безпеки та аналітики. Всі ці функції доступні через єдиний веб-інтерфейс, що спрощує управління та моніторинг.

Переваги:

- Підтримка понад великої кількості інтеграцій із популярними хмарними платформами, що робить Datadog універсальним рішенням для сучасних хмарних інфраструктур.
- Платформа легка у налаштуванні та використанні, пропонує інтуїтивний інтерфейс та швидке виявлення нових сервісів і контейнерів.
- Працює в хмарі, тому не потребує додаткових локальних серверів для зберігання або обробки даних.

Недоліки: Продовження таблиці 1.1.

- Як комерційна платформа, Datadog може бути дорогим рішенням для малих і середніх підприємств, особливо з великими обсягами даних.
- Оскільки це хмарне рішення, його ефективність залежить від стабільності інтернет-з'єднання.
- Datadog широко використовується в компаніях, що працюють у хмарних середовищах, а також у DevOps-командах, які потребують ефективного моніторингу динамічних середовищ і контейнерів.

Таблиця 1.1 – Порівняння характеристик систем моніторингу

Характеристика	Zabbix	Nagios	Prometheus	SolarWinds	Datadog
Тип ліцензії	Відкритий код (Open Source)	Відкритий код (Open Source)	Відкритий код (Open Source)	Комерційна	Комерційна
Архітектура	Агентна та безагентна	Модульна, плагіни	Агентна, pull-підхід	Модульна	Агентна, хмарна
Основне призначення	Сервери, мережі, додатки,	Мережеві пристрої, бази даних	Контейнери, хмарні середовища	Мережеві пристрої, сервери, додатки	Хмарні та DevOps середовища, додатки

Продовження таблиці 1.1.

Основне призначення	Сервери, мережі, додатки, віртуальні середовища	Мережеві пристрої, сервери, бази даних	Контейнери, хмарні середовища	Мережеві пристрої, сервери, додатки	Хмарні та DevOps середовища, додатки
Підтримка контейнерів	Так	Можна через плагіни	Ідеально підходить	Обмежено	Так
Масштабованість	Висока через Zabbix Proху	Висока, за умови правильних налаштувань	Висока, особливо з Kubernetes	Висока (підходить для великих підприємств)	Висока (особливо для хмарних середовищ)
Гнучкість налаштувань	Дуже гнучка	Гнучка через плагіни	Висока гнучкість завдяки PromQL	Висока	Велика кількість інтеграцій
Інтерфейс	Веб-інтерфейс	Обмежений, потребує додаткових рішень	Мінімалістичний, інтеграції через Grafana	Професійний, багатофункціональний	Інтуїтивний, хмарний
Можливість інтеграції	API, веб-хуки	Плагіни та інтеграції	Експортери для різних систем	Інтеграція з іншими продуктами SolarWinds	Понад 400 інтеграцій
Автоматичне виявлення	Так	Обмежено через плагіни	Так	Так	Так
Сповіщення	Email, SMS, Slack, інші	Email, SMS	Alertmanager для сповіщень	Email, SMS, інші	Email, SMS, Slack, інші
Переваги	Масштабованість, інтеграції, гнучкість	Широкий вибір плагінів, велика спільнота	Підтримка хмарних та контейнерних середовищ	Комплексність, професійна підтримка	Простота у використанні, хмарне рішення
Недоліки	Вимоглива до ресурсів, складне налаштування	Складність візуалізації, потребує плагінів	Вимагає додаткових інструментів для алертів	Висока вартість, складне впровадження	Висока вартість, залежність від хмари
Сфера застосування	Великі підприємства, мережі, сервери	Великі підприємства, серверні ферми	DevOps, хмарні інфраструктури	Великі підприємства, урядові організації	Підприємства, що працюють в хмарі

Порівняння систем моніторингу показало, що вибір платформи значною мірою залежить від потреб організації, масштабу інфраструктури та бюджету. Zabbix є потужним рішенням з відкритим кодом, яке підходить для великих організацій, але складність його налаштування може бути викликом. Nagios — простіший у використанні для базового моніторингу, проте для розширення його функціональності потрібно докласти більше зусиль. Prometheus відмінно інтегрується з хмарними та контейнерними середовищами, що робить його ідеальним для DevOps-команд, але в традиційних інфраструктурах може бути менш ефективним. SolarWinds — це комерційне рішення з багатим функціоналом, орієнтоване на великі корпорації, однак його вартість і складність можуть бути перешкодою для малих компаній. Datadog, завдяки своїй хмарній архітектурі, забезпечує масштабованість і легкість у використанні для сучасних хмарних середовищ, проте висока вартість може бути недоліком для менших організацій. Загалом, Zabbix і Nagios пропонують бюджетні рішення з відкритим кодом, тоді як SolarWinds і Datadog орієнтовані на великі підприємства з відповідними ресурсами.

1.3 Структура та функціональні можливості Zabbix

Zabbix було обрано для моніторингу в кібербезпеці через його гнучкість, масштабованість та потужні можливості [13]. Як платформа з відкритим кодом, вона доступна без ліцензійних витрат, що робить її привабливою для організацій різних розмірів. Zabbix підтримує різноманітний моніторинг IT-інфраструктури, здатен виявляти аномалії, інтегрується з системами безпеки і має інтуїтивно зрозумілий веб-інтерфейс.

Архітектура Zabbix містить наступні основні компоненти [14]:

Zabbix Server, рис.1.7, центральний компонент, який збирає і обробляє всі дані про стан інфраструктури. Він виконує основні завдання зберігання, аналізу та відображення інформації.

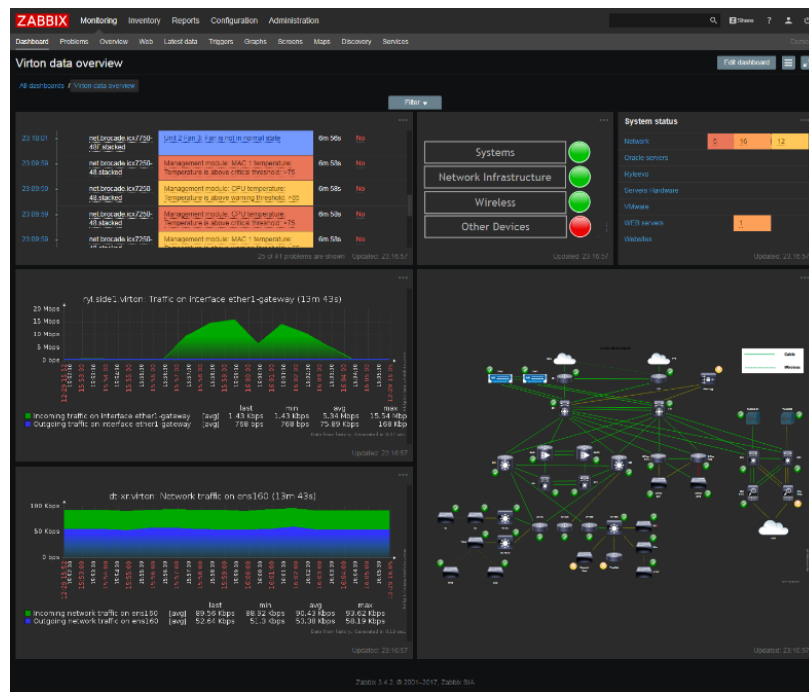


Рисунок 1.7 – Zabbix Server

Zabbix Agents, рис.1.8, агенти, які встановлюються на кінцеві пристрої та збирають дані про їхню продуктивність. Агенти можуть працювати як у активному, так і в пасивному режимі, відправляючи дані до Zabbix Server.

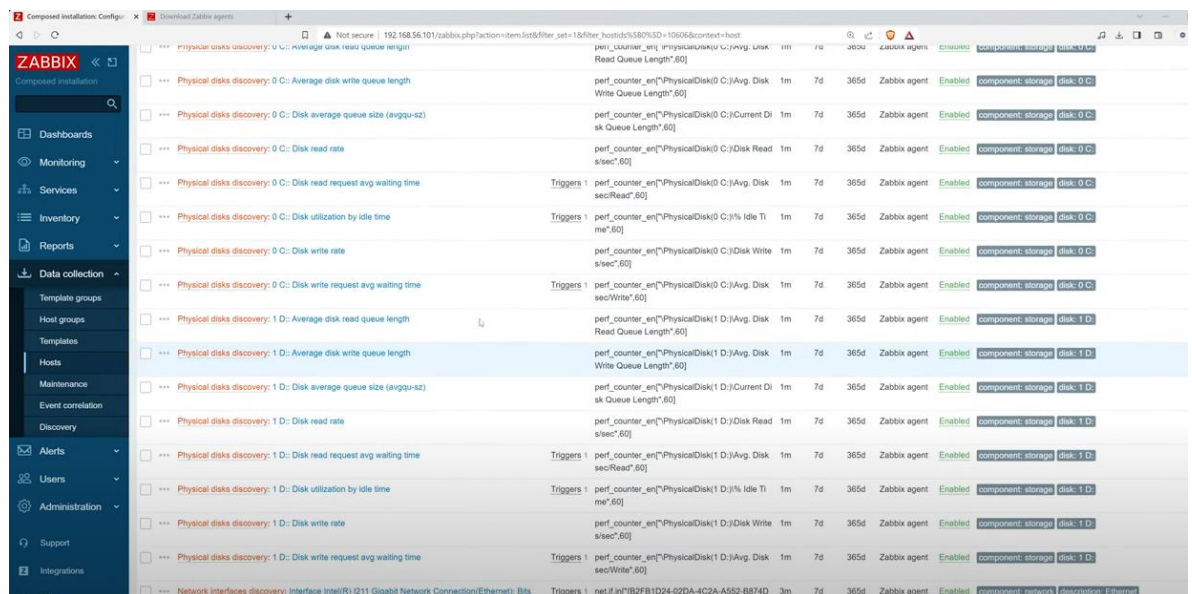


Рисунок 1.8 – Zabbix Agent

Zabbix Proxy, рис.1.9, допоміжний компонент, що використовується для масштабування системи. Проксі збирає дані в окремих сегментах мережі та передає їх на головний сервер, що дозволяє оптимізувати навантаження та забезпечувати моніторинг віддалених мереж.

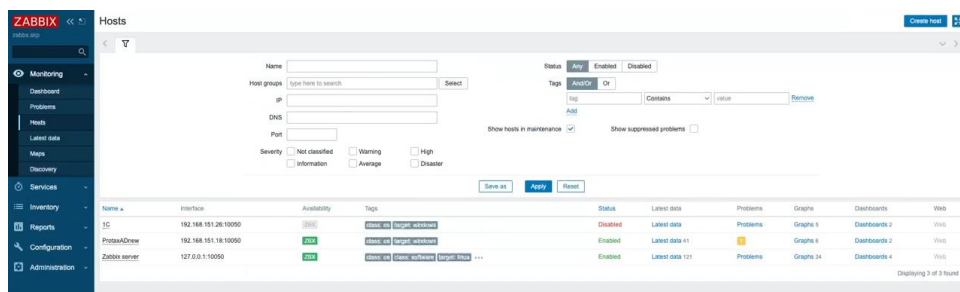


Рисунок 1.9 – Zabbix Proху

Zabbix Frontend, рис.1.10, веб-інтерфейс для взаємодії з системою. Через нього користувачі можуть переглядати зібрані дані, налаштовувати нові елементи моніторингу та керувати сповіщеннями.

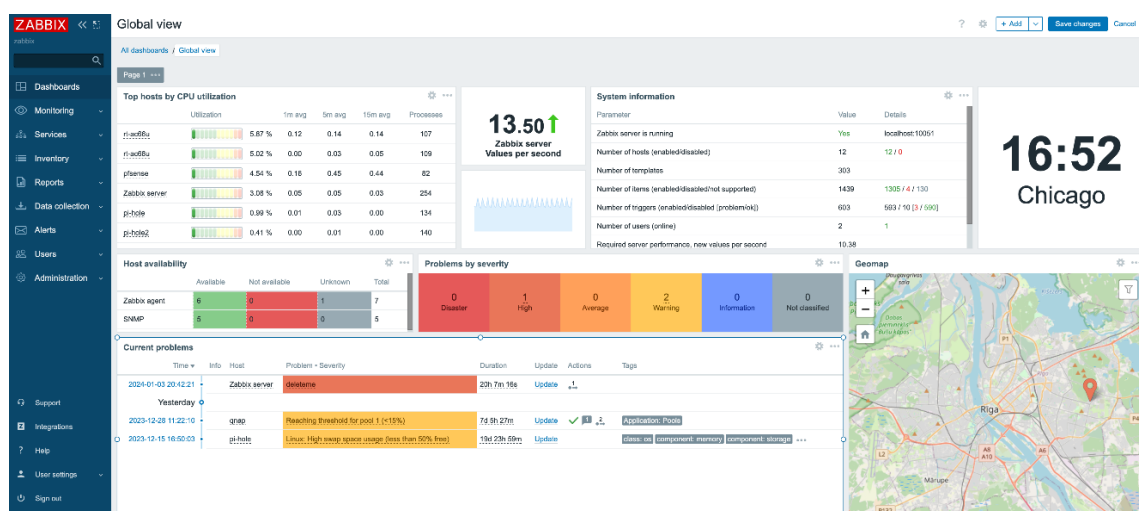


Рисунок 1.10 – Zabbix Frontend

Zabbix не лише моніторить інфраструктуру, але й відіграє важливу роль у забезпеченні безпеки своїх ІТ-інфраструктур [15].

1. Моніторинг систем безпеки.

Zabbix може інтегруватися з різними системами безпеки, такими як міжмережеві екрани (firewalls), системи виявлення та запобігання вторгненням (IDS/IPS), антивірусні програми та рішення для управління безпекою. Ця інтеграція дозволяє системі отримувати дані про стан цих компонентів в реальному часі, що забезпечує можливість швидко реагувати на проблеми, такі як збій в роботі або виявлення загроз. Наприклад, якщо IDS виявляє підозрілу активність, Zabbix може отримати цю інформацію і надіслати сповіщення відповідальним особам для подальшого аналізу.

2. Виявлення аномалій.

Однією з ключових функцій Zabbix є виявлення аномалій у трафіку та поведінці систем. Zabbix може аналізувати патерни використання мережі та виявляти незвичні активності, які можуть свідчити про кібератаки, наприклад, DDoS-атаки або спроби несанкціонованого доступу. Це дозволяє своєчасно вжити заходів для захисту ресурсів і запобігти можливим витокам даних.

3. Моніторинг журналів (логів).

Zabbix може збирати, зберігати та аналізувати журнали з різних систем, включаючи операційні системи, мережеві пристрої, бази даних та додатки [16]. Це забезпечує центральне місце для збору інформації, що дозволяє виявляти інциденти безпеки. Наприклад, шляхом аналізу логів можна виявити підозрілі спроби доступу або аномальні зміни у конфігурації систем. Zabbix підтримує інтеграцію з такими системами, як ELK (Elasticsearch, Logstash, Kibana), що дозволяє покращити аналіз даних.

4. Сповіщення про інциденти.

Zabbix має розвинену механіку сповіщень, яка дозволяє налаштовувати автоматичні повідомлення про критичні події безпеки. Користувачі можуть створювати тригери, які активуються при виявленні певних умов. Сповіщення можуть бути надіслані через електронну пошту, SMS або через інтеграцію з іншими системами, що дозволяє оперативно реагувати на інциденти.

5. Моніторинг оновлень та патчів.

Zabbix дозволяє відстежувати статус оновлень програмного забезпечення та патчів на серверах і мережевих пристроях. Це важливо для забезпечення безпеки систем, оскільки застаріле програмне забезпечення може містити вразливості, які зловмисники можуть експлуатувати. Zabbix може перевіряти наявність доступних оновлень, автоматично повідомляти адміністратора про необхідність їх встановлення та вести облік вже встановлених, що спрощує управління безпекою.

6. Контроль доступу.

Zabbix може моніторити спроби входу в системи та вести облік аутентифікаційних подій. Це дозволяє виявляти підозрілі дії, такі як повторні невдалі спроби входу, що можуть свідчити про спроби злому або фішинг. Система може також перевіряти відповідність політикам доступу та ідентифікувати несанкціоновані дії, що дозволяє покращити загальну безпеку організації.

7. Аудит та звітність.

Zabbix надає можливість створення звітів та аудитів безпеки, що допомагає в оцінці стану безпеки організації [17]. Звіти можуть містити інформацію про виявлені інциденти, вжиті заходи, а також рекомендації щодо покращення безпеки. Це дозволяє керівництву організації отримувати огляд ситуації з безпекою та приймати обґрунтовані рішення.

Висновок до першого розділу

У першому розділі дипломної роботи було розглянуто теоретичні аспекти систем моніторингу, їхнє призначення та основні функціональні можливості. Проведено аналіз існуючих рішень у цій сфері, що дало змогу виявити переваги й недоліки популярних платформ моніторингу, а також оцінити їхню ефективність у забезпеченні контролю за інфраструктурою корпоративних мереж.

Досліджені системи моніторингу забезпечують безпековий функціонал, зокрема виявлення аномалій, інтеграцію з IDS/IPS, моніторинг журналів подій, сповіщення про інциденти та контроль доступу, що дозволяє ефективно реагувати на кіберзагрози і підтримувати цілісність IT-інфраструктури. Особливу увагу приділено системі Zabbix: її структурі, архітектурі та основним компонентам. Дослідження дозволило визначити, що Zabbix є багатофункціональним рішенням, яке забезпечує моніторинг пристроїв, серверів і додатків із високою гнучкістю у налаштуванні, масштабуванні та можливістю інтеграції із зовнішніми системами для забезпечення безпечного функціонування мережі.

РОЗДІЛ 2 ПРОЄКТУВАННЯ СИСТЕМИ МОНІТОРИНГУ МЕРЕЖІ З ВИКОРИСТАННЯМ ZABBIX

2.1. Розробка топології комп'ютерної мережі

Вибір топології комп'ютерної мережі є складним завданням, що потребує всебічного аналізу різноманітних факторів. Однією з основних проблем є баланс між продуктивністю мережі та фінансовими витратами на її реалізацію. Вибір топології має вплив на швидкість передачі даних, надійність зв'язку та обсяг витрат на обладнання і обслуговування.

Крім того, необхідно враховувати потреби організації в масштабованості та гнучкості. Якщо організація має намір швидко розширюватись, обрана топологія повинна дозволяти легке додавання нових компонентів без значних затримок у роботі мережі.

Також важливим аспектом є відмовостійкість системи. У разі виникнення збоїв топологія повинна забезпечувати можливість швидкого відновлення зв'язку і мінімізації втрат даних.

Вибір ієрархічної моделі мережі (3-рівнева ієрархічна модель) обумовлений її численними перевагами, що забезпечують оптимальну роботу мережі в умовах зростаючих вимог сучасного бізнес-середовища [18].

По-перше, ця структура дозволяє організувати мережу за трьома чітко визначеними рівнями: рівнем ядра, розподілу та доступу. Така структура сприяє більш ефективній маршрутизації даних, що підвищує швидкість передачі інформації та зменшує затримки, що є критично важливим для організацій, що залежать від швидкого обміну даними.

По-друге, така топологія забезпечує модульність та масштабованість. Це означає, що організації можуть легко додавати нові компоненти чи сегменти мережі без потреби в значній переробці вже існуючої інфраструктури. У разі розширення організації чи зміни вимог до мережі така гнучкість стає важливою перевагою.

І крім того, трирівнева структура забезпечує високу відмовостійкість системи. Завдяки структурованому підходу до організації мережі, у разі виникнення несправностей можна швидко виявити та усунути проблеми без значних витрат часу та ресурсів. Це зменшує ризики втрати даних і забезпечує безперервність роботи мережі. Ієрархічна модель мережі побудована на основі трьох рівнів, кожен з яких виконує певні функції і забезпечує ефективну роботу мережі. Ця структура дозволяє організувати мережу з мінімальними затримками, зручним управлінням та високою відмовостійкістю.

1. Рівень ядра (Core Layer). Рівень ядра є центральною частиною мережі, що відповідає за високошвидкісне пересилання даних між основними компонентами мережі. Його основне завдання — забезпечити швидку передачу трафіку з мінімальними затримками. На цьому рівні використовуються потужні комутатори або маршрутизатори, здатні обробляти великі обсяги трафіку.

2. Рівень агрегації (Distribution Layer). Цей рівень об'єднує нижній рівень доступу з ядром і відповідає за маршрутизацію трафіку, політики безпеки, фільтрацію та агрегацію даних від підлеглих мереж. Рівень агрегації виконує функції обробки та розподілу трафіку між рівнем доступу і ядром. Він також відповідає за впровадження таких функцій, як контроль доступу до мережі, якість обслуговування (QoS) і забезпечення безпеки. Важливою перевагою цього рівня є те, що він забезпечує ефективне керування навантаженням і відмовостійкість мережі.

3. Рівень доступу (Access Layer). Рівень доступу є точкою взаємодії кінцевих користувачів з мережею. Це найнижчий рівень у ієрархічній структурі, де відбувається підключення пристроїв, таких як комп'ютери, принтери, точки доступу Wi-Fi тощо. Основні функції цього рівня включають управління доступом до мережі, комутацію на рівні локальних підключень і підтримку різноманітних типів підключень (дротових або бездротових).

На основі згаданої 3-рівневої ієрархічної моделі мережі було побудовано наступну структурну схему проєкту мережі, рис.2.1.

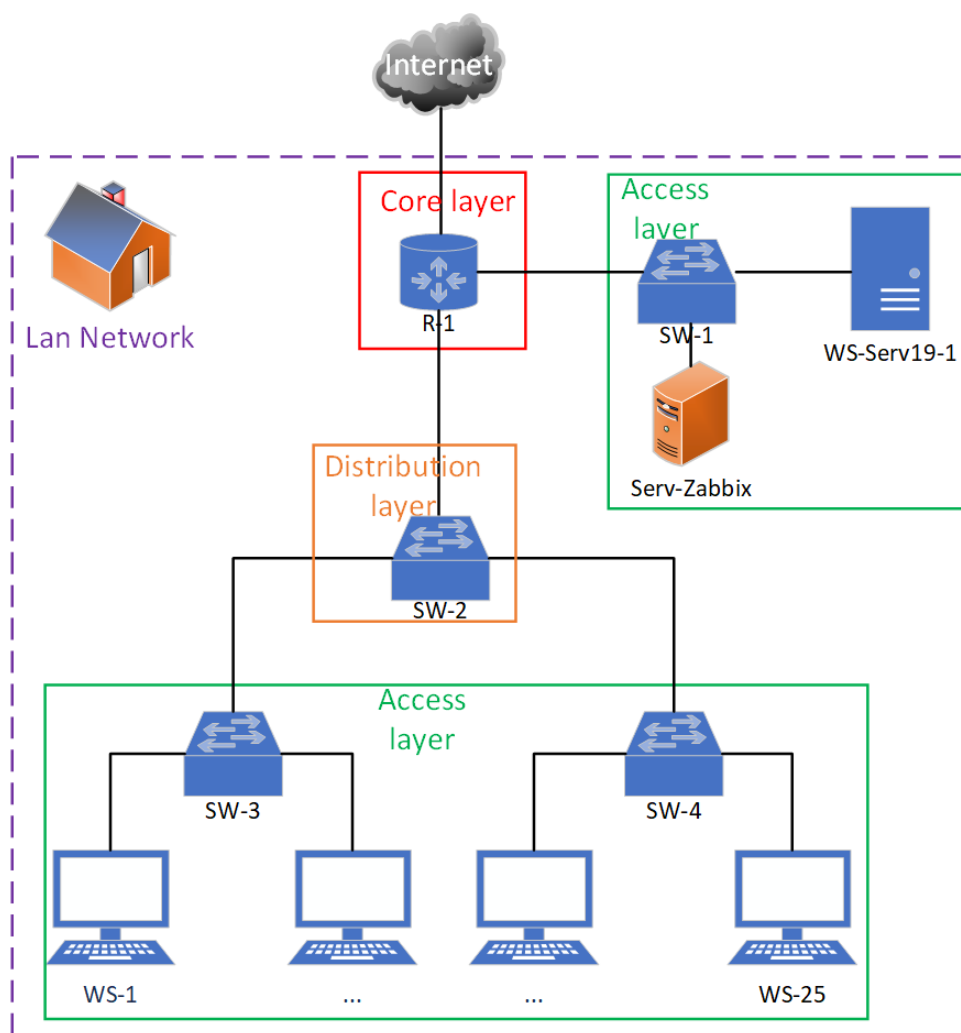


Рисунок 2.1 – Структурна схема мережі

Рівень ядра має один високошвидкісний маршрутизатор R-1, який має вихід до глобальної мережі Інтернет.

Рівень розподілу представлений комутатором SW-2, який з'єднаний з рівнем ядра, тобто з R-1.

Рівень доступу представлений комутаторами SW-1, SW-3 та SW-4 та кінцевим обладнанням.

Комутатор SW-1 підключений до маршрутизатора R-1 через виту пару Gigabit Ethernet. В свою чергу до даного комутатора підключено серверне обладнання, а саме Windows Server 2019 (WS-Serv19-1) та комп'ютер зі встановленим сервером моніторингу Zabbix (Serv-Zabbix). Підключення даних компонентів не випадкове, оскільки подібне розташування ближче до центру мережі покращує відмовостійкість серверного обладнання та пришвидшує

швидкість передачі шляхом прямого підключення до магістрального маршрутизатора R-1.

Комутатори SW-3 та SW-4 підключені до комутатора рівня розподілу SW-2 через виту пару Gigabit Ethernet. В свою чергу до SW-3 та SW-4 під'єднані кінцеві пристрої користувачів, що включає комп'ютери та ноутбуки загальною кількістю 25 одиниць.

Загалом в мережі використано 1 маршрутизатор, 4 комутатори, 25 кінцевих пристроїв користувачів та 2 локальні підмережі.

Наступним важливим етапом в розробці є вибір схеми IP-адресації в мережі з урахування економії адресного простору. В даному випадку для функціонування проєкту необхідна 1 мережа для каналу між маршрутизатором та Інтернетом та 2 локальні підмережі:

R-1 – Internet: 192.168.122.0/24

R-2 – SW-2 (LAN A): 192.168.1.0/24

R-1 – SW-1 (LAN Man): 192.168.99.0/24

Детальний розрахунок адрес для локальних підмереж наведений у таблицях 2.1 – 2.2.

Таблиця 2.1 – Адресація підмережі LAN A

Призначення адрес	IP-адреса
Адреса мережі	192.168.1.0
Маска мережі	255.255.255.0(/24)
Мінімальна адреса	192.168.1.1
Максимальна адреса	192.168.1.254
Шлюз за замовчуванням	192.168.1.254
Призначені комутаторам	192.168.1.252 – 192.168.1.253
Діапазон адрес майбутніх ПК	192.168.1.1 – 192.168.1.25
Діапазон вільних адрес	192.168.1.26 – 192.168.1.251

Таблиця 2.2 – Адресація підмережі LAN Man

Призначення адрес	IP-адреса
Адреса мережі	192.168.99.0
Маска мережі	255.255.255.240(/28)
Мінімальна адреса	192.168.99.1
Максимальна адреса	192.168.99.14
Шлюз за замовчуванням	192.168.99.14
Призначені комутаторам	192.168.99.13
Діапазон адрес майбутніх ПК	192.168.99.1 – 192.168.99.2
Діапазон вільних адрес	192.168.99.3 – 192.168.99.12

Фізична топологія мережі наведена нижче на рис.2.2.

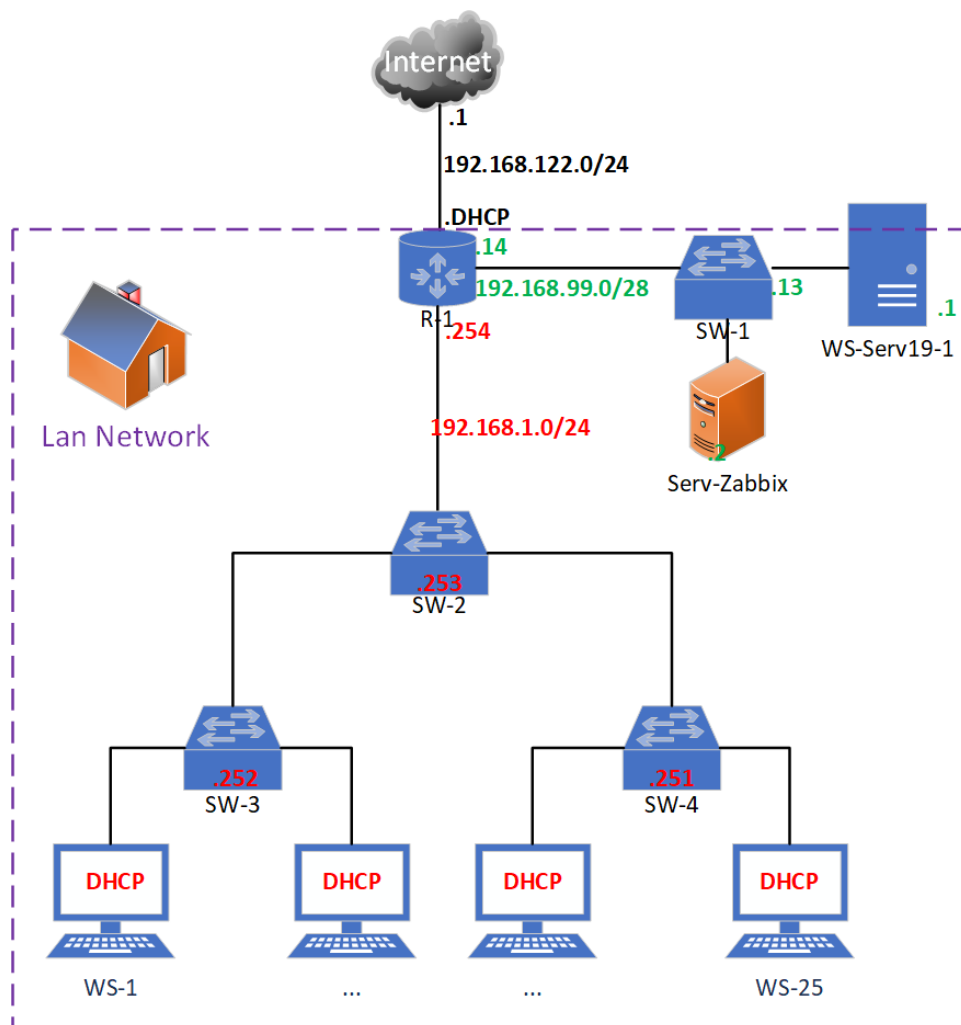


Рисунок 2.2 – Фізична топологія мережі

2.2 Організація роботи Zabbix

В проєкті мережі моніторинг направлений переважно на відстеження працездатності кінцевих пристроїв, тобто станція з встановленою системою ОС Windows. Тому найбільш доцільним варіантом для збору та передачі даних є використання агенту Zabbix через ряд його суттєвих переваг.

- **Всеосяжний моніторинг.** Збирає інформацію про системні ресурси (процесор, пам'ять, диски, трафік мережі), стан служб Windows, список процесів, системні дані та помилки, що допомагає швидко виявляти проблеми.
- **Мінімальне навантаження.** Працює з незначним використанням ресурсів, що знижує навантаження на сервери.
- **Масштабованість.** Дозволяє запускати власноруч створені команди та скрипти для розширення моніторингу.
- **Моніторинг подій та логів Windows.** Підтримка WMI та Windows Performance Counters для збору метрик з різних служб і додатків (SQL Server, IIS тощо) та аналізу подій, що допомагає у виявленні критичних ситуацій.
- **Легка автоматизація.** Можна розгортати й оновлювати агенти у великих масштабах за допомогою Active Directory, використовуючи MSI пакети для безшумної установки.
- **Моніторинг у реальному часі.** Збирає дані про стан системи, такі як завантаження процесора, використання пам'яті, стан дисків тощо, у реальному часі.

Сам же сервер Zabbix може бути встановлений на різних операційних системах включаючи Windows, Ubuntu, Debian, CentOS, Red Hat, Oracle Linux і може бути розгорнутий локально, в контейнері і навіть хмарному середовищі.

Однак все ж рекомендується виконувати його встановлення на станціях з ОС Linux, оскільки ця платформа забезпечує найкращу сумісність і продуктивність. До того ж Zabbix оптимізований під Linux, що дозволяє йому обробляти великі обсяги даних і підтримувати високі навантаження, особливо

при моніторингу великої кількості хостів забезпечуючи стабільність і ефективність роботи.

Місце розгортання залежить від багатьох факторів та індивідуальних побажань, однак в даній мережі логічніше виконати його встановлення на фізичному комп'ютері.

По-перше, це забезпечує повний контроль над ресурсами та конфігурацією системи. Оскільки Zabbix дуже чутливий до параметрів продуктивності, фізичний комп'ютер дозволяє виділити необхідні ресурси, уникнути віртуалізаційних обмежень та знизити ризик затримок у передачі даних.

По-друге, локальне розгортання зменшує залежність від зовнішніх мережевих підключень, що підвищує надійність моніторингу. У критичних ситуаціях це забезпечує безперебійний доступ до сервера, навіть якщо виникнуть проблеми з мережею або Інтернетом.

На рис.2.3 наведена топологія роботи Zabbix з використанням агентів на кінцевих пристроях.

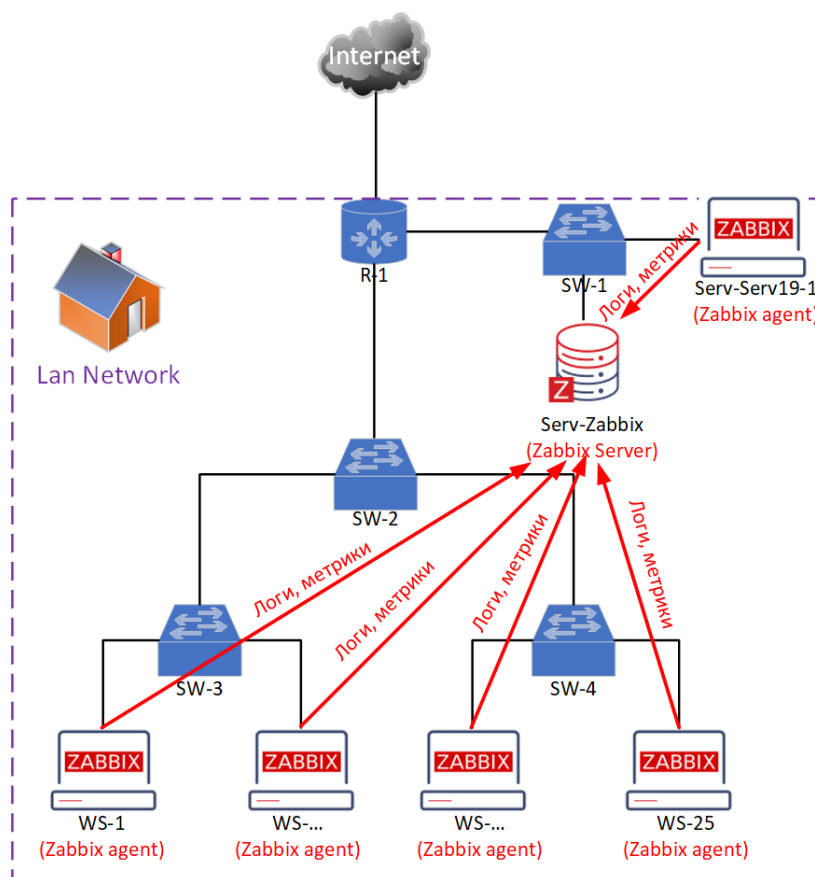


Рисунок 2.3 – Структура Zabbix в проєкті мережі

2.3 Вибір обладнання

Важливим етапом є вибір обладнання. Так при виборі маршрутизатора слід звертати увагу на продуктивність, кількість і типи інтерфейсів, масштабованість, функції безпеки, управління трафіком, надійність, вартість, підтримку сучасних протоколів та можливість віддаленого керування.

Маршрутизатор Cisco ISR 4331, рис.2.4, є оптимальним вибором для проекту мережі завдяки своїй продуктивності, надійності та гнучкості в умовах сучасної корпоративної інфраструктури [19]. Він здатний обробляти великі обсяги трафіку, забезпечуючи швидкість до 100 Мбіт/с з можливістю розширення до 300 Мбіт/с, що дозволяє адаптувати його до змін у мережі, зокрема при збільшенні навантаження. Обсяг пам'яті DRAM у 4GB з можливістю розширення до 16GB дозволяє справлятися із завданнями з високою швидкістю. Його модульна архітектура дозволяє легко додавати нові функції та інтерфейси без значних змін в основному обладнанні, що є зручним для організацій, які планують розширення або модернізацію своєї мережі.



Рисунок 2.4 – Маршрутизатор Cisco ISR 4331

При виборі комутатора важливо враховувати пропускну здатність, кількість і типи портів, можливість масштабування, підтримку функцій безпеки, управління трафіком, надійність, енергоспоживання, вартість володіння та підтримку сучасних мережевих стандартів.

Комутатор Комутатор Cisco Catalyst WS-C2960X-24PS-L є оптимальним вибором завдяки своїй енергоефективності, високій продуктивності та

можливостям масштабування [20]. Він підтримує швидкість Gigabit Ethernet на всіх портах, що забезпечує високу пропускну здатність для сучасних корпоративних мереж. Завдяки функції стекування можна об'єднувати до восьми комутаторів для спрощення управління та підвищення гнучкості. Cisco 2960-X також має вбудовані функції безпеки, зокрема контроль доступу та захист від загроз, що робить його надійним рішенням для мереж з підвищеними вимогами до безпеки. Дані комутатори лінійки Cisco Catalyst WS-C2960X забезпечені 24 або 48 портами Gigabit Ethernet, підтримують технології Power over Ethernet/Power over Ethernet Plus (PoE/PoE+) і чотири канали Gigabit Ethernet Small Form-Factor Pluggable (SFP) або два канали 10 Gigabit Ethernet Small Form-Factor (SFP+). Комутатори працюють на базі двох ядерних процесорів 600МГц. Технологія FlexStack-Plus дозволяє зібрати стек з 8 комутаторів та забезпечує пропускну здатність 80 Гбіт/сек. Для забезпечення візуалізації та контролю трафіку різних програм у мережі, Cisco додали функцію NetFlow Lite, яка дозволяє інженерам моніторити та керувати потоками даних на всіх сегментах мережі.



Рисунок 2.5 – Комутатор Cisco Catalyst WS-C2960X-24PS-L

Для встановлення Zabbix на комп'ютер необхідно вибрати систему з достатньою продуктивністю для обробки запитів у режимі реального часу. Рекомендується використовувати комп'ютер із багатоядерним процесором, наприклад Intel Xeon або AMD Ryzen, який забезпечить швидке реагування системи. Оперативна пам'ять повинна бути не менше 8 ГБ, а для великих мереж — 16 ГБ або більше, щоб ефективно обробляти запити та зберігати історичні дані. Важливим є наявність SSD-диска обсягом від 200 ГБ для швидкої роботи

бази даних, а також гігабітної мережевої карти для отримання даних із мережевих пристроїв.

Вибір ОС Ubuntu обґрунтований її стабільністю, широкою підтримкою від спільноти та сумісністю з Zabbix. Ubuntu забезпечує легкість налаштування, підтримку необхідних пакетів і низькі системні вимоги, що дозволяє максимально ефективно використовувати ресурси для моніторингу та управління мережею.

Windows Server 2019 є потужною та універсальною операційною системою для управління серверними ресурсами, підтримує безліч серверних ролей і служб. Вибір даної ОС дозволяє використовувати всі її функції для побудови надійної серверної інфраструктури, з можливістю масштабування і забезпечення високого рівня безпеки та керованості.

Для встановлення Windows Server 2019 необхідно вибрати комп'ютер із достатніми ресурсами для забезпечення стабільної роботи серверної інфраструктури. Оптимальним є використання багатоядерного процесора, такого як Intel Xeon, що забезпечує високу продуктивність для обробки кількох завдань одночасно. Рекомендується щонайменше 16 ГБ оперативної пам'яті, особливо якщо сервер використовуватиметься для віртуалізації або великих баз даних, оскільки це дозволить системі обробляти значні навантаження без затримок. Для зберігання даних найкраще використовувати SSD-диск обсягом від 200 ГБ для швидкого доступу до файлів і забезпечення стабільної роботи баз даних та серверних додатків. Важливо також мати гігабітну мережеву карту для забезпечення швидкого та надійного обміну даними з іншими серверами або клієнтськими машинами

Висновки до другого розділу

В першому підрозділі було виконано розробку топології комп'ютерної мережі з використанням трирівневої ієрархічної моделі. Виконано розподіл адресації між пристроями з використанням наступних підмереж: LAN А

(192.168.1.0/24) для користувачів і LAN Man (192.168.99.0/28) для серверів та адміністративного обладнання.

Для моніторингу кінцевих пристроїв мережі буде використовуватись Zabbix із встановленими агентами на кінцевих пристроях. Дана система забезпечить моніторинг всіх ключових ресурсів, служб і логів з мінімальним навантаженням на пристрої і все це в режимі реального часу.

За результатами аналізу вимог мережі, на роль маршрутизатора обрано Cisco ISR 4331 зважаючи на його продуктивність, масштабованість і підтримку сучасних протоколів. Комутатори моделі Cisco Catalyst WS-C2960X-24PS-L забезпечать достатню пропускну здатність для кінцевих пристроїв. Для сервера Zabbix рекомендовано комп'ютер з багатоядерним процесором, не менше 16 ГБ ОЗП, SSD на 200 ГБ та ОС Ubuntu для стабільної роботи. Для комп'ютера на ОС Windows Server 2019 передбачено потужний процесор Intel Xeon, 16 ГБ ОЗП, SSD та гігабітна мережева карта, що зрештою забезпечить високопродуктивну роботу серверної інфраструктури.

РОЗДІЛ 3 ПОБУДОВА, НАЛАГОДЖЕННЯ ТА ТЕСТУВАННЯ РОБОТИ МЕРЕЖІ

3.1 Базове налаштування мережі

Подальше налагодження роботи мережі буде виконуватись з використанням мережевого емулятора GNS3. Дану програму часто використовують для емуляції мережевих проєктів через його зручність, функціональність і гнучкість. GNS3 дозволяє створювати складні мережі, включаючи реальні мережеві пристрої (Mikrotik, Cisco, Juniper) та образи ОС реального обладнання (Windows, Linux тощо). GNS3 також підтримує інтеграцію з різними віртуальними машинами, що дозволяє емуляцію робочих станцій і серверів в мережі. Інтерфейс програми зручний і підходить як для навчальних цілей, так і для тестування складних мережевих конфігурацій у реальних умовах. Схема проєкту мережі в емуляторі наведена на рис.3.1.

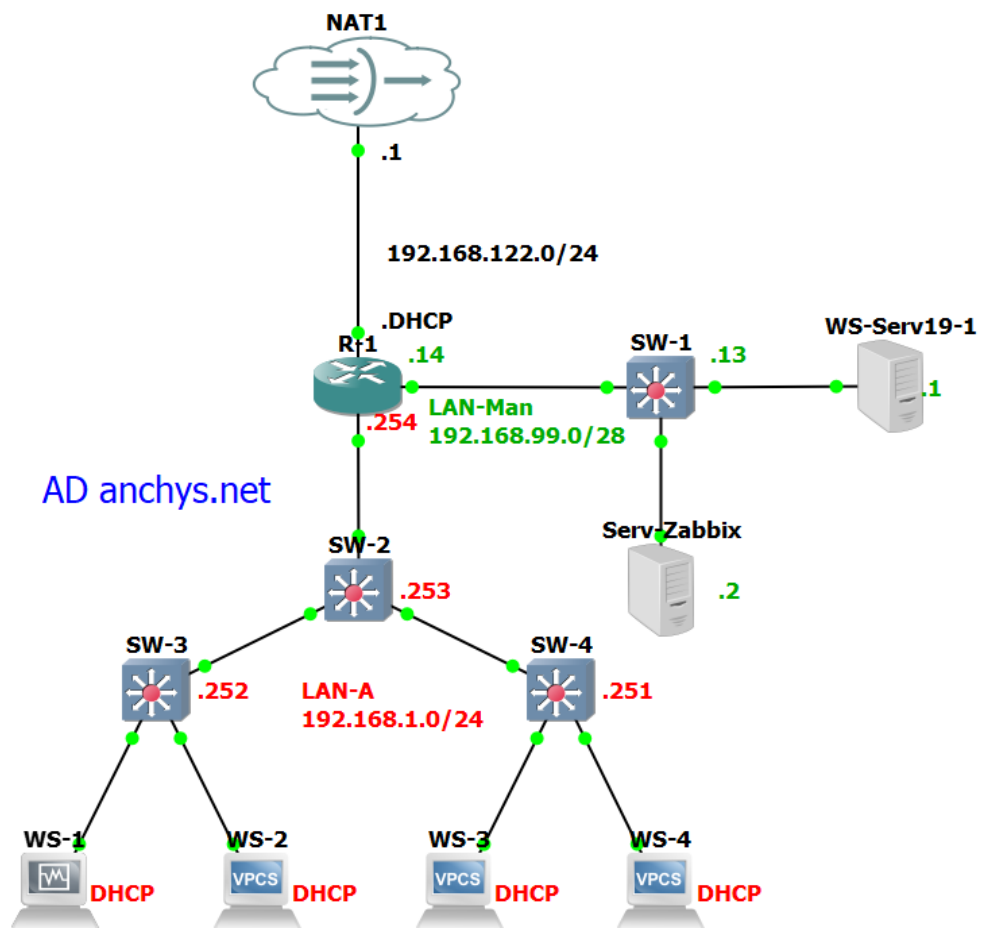


Рисунок 3.1 – Проєкт в мережі GNS3

Тож розпочнемо налаштування проєкту з базового налагодження роботи мережевих пристроїв. Розпочинаємо з налаштувань маршрутизатор R-1. Інтерфейсам маршрутизатора, а саме fa0/0, fa1/0 і fa1/1 спершу задається опис (description). Відповідно fa0/0 призначений для підключення до Інтернету, а fa1/0 та fa1/1 – до комутаторів локальної мережі. Далі інтерфейсам призначаються IP-адреси. fa0/0 отримує IP-адресу динамічно через DHCP (ip address dhcp) і позначається як зовнішній інтерфейс для NAT (ip nat outside), fa1/0 отримує статичну IP-адресу і позначається як внутрішній для NAT (ip nat inside), а fa1/1 аналогічний fa1/0, тільки з іншою локальною IP-адресою. Далі маршрутизатор налаштовується для роботи з доменами: задається доменне ім'я (ip domain-name anchys.net), активується функція перетворення доменів у IP-адреси (ip domain-lookup), а також вмикається робота DNS-сервера (ip dns server). Для обробки доменних імен на пристрій додаються адреси DNS-серверів, тобто адресу WS-Serv19-1 (ip name-server 192.168.99.1) і публічного Google (ip name-server 8.8.8.8). Для роботи NAT створюється список доступу (access-list 1 permit 192.168.0.0 0.0.255.255), який визначає діапазон локальних адрес, що підлягають трансляції через застосовування NAT, а точніше PAT з використанням зовнішнього інтерфейсу fa0/0 (ip nat inside source list 1 interface fastethernet 0/0 overload), рис.3.2.

```
R-1(config)#interface fa0/0
R-1(config-if)#description link-to-internet
R-1(config-if)#ip address dhcp
R-1(config-if)#ip nat outside
R-1(config-if)#no shutdown
R-1(config-if)#exit
R-1(config)#interface fa1/0
R-1(config-if)#description link-to-sw-1
R-1(config-if)#ip address 192.168.99.14 255.255.255.240
R-1(config-if)#ip nat inside
R-1(config-if)#no shutdown
R-1(config-if)#exit
R-1(config)#interface fa1/1
R-1(config-if)#description link-to-lan
R-1(config-if)#ip address 192.168.1.254 255.255.255.0
R-1(config-if)#ip nat inside
R-1(config-if)#no shutdown
R-1(config-if)#exit
R-1(config)#ip domain-name anchys.net
R-1(config)#ip domain-lookup
R-1(config)#ip dns server
R-1(config)#ip name-server 192.168.99.1
R-1(config)#ip name-server 8.8.8.8
R-1(config)#access-list 1 permit 192.168.0.0 0.0.255.255
R-1(config)#ip nat inside source list 1 interface fastethernet 0/0 overload
```

Рисунок 3.2 – Налаштування маршрутизатора R-1

Далі перейдемо до комутаторів. Для прикладу розглянемо SW-2, однак інші пристрої налаштовуються схожим чином. Спочатку інтерфейсу VLAN 1, який використовується як основний інтерфейс, задається IP-адреса (ip address), що дозволяє отримувати доступ до комутатора в локальній мережі за цією адресою. Далі задається шлюз за замовчуванням (default-gateway), де зазначається адреса маршрутизатора R-1, тобто 192.168.1.254. Налаштований шлюз за замовчуванням дозволить комутатору передавати і отримувати трафік за межами локальної мережі. Потім налаштовується робота DNS подібно, як це було виконано на маршрутизаторі раніше. Команда ip name-server вказує локальний DNS-сервер для перетворення доменних імен у IP-адреси, в даному випадку це адреси 192.168.1.254 (маршрутизатора R-1) та 192.168.99.1 (сервер WS-Serv19-1). Команда ip domain name anchys.net задає доменне ім'я для комутатора, а ip domain lookup активує функцію перетворення доменних імен. Ці виконані налаштування забезпечать мережеву доступність комутатора, його можливість звертатися до зовнішніх ресурсів і використовувати доменні імена для зв'язку, рис.3.3.

```
SW-2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
SW-2(config)#int vlan1
SW-2(config-if)#ip address 192.168.1.253 255.255.255.0
SW-2(config-if)#no shut
SW-2(config-if)#exit
SW-2(config)#ip default-gateway 192.168.1.254
SW-2(config)#ip name-server 192.168.1.254
SW-2(config)#ip domain name anchys.net
SW-2(config)#ip domain lookup
```

Рисунок 3.3 – Налаштування комутатора SW-2

На всіх мережевих пристроях доцільно налаштувати базову аутентифікацію віддалене підключення через SSH. Команда enable secret enable-pass задає захищений пароль для доступу до привілейованого режиму налаштувань. За допомогою команди username admin privilege 15 secret admin-pass створено обліковий запис admin із максимальним рівнем привілеїв (15) і паролем admin-pass. Команда transport input ssh дозволяє підключення лише через SSH для термінальних ліній 0 - 4, а login local задає використання

локальних облікових записів для автентифікації. Аналогічна команда `login local` застосована і до консольної лінії (`line cons 0`). Для забезпечення роботи SSH згенеровано RSA-ключі командою `crypto key generate rsa modulus 1024`. SSH переведено на другу версію командою `ip ssh version 2`, а параметр `ip ssh time-out 30` задає максимальний час очікування для встановлення з'єднання (30 секунд). Точно такі самі налагодження виконуються на всіх інших комутаторах мережі, рис.3.4.

```
R-1(config)#enable secret enable-pass
R-1(config)#username admin privilege 15 secret admin-pass
R-1(config)#line vty 0 4
R-1(config-line)#transport input ssh
R-1(config-line)#login local
R-1(config-line)#exit
R-1(config)#line cons 0
R-1(config-line)#login local
R-1(config-line)#exit
R-1(config)#crypto key generate rsa modulus 1024
% You already have RSA keys defined named R-1.anchys.net.
% They will be replaced.

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]
R-1(config)#ip ssh version 2
R-1(config)#ip ssh time-out 30
```

Рисунок 3.4 – Налагодження автентифікації та SSH на маршрутизаторі R-1

Далі переходимо до налагодження Active Directory та інших сервісів на станції WS-Serv19-1. Перш за все призначаємо статичну адресу згідно виконаного розподілу адресації пристроїв, рис.3.5.

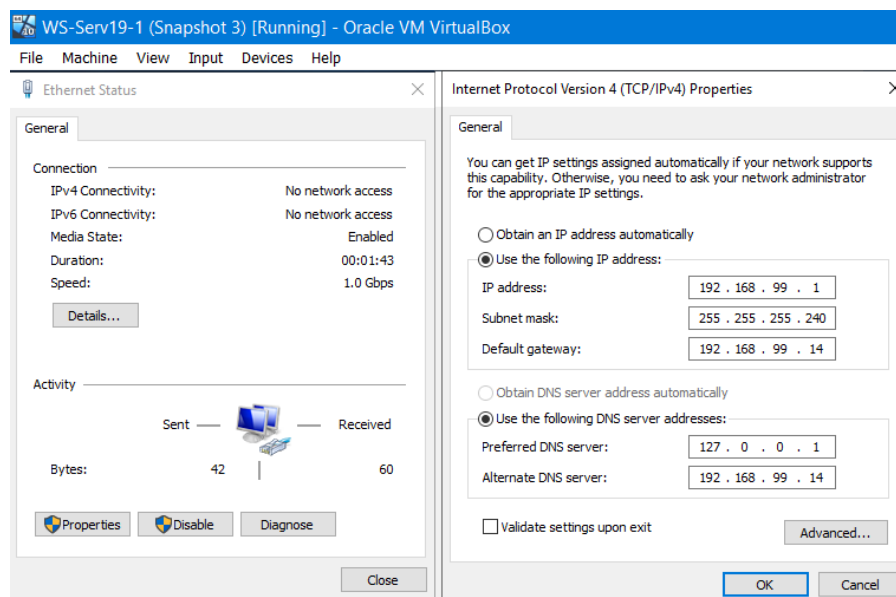


Рисунок 3.5 – Налаштування статичної адреси WS-Serv19-1

За цим виконуємо встановлення всіх необхідних ролей та компонентів та створюємо домен, рис.3.6 – 3.7.

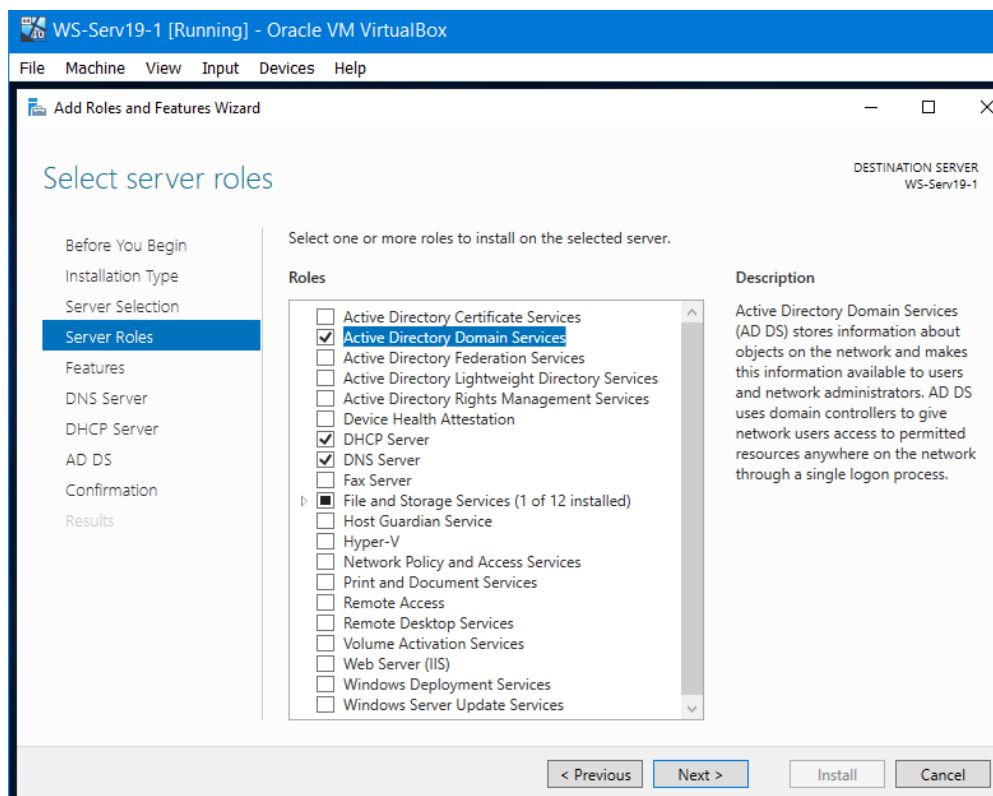


Рисунок 3.6 – Додання необхідних ролей та компонентів

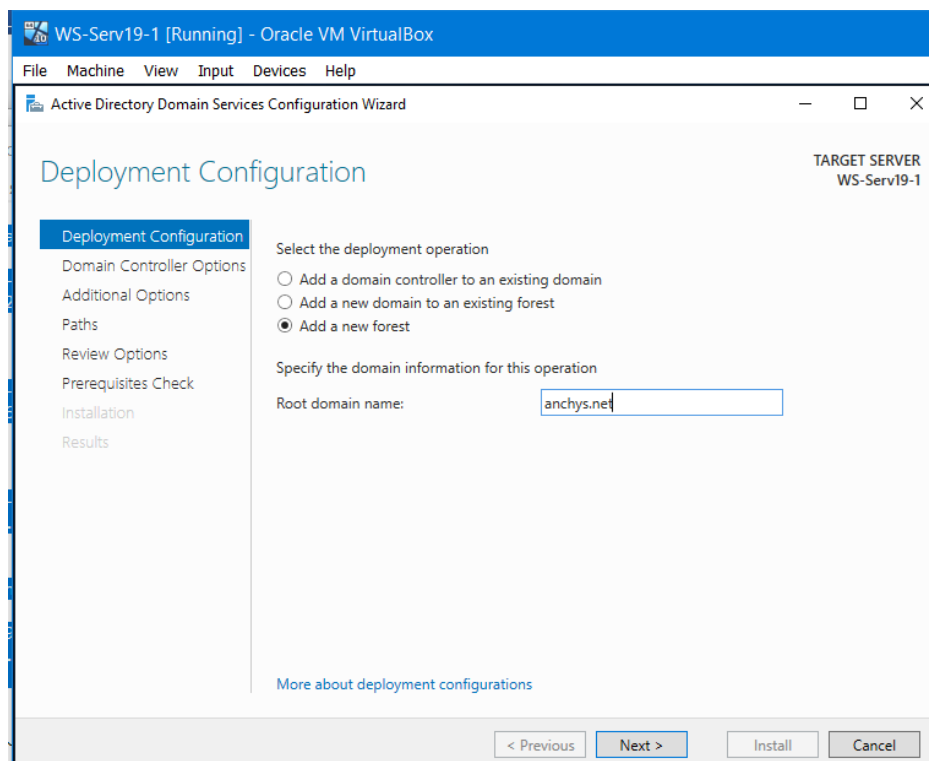


Рисунок 3.7 – Створення домену anchys.net

Після цього налаштовується робота ДНСП-сервера, який буде динамічно видавати адреси для кінцевих пристроїв локальної мережі, рис.3.8.

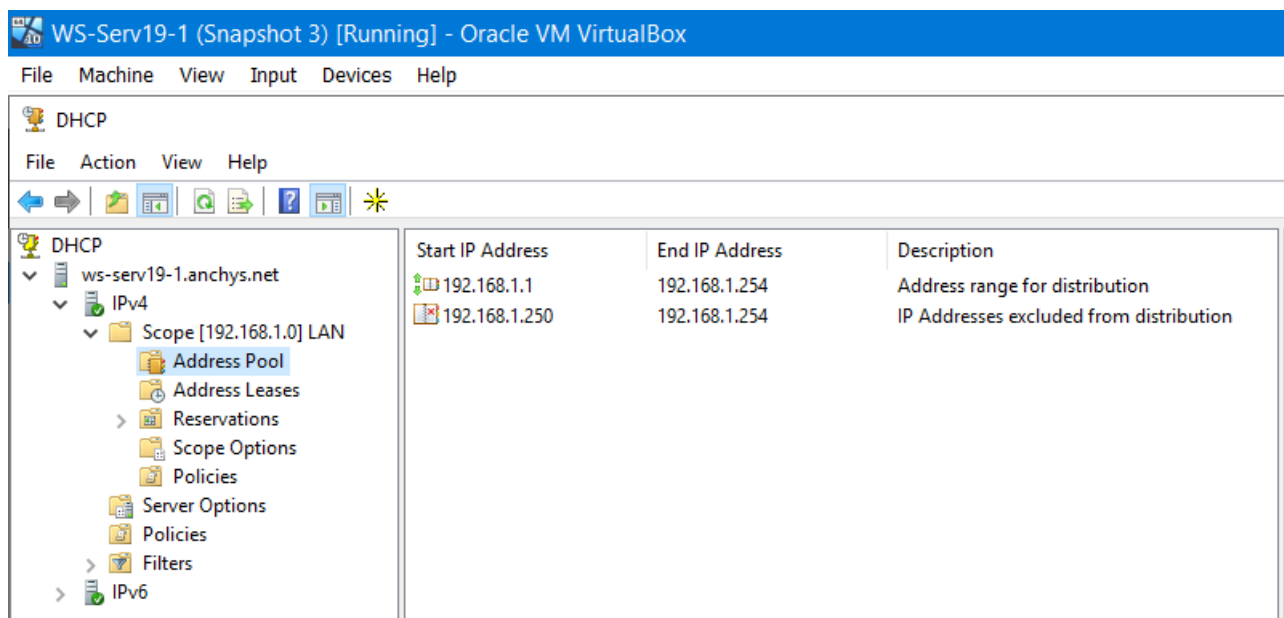


Рисунок 3.8 – Налаштування DHCP-сервера

Оскільки сервер DHCP знаходиться в іншій підмережі (192.168.99.0/28) аніж кінцеві пристрої (192.168.1.0/24), то клієнти не зможуть безпосередньо отримувати відповіді від нього через широкомовні запити. В такому випадку маршрутизатор R-1 необхідно додатково налаштувати як DHCP-ретранслятор. Для цього на інтерфейсі R-1, який виходить до локальної мережі (fa1/1) виконується команда `ip helper-address` з адресою DHCP-сервера (192.168.99.1), рис.3.8. Дана команда означає, що всі широкомовні запити DHCP, що надходять через цей інтерфейс (fa1/1) від кінцевих пристроїв, будуть пересилатись до DHCP-сервера, тобто до станції WS-Serv19-1 і тим самим пристрої користувачів зможуть отримувати динамічні адреси.

```
R-1(config)#int fa1/1
R-1(config-if)#ip helper-address 192.168.99.1
```

Рисунок 3.9 – Налаштування пересилання DHCP-запитів до сервера

Для налагодження DNS-сервера необхідно виконати додавання відповідних доменних імен пристроїв з їх реальним значенням IP-адрес, рис.3.10.

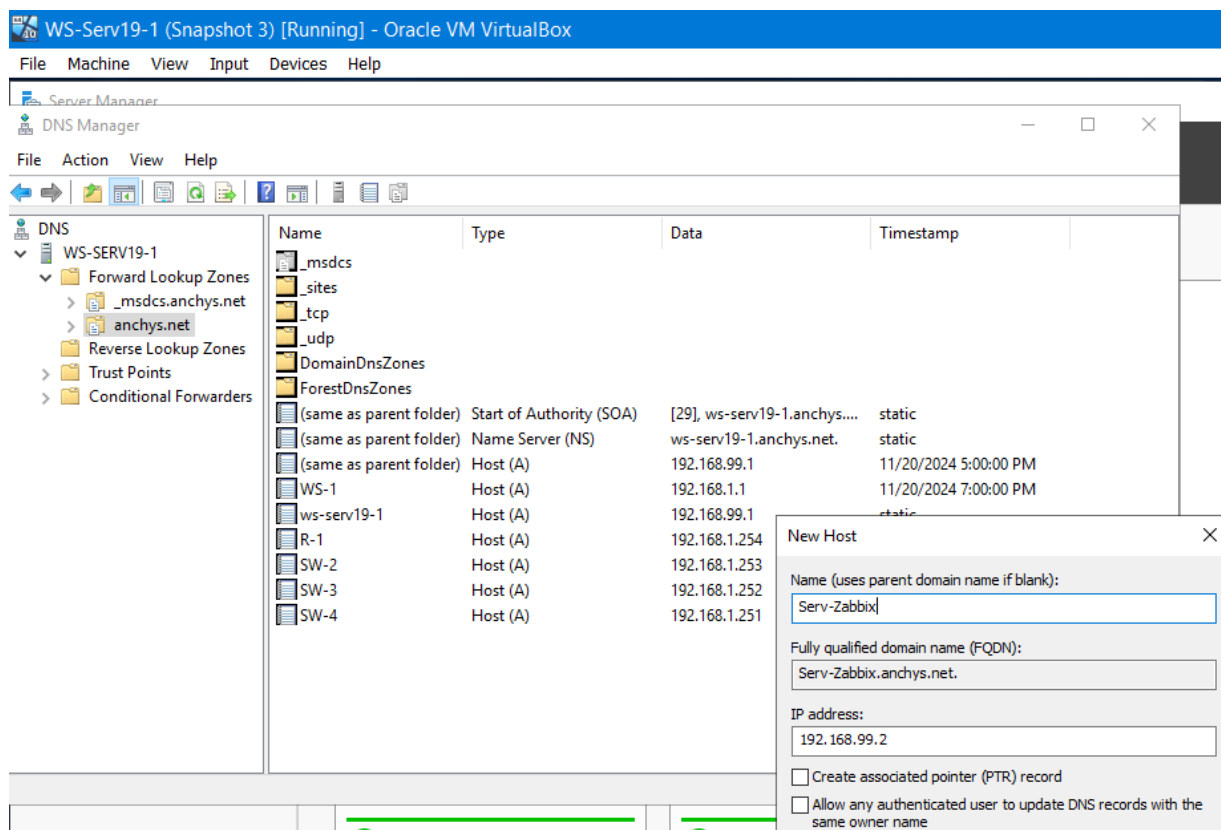


Рисунок 3.10 – Додання DNS записів

Залишається створити облікові записи користувачів, які будуть використовуватися для входу в систему на кінцевих робочих станціях, рис.3.11.

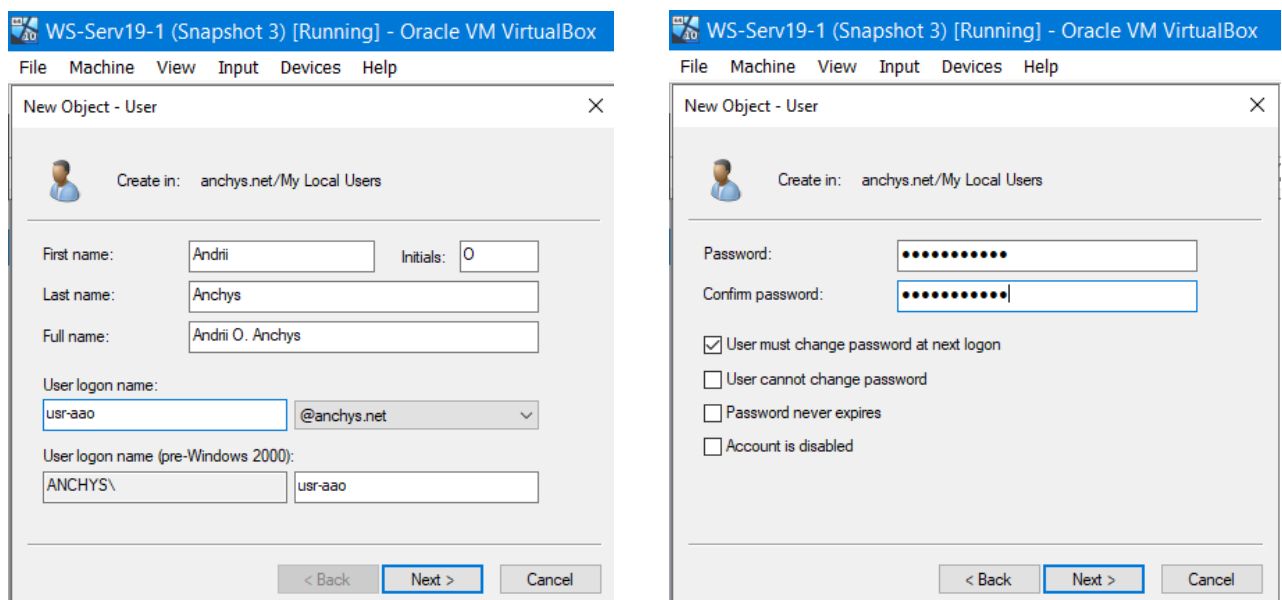


Рисунок 3.11– Створення облікового запису користувача

Перейдемо до налаштування кінцевих робочих станцій. Для прикладу розглянемо WS-1. В параметрах інтерфейсу вказуємо динамічне отримання адрес, рис.3.12.

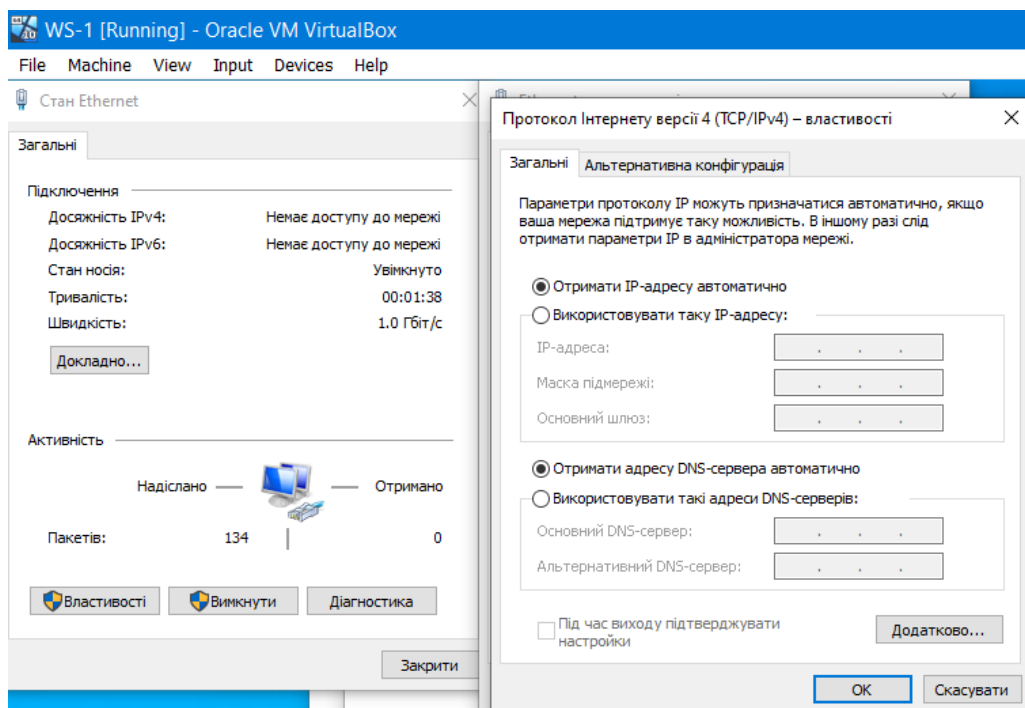


Рисунок 3.12 – Налаштування параметрів адресації на робочій станції
І додаємо робочу станції до домену anchys.net, рис.3.13.

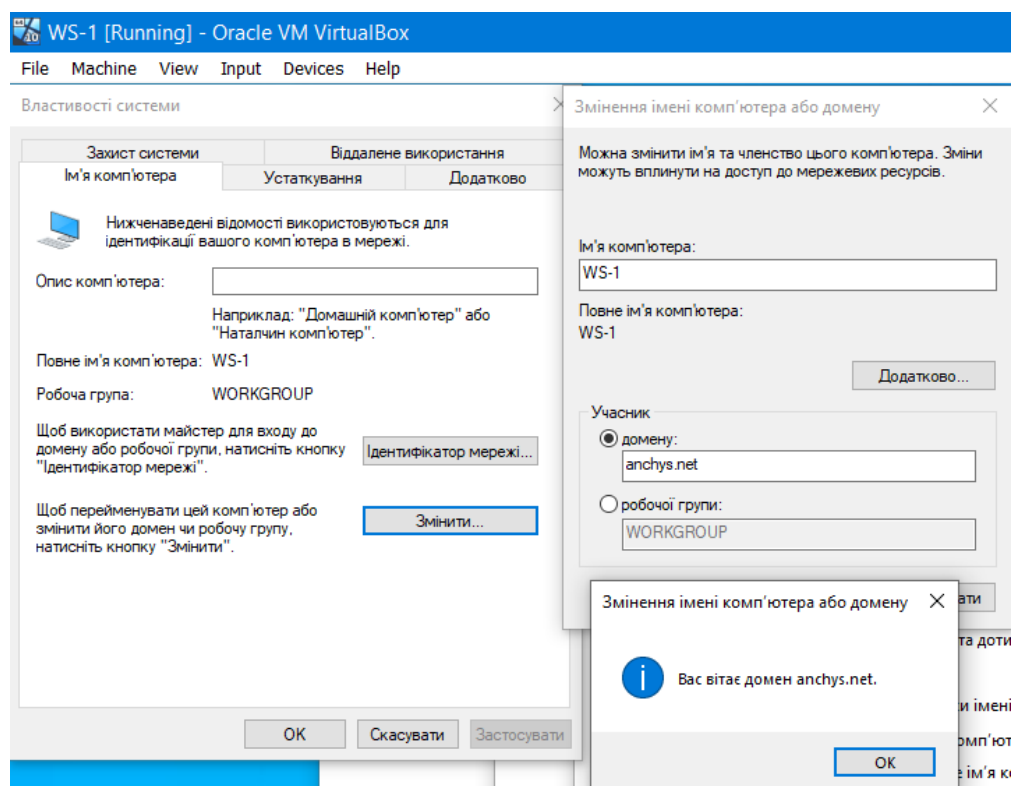


Рисунок 3.13 – Приєднання комп'ютера до домену anchys.net

Після автоматичного перезавантаження комп'ютера виконаємо вхід в систему з використання одного з облікових записів користувачів створених в Active Directory, рис.3.14.

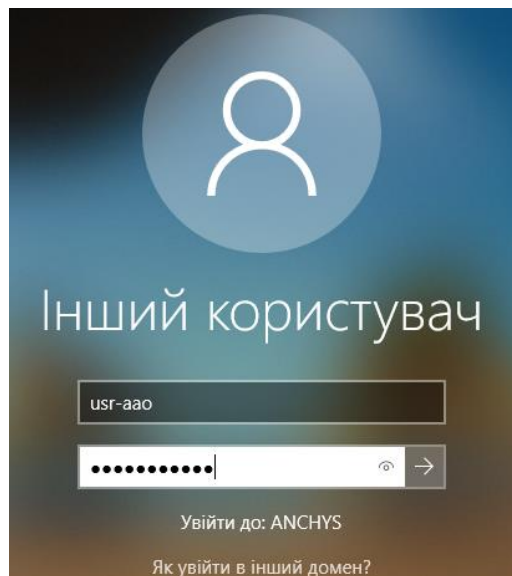


Рисунок 3.14 – Вхід в систему з використанням облікового запису користувача

І відразу звідси перевіriamo доступність всіх вузлів мережі виконавши пінг за їх доменними іменами, рис.3.15.

```
WS-1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
C:\Windows\system32\cmd.exe
C:\Users\usr-aa0>ping ws-serv19-1

Pinging WS-Serv19-1.anchys.net [192.168.99.1] with 32 bytes of data:
Reply from 192.168.99.1: bytes=32 time=20ms TTL=127
Reply from 192.168.99.1: bytes=32 time=18ms TTL=127
Reply from 192.168.99.1: bytes=32 time=17ms TTL=127
Reply from 192.168.99.1: bytes=32 time=14ms TTL=127

Ping statistics for 192.168.99.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 14ms, Maximum = 20ms, Average = 17ms

C:\Users\usr-aa0>ping ws-3

Pinging ws-3.anchys.net [192.168.1.3] with 32 bytes of data:
Reply from 192.168.1.3: bytes=32 time=3ms TTL=64
Reply from 192.168.1.3: bytes=32 time=1ms TTL=64
Reply from 192.168.1.3: bytes=32 time=1ms TTL=64
Reply from 192.168.1.3: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 3ms, Average = 1ms
```

Рисунок 3.15 – Перевірка зв'язку між пристроями мережі

Також перевіriamo можливість віддаленого підключення через SSH до мережевих пристроїв та аутентифікацію виконавши підключення з використанням PuTTY на WS-Serv19-1, рис.3.16.

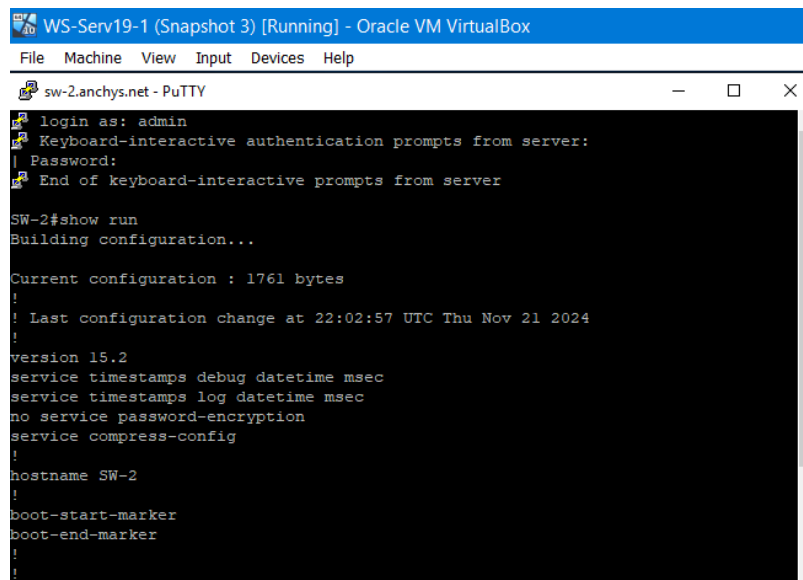


Рисунок 3.16 – Підключення через SSH до комутатора SW-2

3.2 Встановлення системи моніторингу Zabbix

Перш за все здійснюємо налаштування статичної адреси на станції Serv-Zabbix згідно виконаного розподілу, рис.3.17.

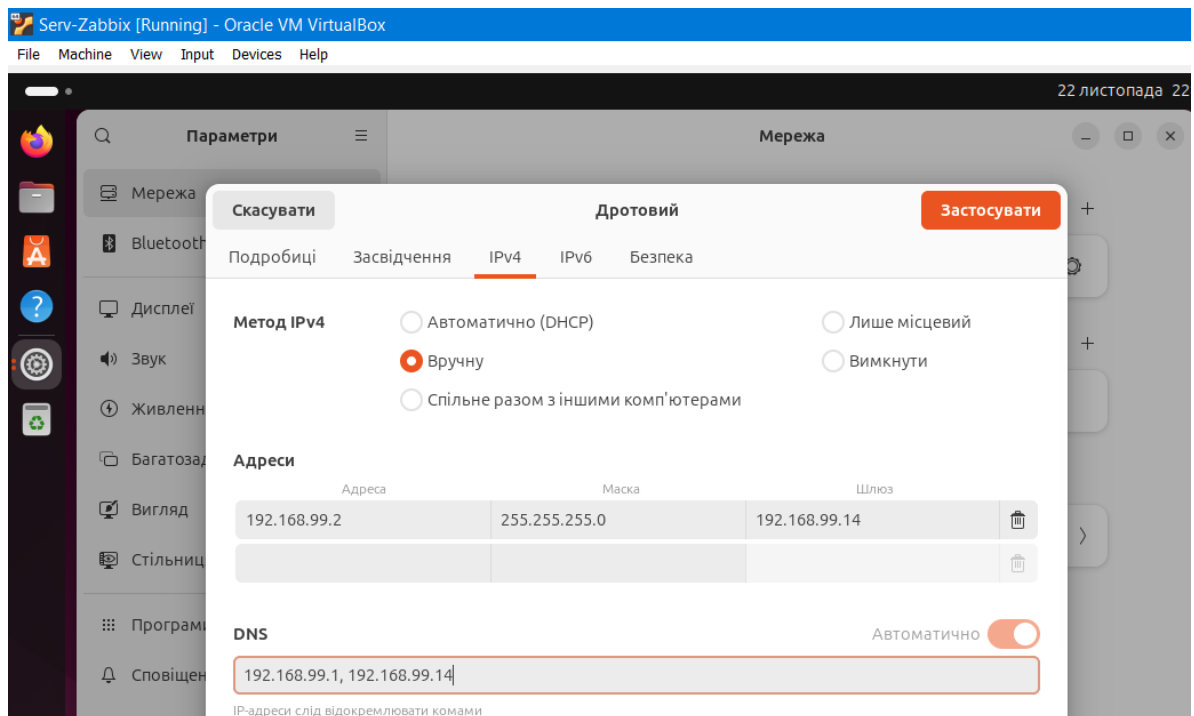


Рисунок 3.17 – Задання статичних параметрів адресації

Далі приступаємо до встановлення системи моніторингу Zabbix. Для цього переходимо на офіційний сайт Zabbix в розділ завантаження і зазначаємо необхідні параметри такі, як версія самого Zabbix, операційна система та її

версія, необхідні компоненти до завантаження, базу даних та веб-сервер, рис.3.18.

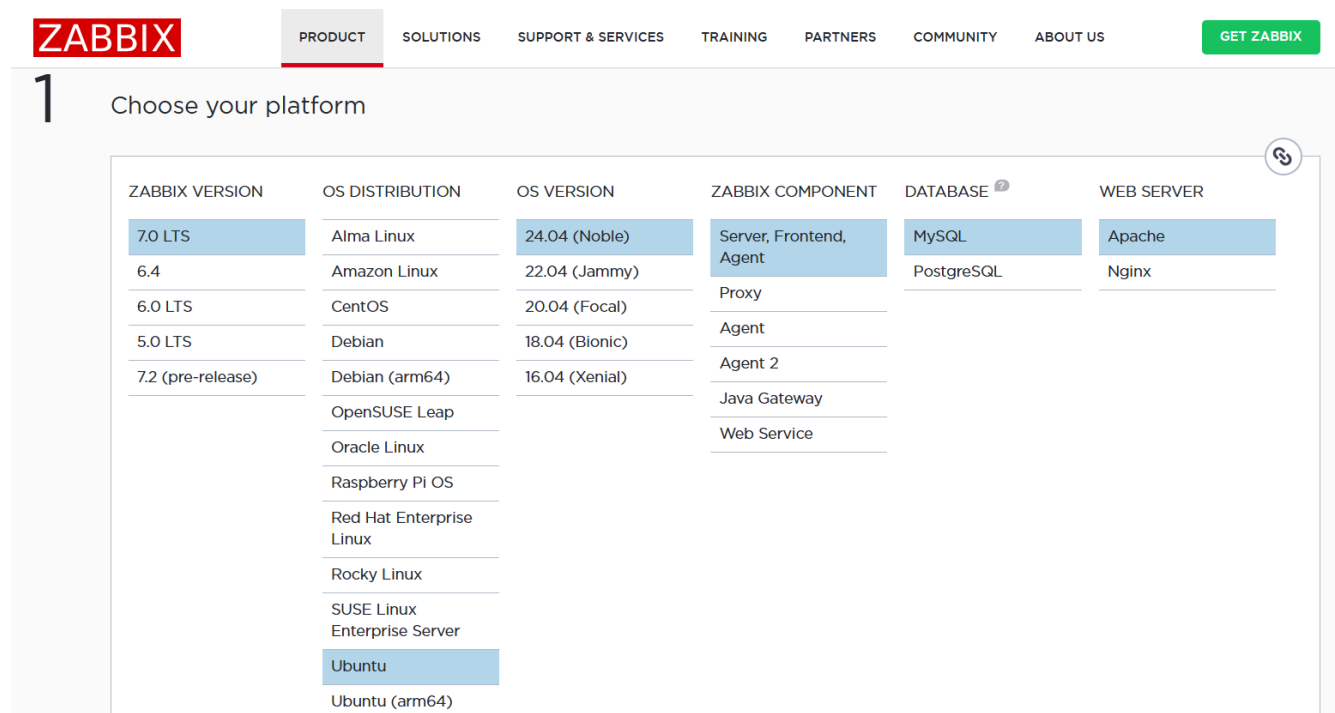


Рисунок 3.18 – Вибір параметрів встановлення Zabbix

І після цього просто виконуємо всі команди, що були згенеровані на згаданій сторінці завантаження з офіційного сайту Zabbix, рис.3.19 – 3.25.

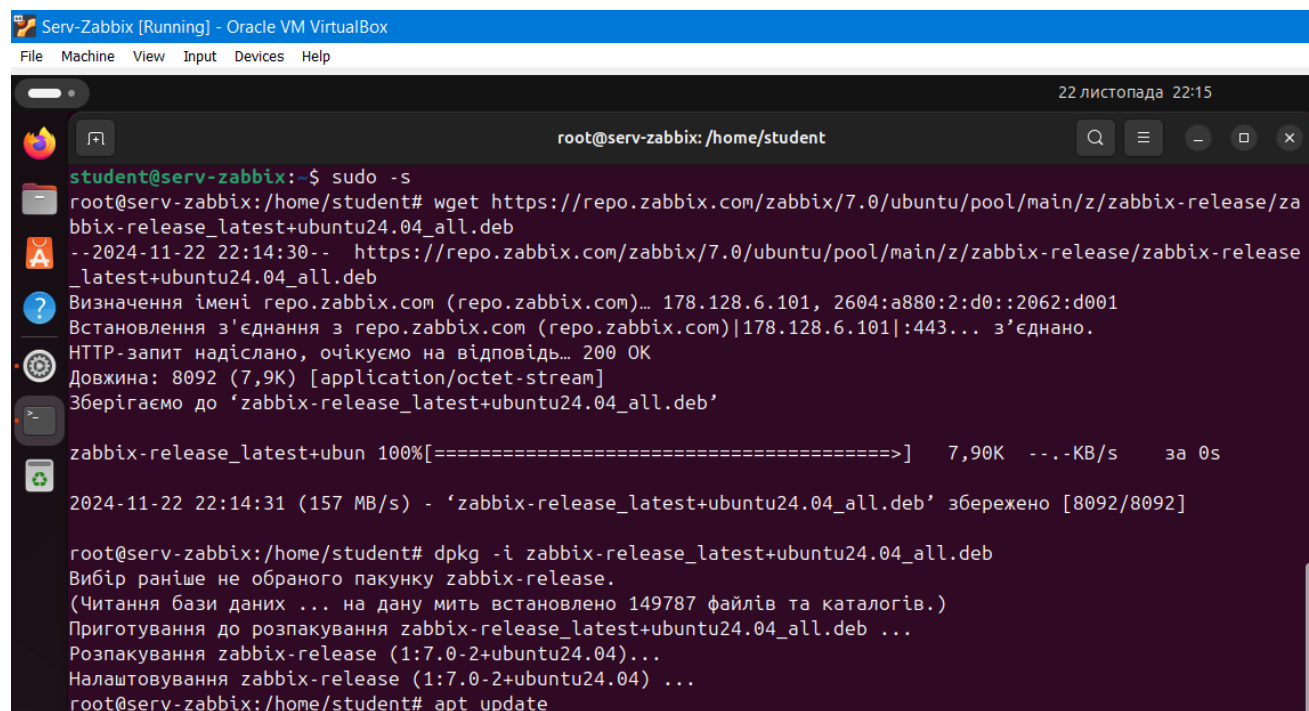


Рисунок 3.19 – Встановлення репозиторію Zabbix

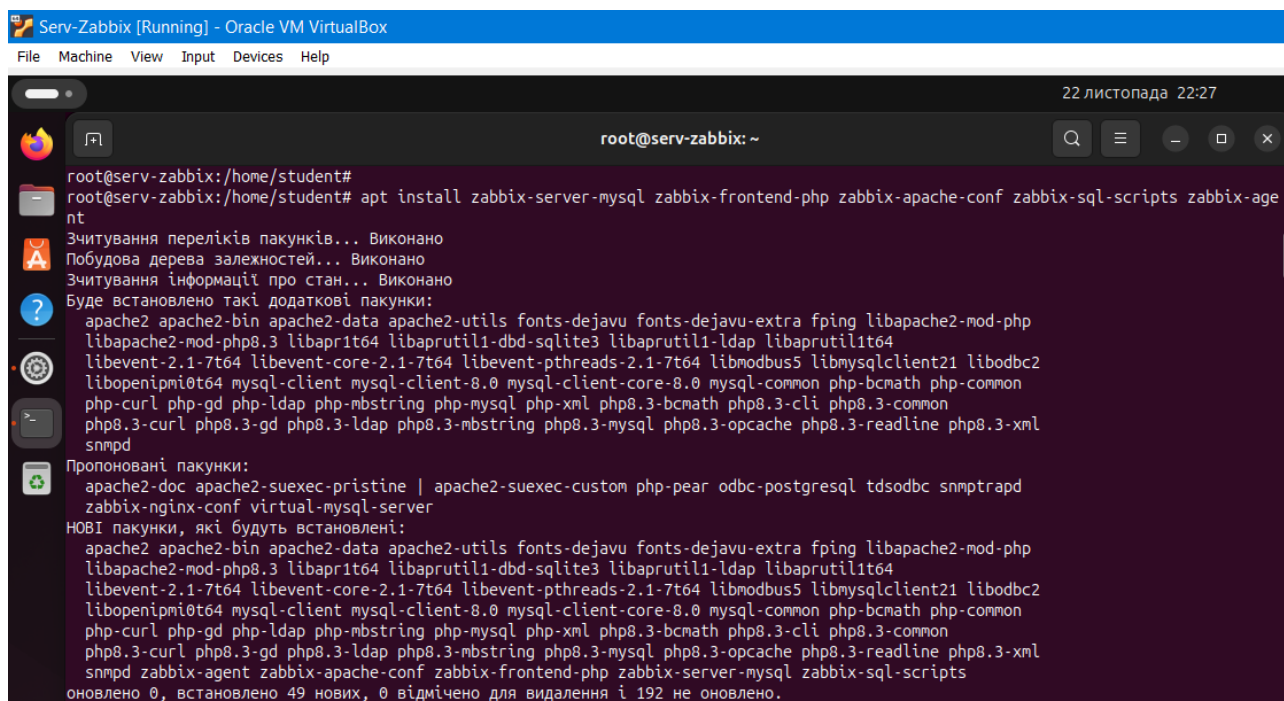


Рисунок 3.20 – Встановлення компонентів Zabbix

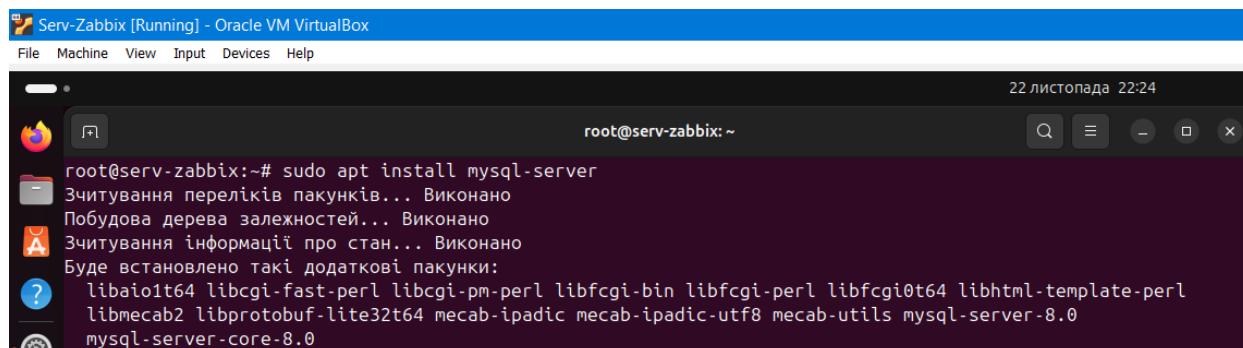


Рисунок 3.21 – Встановлення бази даних mysql

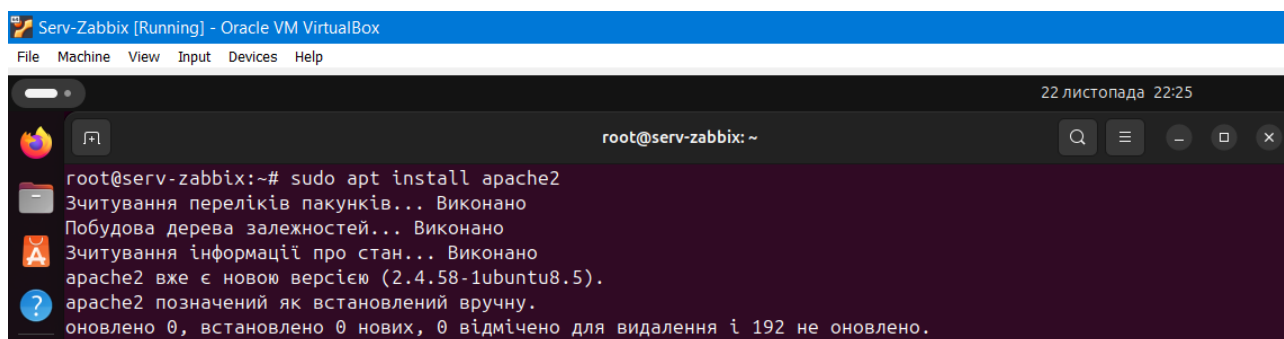
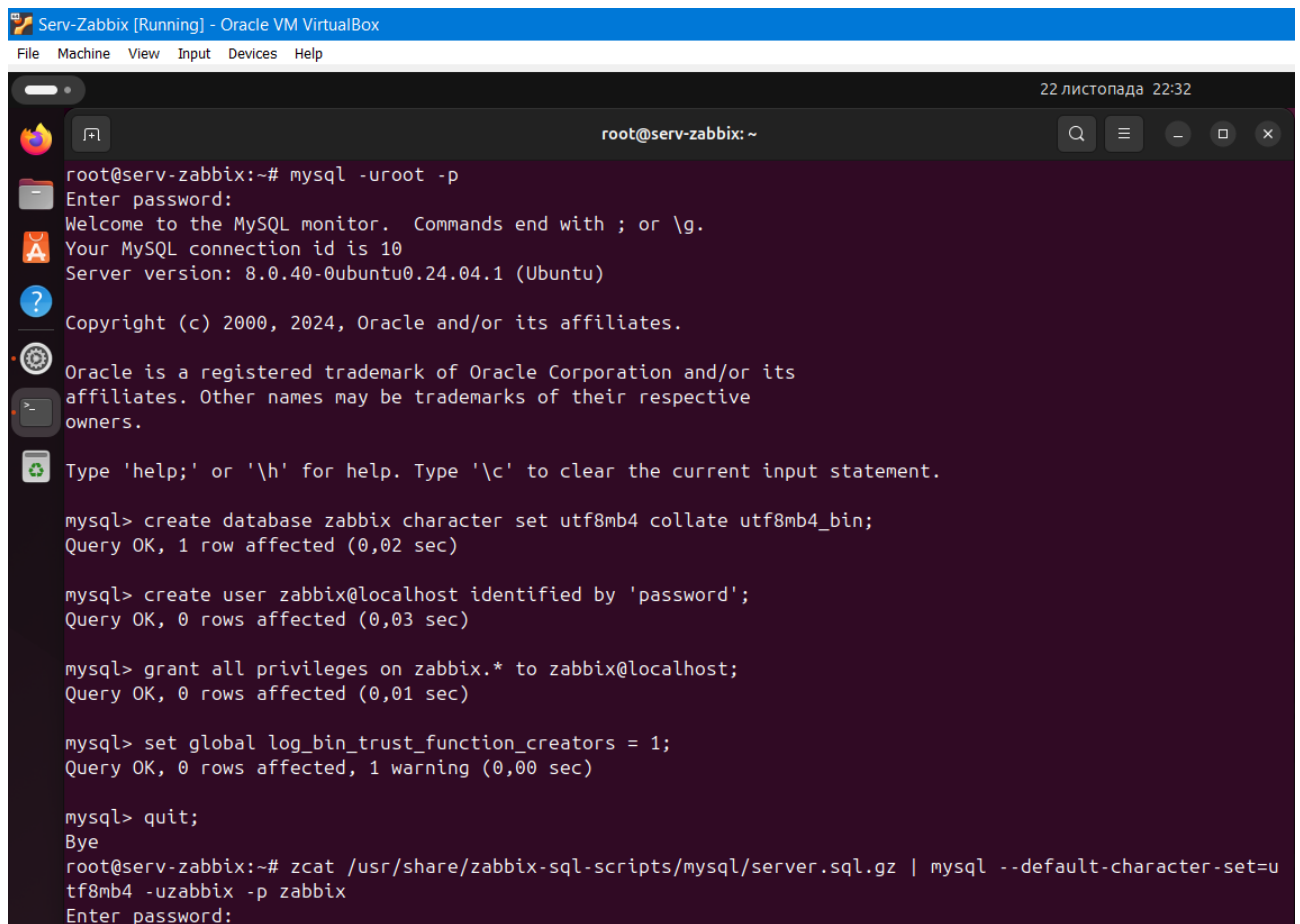


Рисунок 3.22 – Встановлення веб-серверу apache2



```
root@serv-zabbix:~# mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.40-0ubuntu0.24.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database zabbix character set utf8mb4 collate utf8mb4_bin;
Query OK, 1 row affected (0,02 sec)

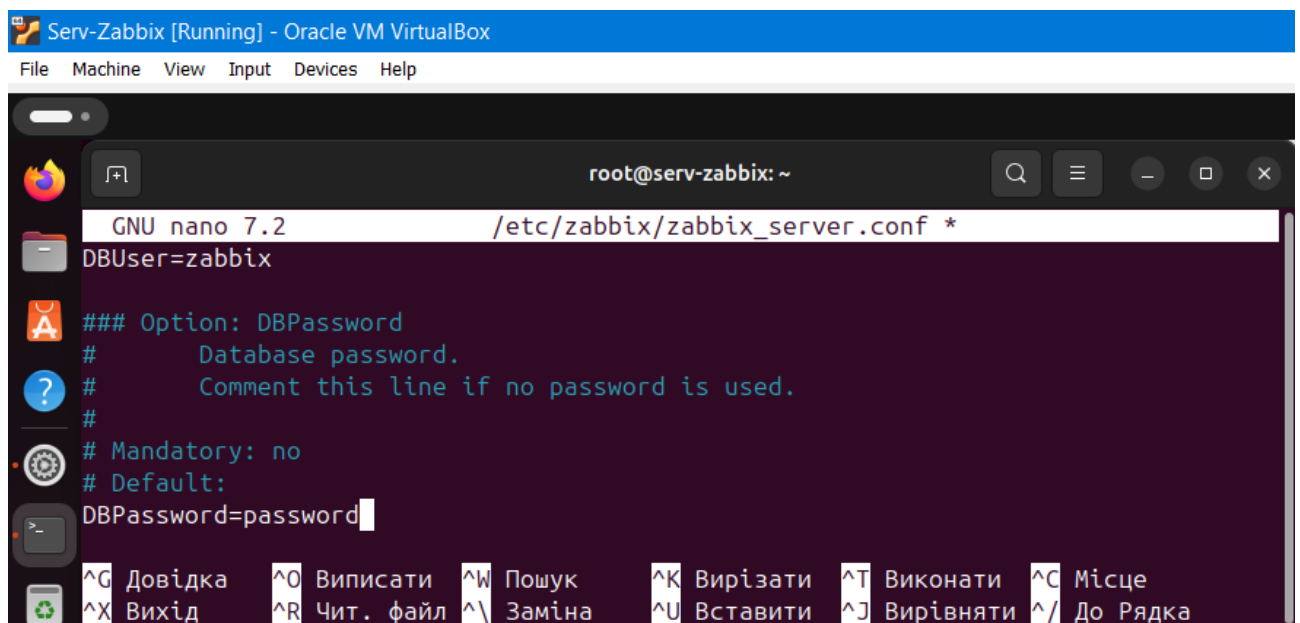
mysql> create user zabbix@localhost identified by 'password';
Query OK, 0 rows affected (0,03 sec)

mysql> grant all privileges on zabbix.* to zabbix@localhost;
Query OK, 0 rows affected (0,01 sec)

mysql> set global log_bin_trust_function_creators = 1;
Query OK, 0 rows affected, 1 warning (0,00 sec)

mysql> quit;
Bye
root@serv-zabbix:~# zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysql --default-character-set=utf8mb4 -uzabbix -p zabbix
Enter password:
```

Рисунок 3.23 – Ініціалізація початкової бази даних



```
GNU nano 7.2 /etc/zabbix/zabbix_server.conf *
DBUser=zabbix

### Option: DBPassword
# Database password.
# Comment this line if no password is used.
#
# Mandatory: no
# Default:
DBPassword=password
```

Рисунок 3.24 – Задання паролю від бази даних в налаштування Zabbix

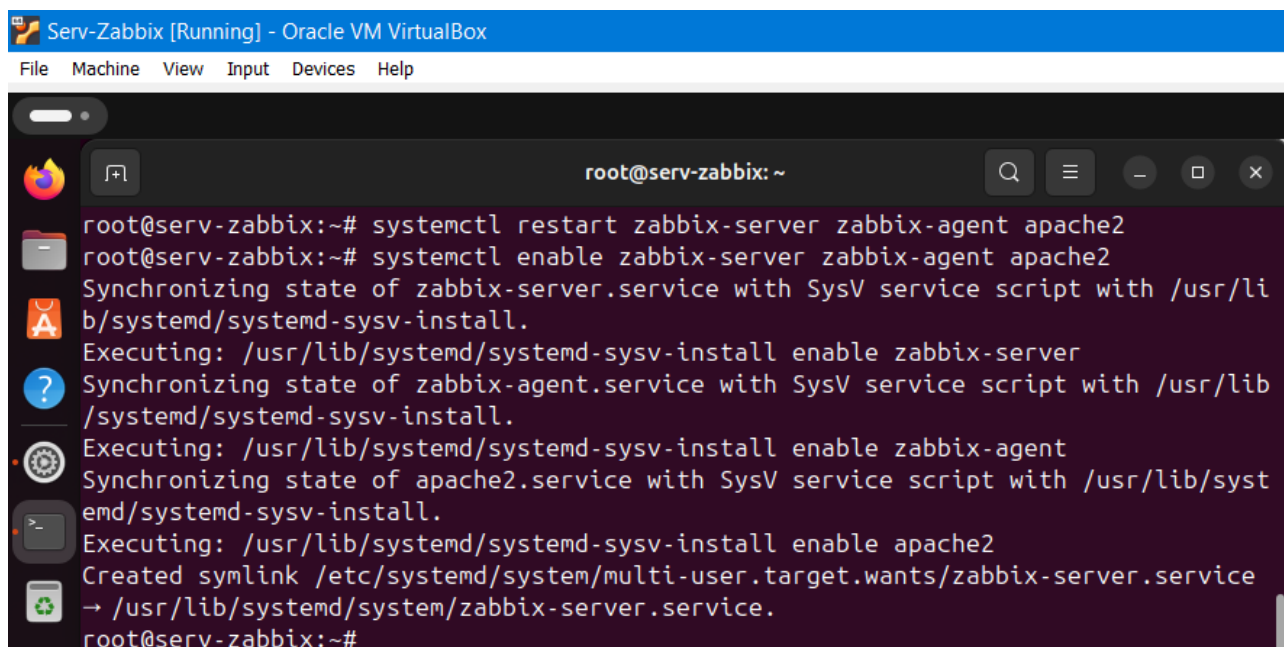


Рисунок 3.25 – Запуск служби Zabbix

Після цього переходимо у веб-інтерфейс Zabbix за посиланням <http://localhost/zabbix>, де проводимо подальше налаштування, рис.3.26.

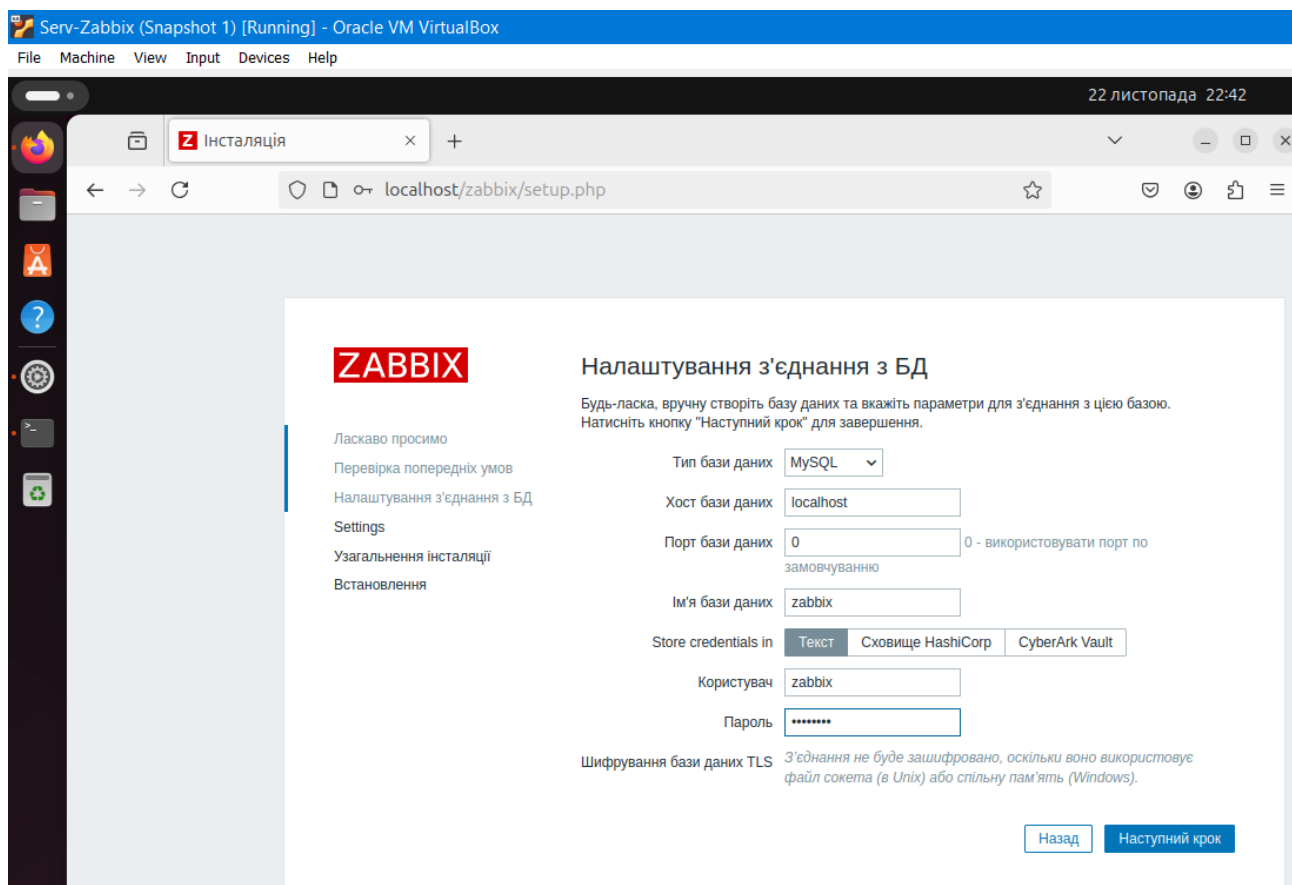


Рисунок 3.26 – З'єднання з базою даних

І використовуючи стандартний логін «Admin» та пароль «zabbix» виконуємо безпосередній вхід до веб-інтерфейсу Zabbix, рис.3.27. З міркувань безпеки доцільно відразу змінити стандартний пароль після цього.

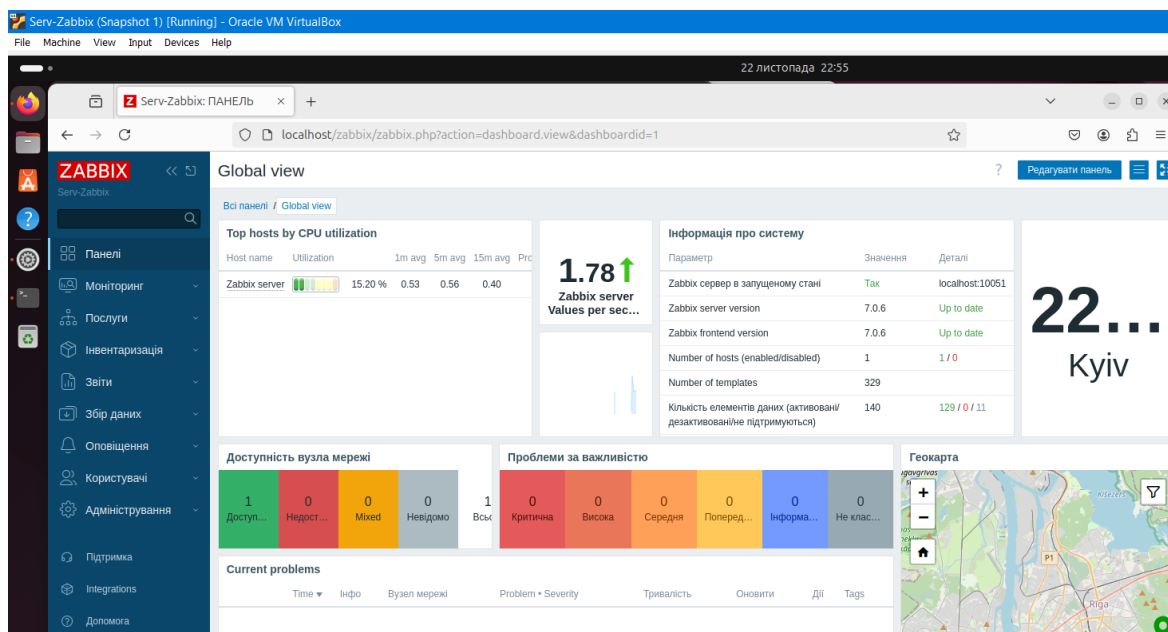


Рисунок 3.27 – Вхід у веб-інтерфейс Zabbix

3.3 Налаштування і перевірка роботи системи моніторингу Zabbix

За замовчуванням на сервері Zabbix вже наявний агент, однак необхідно його налаштувати і на всіх інших серверах і кінцевих робочих станціях користувачів. Для цього переходимо на офіційний сайт Zabbix в розділ агента. За замовчуванням тут вже стоять всі необхідні параметри, однак залежно від версії сервера Zabbix чи операційної системи машини тут необхідно буде внести корективи, рис.3.28. В даному випадку завантажуюмо агент для ОС Windows.

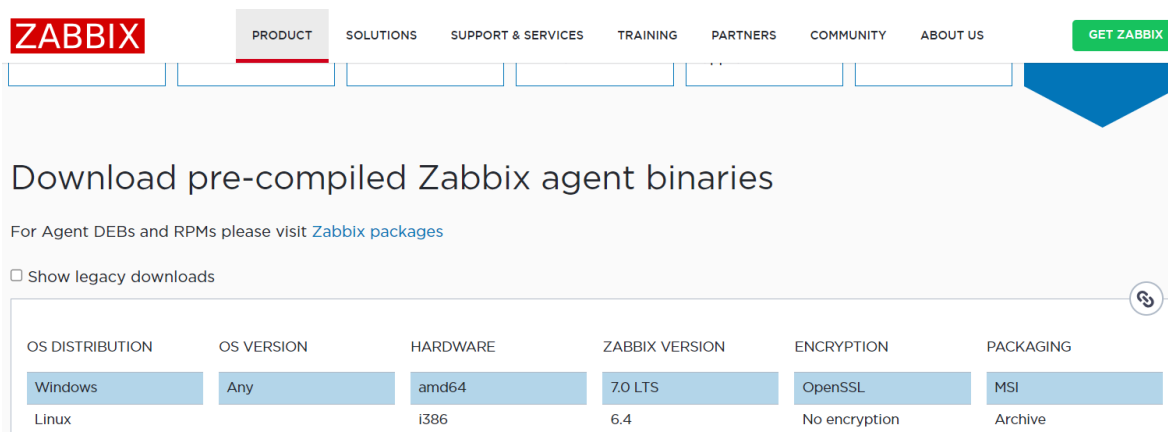


Рисунок 3.28 – Завантаження агента Zabbix

Далі запускаємо завантажений .msi файл з сайту і слідуємо подальшим вказівкам. Для прикладу виконаємо розгортання агента на WS-Serv19-1, але на всіх інших кінцевих станціях ця дія аналогічна, рис.3.29.

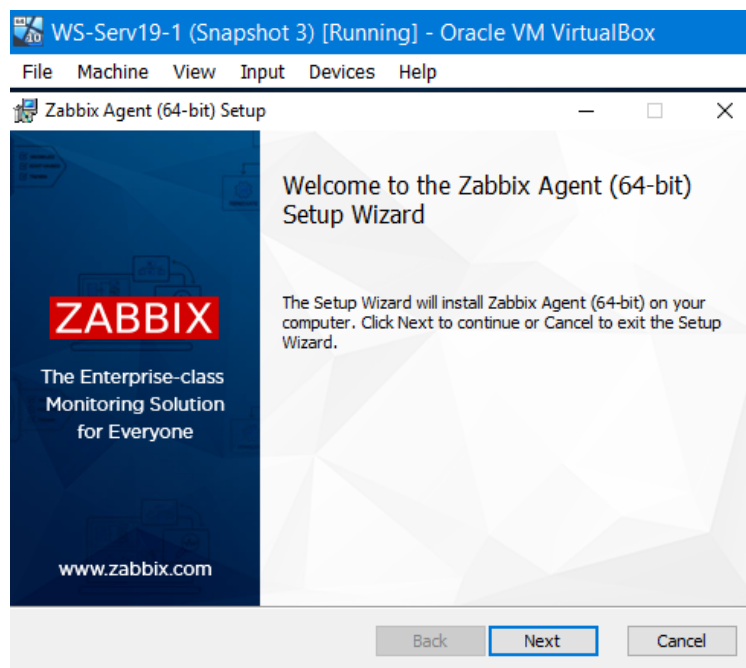


Рисунок 3.29 – Встановлення агента Zabbix

На етапі встановлення необхідно ввести адресу серверу Zabbix або його доменне ім'я, рис.3.30.

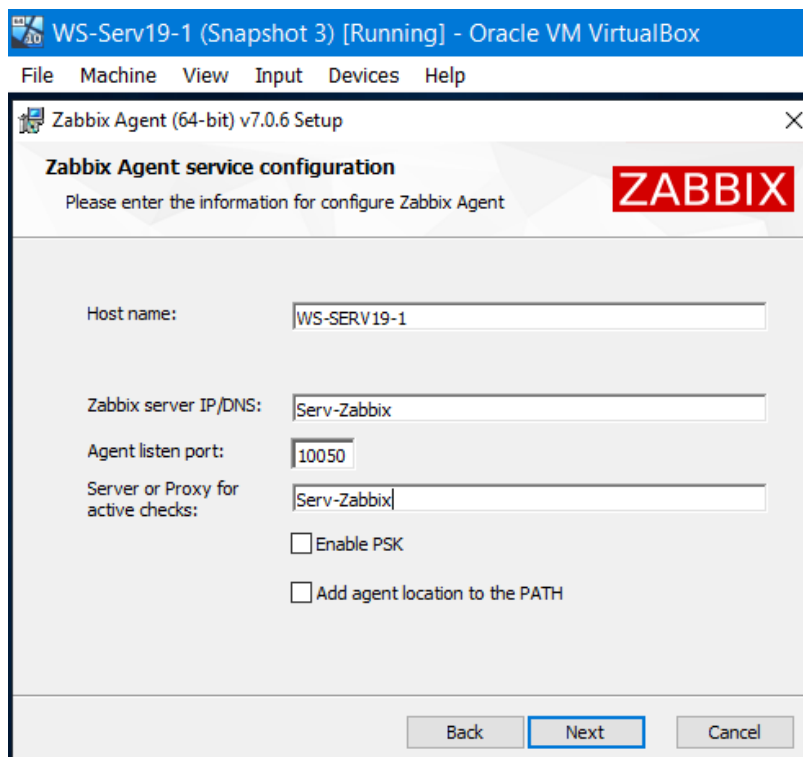


Рисунок 3.30 – Зазначення адреси сервера Zabbix

Повертаємось до серверу Zabbix. У вкладці Моніторинг – Вузли мережі виконаємо додавання даного комп'ютера вказавши всі необхідні дані, рис.3.31. По такому самому принципу виконується додавання інших вузлів мережі.

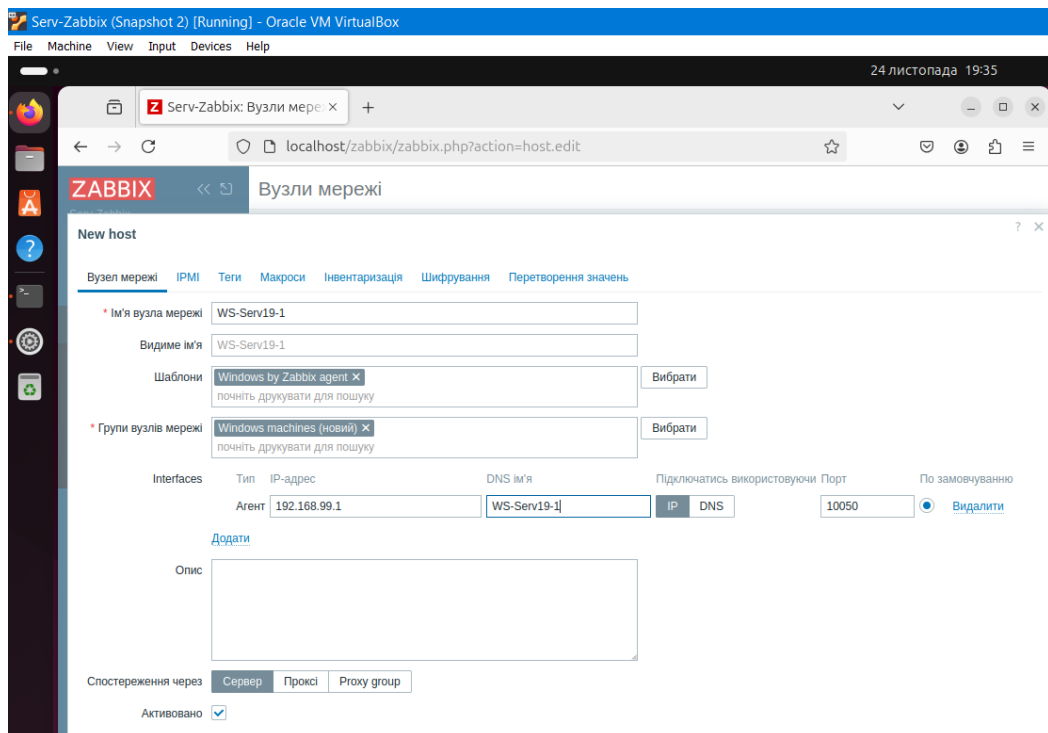


Рисунок 3.31 – Додавання вузла WS-Serv19-1 на сервері Zabbix

Якщо все було виконано правильно, то в даному меню буде відображено стан агентів, які в даний момент є активними, рис.3.32.

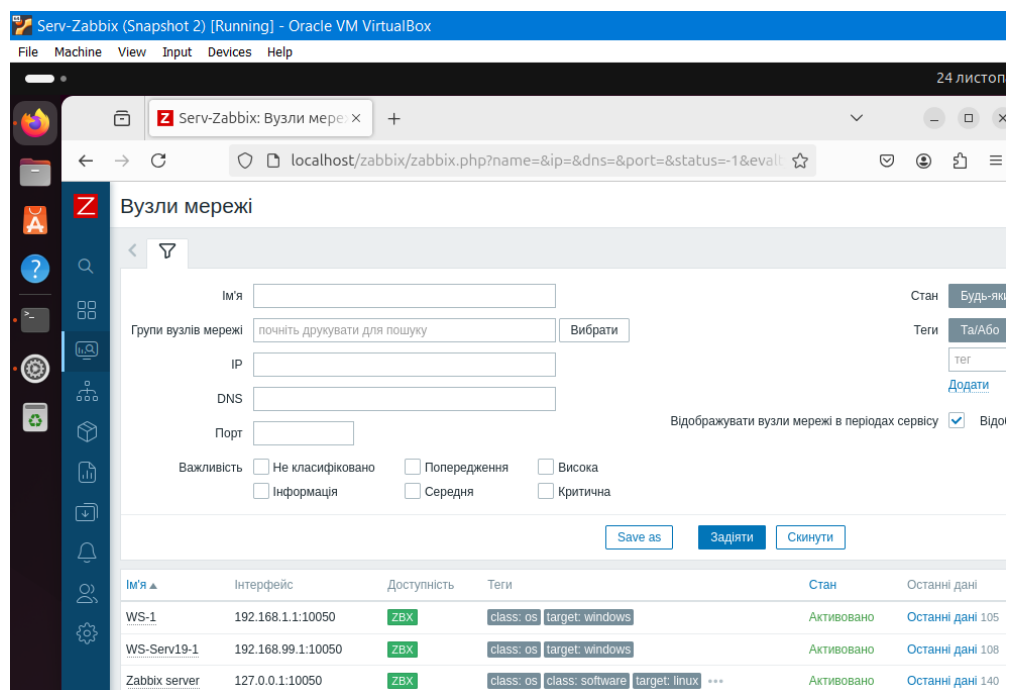


Рисунок 3.32 – Агенти Zabbix активні

Якщо все було виконано правильно, то після цього агенти почнуть передавати дані до сервера. Для прикладу переглянемо останні повідомлення, що надійшли зі WS-Serv19-1, рис.3.33. Переважно тут наведено системні метрики такі як: об'єм використаного місця, роботи процесора, тощо.

Вузел мережі	Ім'я	Остання перевірка	Останнє значення	Зміна	Типи	Інфо
WS-Serv19-1	0 C:: Average disk read queue length	2c	0.000151	-0.03143	component: storage disk: 0 C:	Графік
WS-Serv19-1	0 C:: Average disk write queue length	1c	0.000631	-0.001457	component: storage disk: 0 C:	Графік
WS-Serv19-1	0 C:: Disk average queue size (avgqu-sz)	58c	0.03333	+0.03333	component: storage disk: 0 C:	Графік
WS-Serv19-1	0 C:: Disk read rate	57c	44.9285 r/s	+44.2087 r/s	component: storage disk: 0 C:	Графік
WS-Serv19-1	0 C:: Disk read request avg waiting time	0	0.092mc	-0.19mc	component: storage disk: 0 C:	Графік
WS-Serv19-1	0 C:: Disk utilization by idle time	3c	0.2448 %	-2.8012 %	component: storage disk: 0 C:	Графік
WS-Serv19-1	0 C:: Disk write rate	56c	5.4269 w/s	+2.5742 w/s	component: storage disk: 0 C:	Графік
WS-Serv19-1	0 C:: Disk write request avg waiting time	59c	0.21mc	+0.028mc	component: storage disk: 0 C:	Графік
WS-Serv19-1	Cache bytes	15c	32.01 MB	-892 KB	component: memory	Графік
WS-Serv19-1	Context switches per second	5c	407.4524	-25.5771	component: cpu	Графік
WS-Serv19-1	CPU DPC time	9c	0 %		component: cpu	Графік
WS-Serv19-1	CPU interrupt time	8c	0 %		component: cpu	Графік
WS-Serv19-1	CPU privileged time	7c	3.075 %	+0.000473 %	component: cpu	Графік

Рисунок 3.33 – Отримані дані зі WS-Serv19-1

Можна легко переглянути отримані метрики у вигляді графіків за будь-який час роботи пристроїв перейшовши у відповідну вкладку. Для прикладу нижче наведена частина графіків, що відображають використання процесора WS-Serv19-1, рис.3.34. Повна версія рисунку наведена на додатках Б та В.

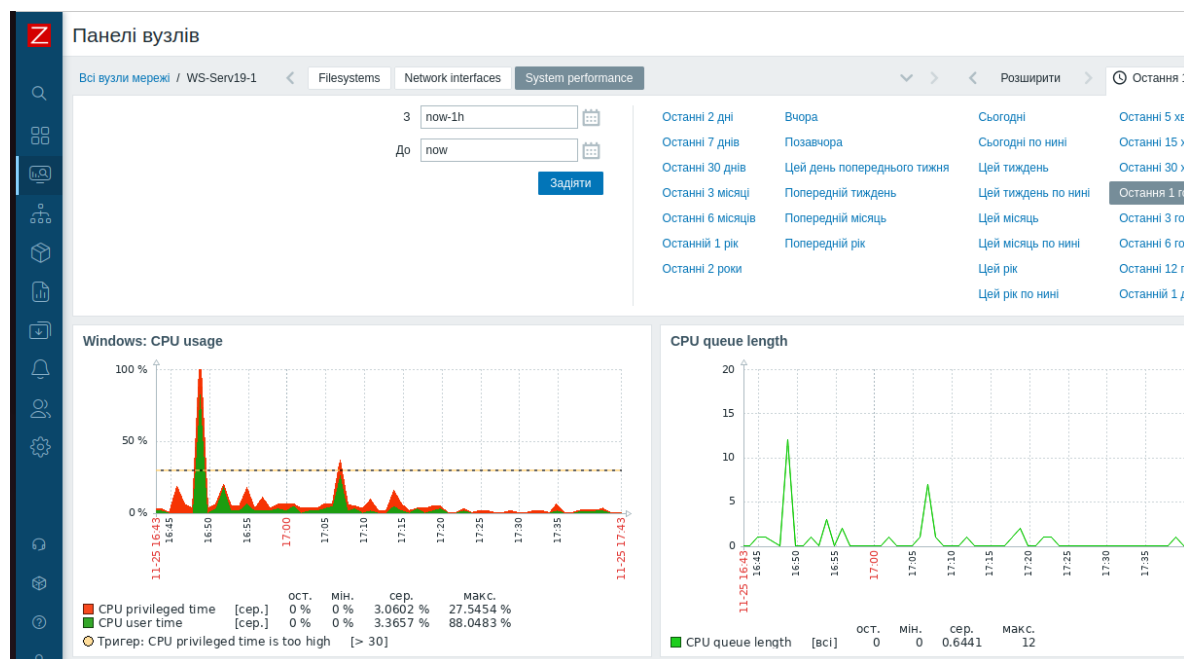


Рисунок 3.34 – Графіки використання процесора на WS-Serv19-1

Також вже можна отримати сповіщення про проблеми на однойменній вкладці. Для прикладу, тут можна побачити, що 24.11.24 о 20:00 та 20:36 було зафіксовано недостатній об'єм вільного місця на комп'ютерах WS-Serv19-1 та WS-1. Далі 25.11.24 об 15:01 було виявлено, що агент на WS-1 недоступний, а вже 15:17 недоступним став агент і на WS-Serv19-1, однак вже о 15:19 дана проблема була вирішена, рис.3.35.

Час	Важливість	Час відновлення	Стан	Інфо	Вузел мережі	Проблема	Тривалість
15:18:37	Попередження		ПРОБЛЕМА		WS-Serv19-1	Host has been restarted (uptime < 10m)	3хв 1с
15:18:14	Попередження	15:18:50	ВИРІШЕНО		WS-Serv19-1	CPU queue length is too high (over 3 for 5m)	36с
15:17:08	Середня	15:19:08	ВИРІШЕНО		WS-Serv19-1	Zabbix agent is not available (for 3m)	2хв
15:01:23	Середня		ПРОБЛЕМА		WS-1	Zabbix agent is not available (for 3m)	20хв 15с
Сьогодні							
24.11.2024 20:36:17	Попередження		ПРОБЛЕМА		WS-1	FS [(C:)]: Space is low (used > 80%, total 19.4GB)	18г 45хв 21с
24.11.2024 20:00:02	Попередження		ПРОБЛЕМА		WS-Serv19-1	FS [(C:)]: Space is low (used > 80%, total 12.5GB)	19г 21хв 36с

Рисунок 3.35 – Наявні виявлені та вирішені проблеми Zabbix

Окремо можна переглянути лише наявні проблеми, рис.3.36, або історію виявлених, але вже усунутих проблем, рис.3.37. Крім цього, можна виконувати фільтрацію пошуку за пристроєм, тегами, часом та іншим властивостям.

Проблеми

Відобразити: ☐ Останні проблеми ☒ Проблеми ☐ Історія

Групи вузлів мережі:

Вузли мережі:

Тригери:

Проблема:

Важливість: ☐ Не класифіковано ☐ Попередження ☐ Висока ☐ Інформація ☐ Середня ☐ Критична

З віком менше ніж: днів

Show symptoms: ☐

Відображувати подавлені проблеми: ☐

Статус підтвердження: ☒ Все ☐ Unacknowledged ☐ Підтверджено ☐ Многу

Інвентарні дані вузла мережі: Тип

Теги:

Відображати теги: ☒ Немає ☐ 1 ☐ 2 ☐ 3 ☐ Ін'я теги ☐ Повне ☐ Скорочено ☐ Немає

Пріоритет відображення тегів:

Show operational data: ☒ Немає ☐ Separately ☐ With problem name

Компактний вигляд: ☐ Відображувати вибір часу: ☒

Відображувати деталі: ☐ Підсвічувати весь рядок: ☐

Час	Важливість	Інфо	Вузел мережі	Проблема	Тривалість	Оновити	Дії	Теги
24.11.2024 20:00:02	Попередження		WS-Serv19-1	FS [(C:)]: Space is low (used > 80%, total 12.5GB)	19г 37хв 38с	Оновити		class: os component: storage filesystem: C: ***

Відображено 1 із 1 знайдених

Рисунок 3.36 – Виявлені проблеми

Час	Важливість	Час відновлення	Стан	Інфо	Вузел мережі	Проблема	Тривалість	Оновити	Дії	Теги
16:38:19	Попередження	16:45:31	ВИРИШЕНО	WS-Serv19-1	Host has been restarted (uptime < 10m)	7хв 12с	Оновити	class: os component: system scope: notice	...	
16:37:59	Попередження	16:44:46	ВИРИШЕНО	WS-1	Host has been restarted (uptime < 10m)	6хв 47с	Оновити	class: os component: system scope: notice	...	
16:36:08	Середня	16:39:08	ВИРИШЕНО	WS-Serv19-1	Zabbix agent is not available (for 3m)	3хв	Оновити	class: os component: system scope: availability	...	
16:35:30	Попередження	16:45:00	ВИРИШЕНО	Zabbix server	Zabbix server has been restarted (uptime < 10m)	9хв 30с	Оновити	class: os component: system scope: notice	...	
16:00										
15:38:23	Середня	16:38:23	ВИРИШЕНО	WS-1	Zabbix agent is not available (for 3m)	1г	Оновити	class: os component: system scope: availability	...	
15:38:08	Середня	15:44:08	ВИРИШЕНО	WS-Serv19-1	Zabbix agent is not available (for 3m)	6хв	Оновити	class: os component: system scope: availability	...	
15:18:37	Попередження	15:25:01	ВИРИШЕНО	WS-Serv19-1	Host has been restarted (uptime < 10m)	6хв 24с	Оновити	class: os component: system scope: notice	...	
15:18:14	Попередження	15:18:50	ВИРИШЕНО	WS-Serv19-1	CPU queue length is too high (over 3 for 5m)	36с	Оновити	class: os component: cpu scope: performance	...	
15:17:08	Середня	15:19:08	ВИРИШЕНО	WS-Serv19-1	Zabbix agent is not available (for 3m)	2хв	Оновити	class: os component: system scope: availability	...	
15:01:23	Середня	15:31:23	ВИРИШЕНО	WS-1	Zabbix agent is not available (for 3m)	30хв	Оновити	class: os component: system scope: availability	...	
15:01:02	Попередження	15:08:01	ВИРИШЕНО	WS-Serv19-1	Host has been restarted (uptime < 10m)	6хв 59с	Оновити	class: os component: system scope: notice	...	
15:00										
14:58:00	Попередження	15:07:30	ВИРИШЕНО	Zabbix server	Zabbix server has been restarted (uptime < 10m)	9хв 30с	Оновити	class: os component: system scope: notice	...	

Відображено 12 із 12 знайдених

Рисунок 3.37 – Історія виявлених та вирішених проблем

Захист передачі даних в Zabbix.

За замовчуванням дані від агентів до сервера Zabbix передаються у відкритому вигляді, що дає змогу злоумисникам виконати перехоплення даних в мережі та отримати певні конфіденційні дані стосовно роботи кінцевих станцій, рис.3.38.

Zabbix-capture.pcapng [SW-8-A1 Ethernet3/2 to R-8-A1 FastEthernet0/1]

Файл Правка Вигляд Перехід Захоплення Аналіз Статистика Телефонія Wireless Інструменти Довідка

_ws.col.protocol == "Zabbix"

Time	No.	Source	Destination	Protocol	Leng	Info
213.306347	74360	192.168.99.2	192.168.1.1	Zabbix	180	Zabbix Server/proxy request for passive agent checks, Len=113 (6022)
213.313221	74363	192.168.1.1	192.168.99.2	Zabbix	128	Zabbix Agent response for passive checks, Len=61 (10050 + 60222)
214.305406	74520	192.168.99.2	192.168.1.1	Zabbix	184	Zabbix Server/proxy request for passive agent checks, Len=117 (6022)
214.305421	74521	192.168.99.2	192.168.1.1	Zabbix	165	Zabbix Server/proxy request for passive agent checks, Len=98 (6024)

> Frame 74520: 184 bytes on wire (1472 bits), 184 bytes captured (1472 bits) on 0

> Ethernet II, Src: ca:03:06:22:00:1d (ca:03:06:22:00:1d), Dst: PCSSystemtec

> Internet Protocol Version 4, Src: 192.168.99.2, Dst: 192.168.1.1

> Transmission Control Protocol, Src Port: 60234, Dst Port: 10050, Seq: 1, Ac

✓ Zabbix Server/proxy request for passive agent checks, Len=117

Wireshark · Follow TCP Stream (tcp.stream eq 519) · Zabbix-capture.pcap...

```

ZBXD.u.....{"request":"passive checks","data":[{"key":"perf_counter_en[\\
\"\\PhysicalDisk(0 C:)\\% Idle Time\",60]\",\"timeout\":3}]}
ZBXD.>.....{\"version\":\"7.0.6\",\"variant\":\"1\",\"data\":{\"value\":\"39.863972\"}}

```

Рисунок 3.38 – Перехоплені пакети Zabbix між агентом та сервером

Тому налаштування PSK (Pre-Shared Key) Encryption у Zabbix є важливим кроком для забезпечення безпеки переданих даних між сервером і його агентами. Ця функція шифрує трафік, використовуючи заздалегідь узгоджений ключ, що

дозволяє захистити дані від перехоплення та несанкціонованого доступу. Задля цього з використання утиліти openssl виконуємо генерацію випадкового 256-бітного ключа, рис.3.39.

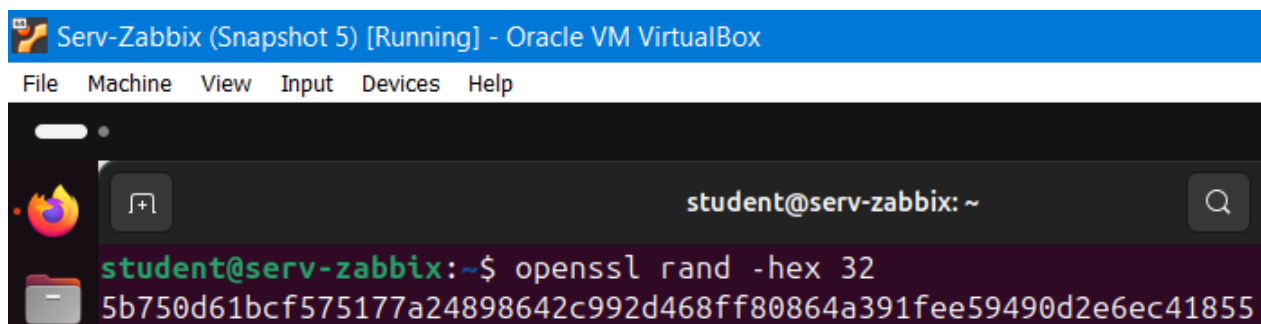


Рисунок 3.39 – Створення випадкового 256-бітного ключа

На сервері Zabbix для кожного доданого вузла виставляємо тип шифрування «PSK», вводимо ім'я для даного ключа та вставляємо сам згенерований ключ, рис.3.40.

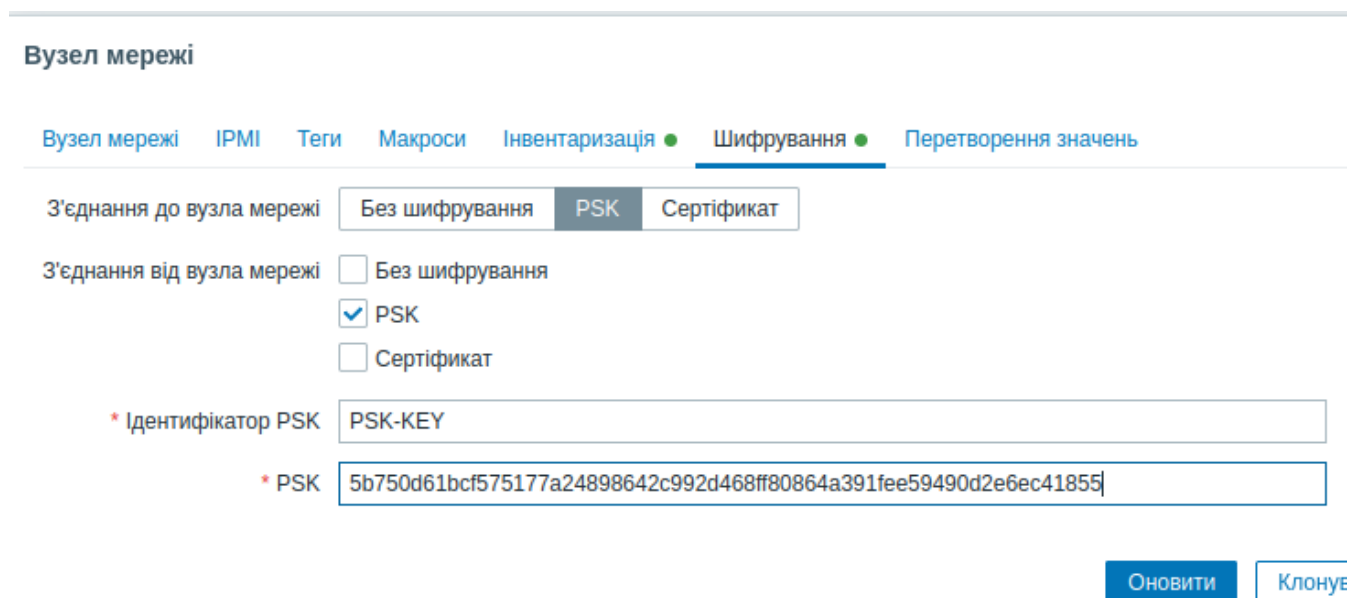


Рисунок 3.40 – Додання ключа PSK для вузлів на сервері Zabbix

Переходимо до налаштувань шифрування на агентах. В папці де розміщено агент створюємо текстовий файл «zabbix-psk» в який вставляємо раніше згенерований ключ PSK, рис.3.40. З метою безпеки вкрай важливо, щоб доступ до даної папки мав лише адміністратор.

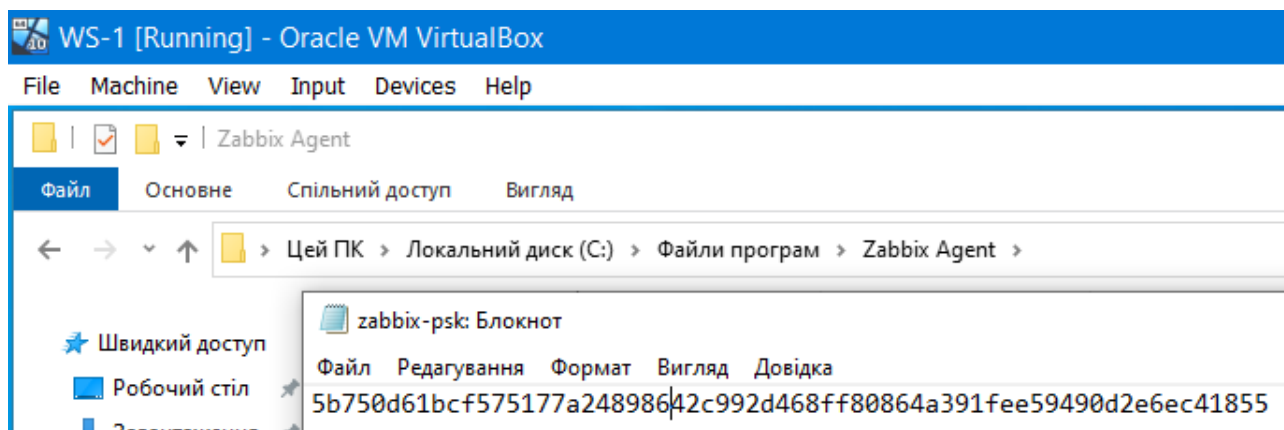


Рисунок 3.41 – Створення текстового файлу з ключем PSK на агенті

В цьому ж каталозі відкриваємо файлу конфігурацій «zabbix_agentd.conf» і додаємо наступні строки для налаштування роботи шифрування, рис.3.42.

TLSConnect=psk

TLSAccept=psk

TLSPSKFile=C:\Program Files\Zabbix Agent\zabbix-psk.txt

TLSPSKIdentity=PSK-KEY

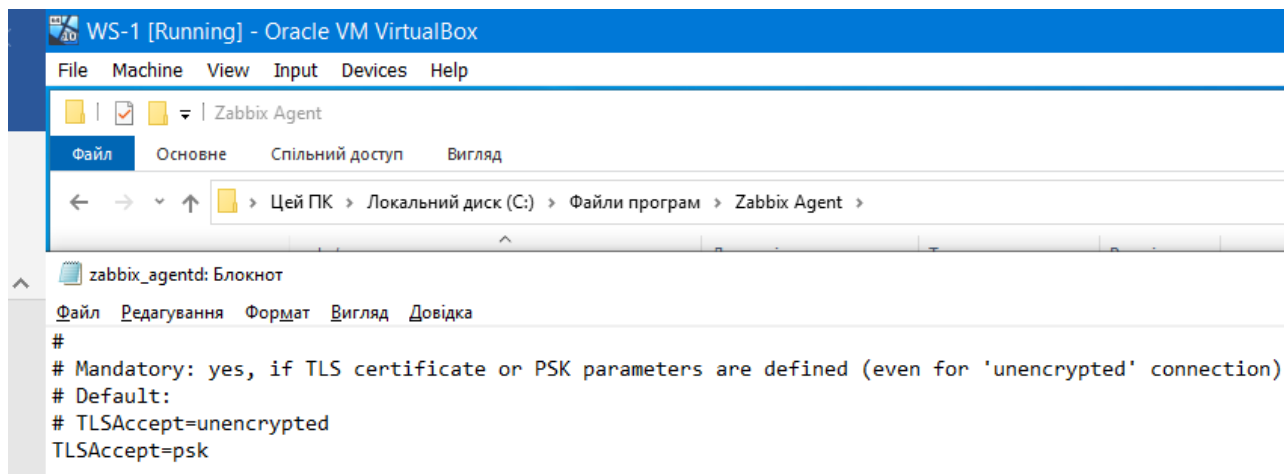


Рисунок 3.42 – Налаштування шифрування на агенті з використанням PSK

Після цього перезапускаємо агент і знову виконаємо перехоплення даних, рис.3.43. Однак тепер всі вони передаються в шифрованому вигляді з використанням протоколу TLS, що свідчить про правильну налаштовану роботу ключів PSK.

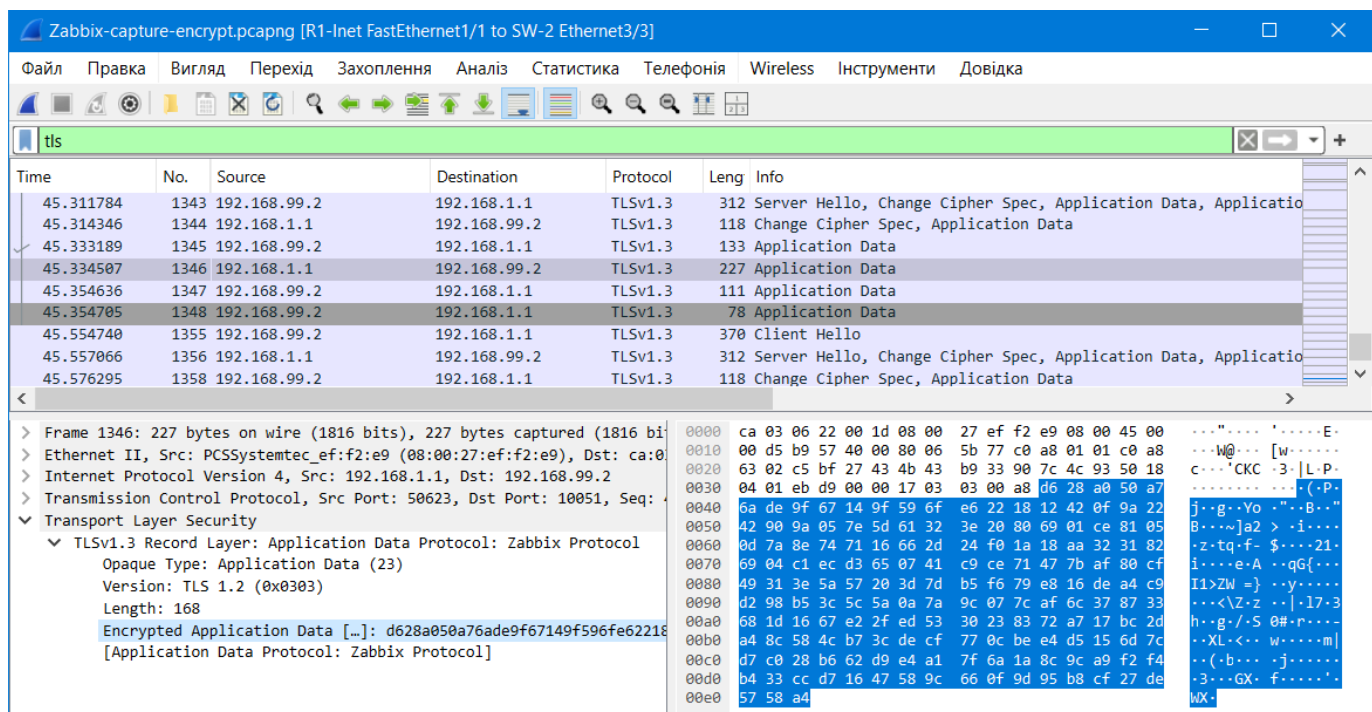


Рисунок 3.43 – Перехоплені пакети Zabbix між агентом та сервером
Налаштування тригерів.

Zabbix надає гнучкі можливості для створення власних правил генерування сповіщень, так званих тригерів. Для прикладу, створимо тригер, щоб у випадку хибно введеного паролю на будь-якій Windows станції в Zabbix генерувалося відповідне сповіщення з рівнем важливості «Попередження».

Для цього спершу необхідно створити елемент даних для шаблону «Windows by Zabbix agent», який буде відповідати за збір подій хибно виконаного входу, тобто події з ID 4625 рис.3.44.

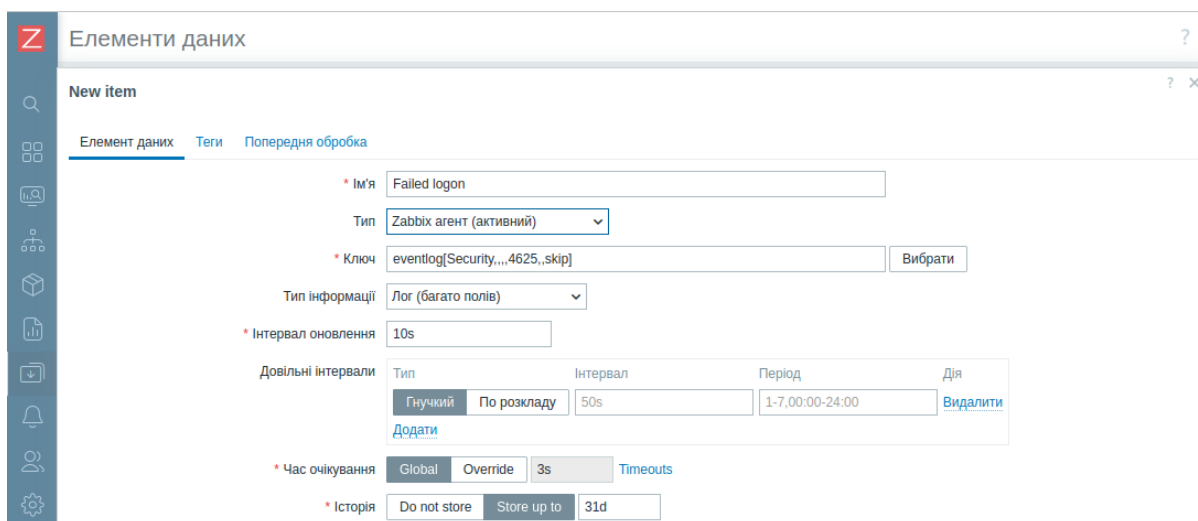


Рисунок 3.44 – Створення елемента даних на збір подій з ID 4625

Перевіримо роботу створеного елемента даних ввівши хибний пароль для користувача usr-aa0 (Andrii Anchys) на станції WS-1, рис.3.45. Після цього бачимо, що на сервері Zabbix відобразилась дана інформація, рис.3.46.

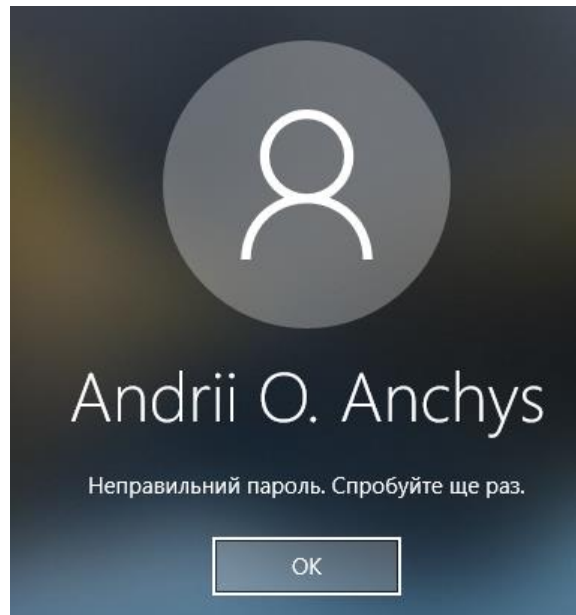


Рисунок 3.45 – Хибно введений пароль

WS-1: Failed logon

З: now-1h
До: now
Задіяти

Останні 2 дні
Останні 7 днів
Останні 30 днів
Останні 3 місяці
Останні 6 місяців
Останній рік
Останні 2 роки

Вчора
Позавчора
Цей день попереднього тижня
Попередній тиждень
Попередній місяць
Попередній рік

Сьогодні
Сьогодні по нині
Цей тиждень
Цей тиждень по нині
Цей місяць
Цей місяць по нині
Цей рік
Цей рік по нині

Час виникнення (штамп)	Локальний (дійсний) час	Джерело	Важливість	ID події	Значення
25.11.2024 18:27:36	25.11.2024 18:27:34	Microsoft-Windows-Security-Auditing	Аудит відмов	4625	An account failed to log on.

Subject:

Security ID: NT AUTHORITY\SYSTEM

Account Name: WS-1\$

Account Domain: ANCHYS

Logon ID: 0x3E7

Logon Type: 2

Account For Which Logon Failed:

Security ID: NULL SID

Account Name: usr-aa0

Рисунок 3.46 – Отримана інформація про подію з ID 4625 на WS-1

Тепер потрібно безпосередньо налаштувати тригер, рис.3.47.

Тригери

New trigger

Тригер Теги Залежності

* Ім'я: Failed logon

Назва події: Failed logon

Operational data:

Важливість: Не класифіковано Інформація **Попередження** Середня Висока Критична

* Вираз: logeventid(/Windows by Zabbix agent/ event log[Security,,,4625,,skip])=1 [Додати](#)

[Конструктор виразу](#)

генерація події ОК: Вираз Вираз для відновлення Немає

Режим генерації подій ПРОБЛЕМА: Один раз Багатократно

подія ОК закриває: Всі проблеми Всі проблеми якщо значення тегів співпадають

Дозволити закриття вручну ☐

Рисунок 3.47 – Налаштування тригера

І в списку виявлених проблем на WS-1 бачимо дану подію і можемо переглянути додаткові відомості рис.3.48-3.49. Схожим чином можна налаштовувати тригери для будь-яких інших подій Windows.

☐	Час	Важливість	Час відновлення	Стан	Інфо	Вузел мережі	Проблема	Тривалість	Оновити	Дії	Теги
☐	18:46:16	Попередження		ПРОБЛЕМА		WS-1	Failed logon	3хв 24с	Оновити		class: os target: wind
☐	18:06:17	Попередження		ПРОБЛЕМА		WS-1	FS [(C:)]: Space is low (used > 80%, total 19.4GB)	43хв 23с	Оновити		class: os component: filesystem: C: ***

Рисунок 3.48 – Виявлення проблем з хибним входом в систему

25.11.2024 18:46:16 25.11.2024 18:46:13 Microsoft-Windows-Security-Auditing Аудит відмов 4625 An account failed to log on.

Subject:

Security ID: NT AUTHORITY\SYSTEM

Account Name: WS-1\$

Account Domain: ANCHYS

Logon ID: 0x3E7

Logon Type: 2

Account For Which Logon Failed:

Security ID: NULL SID

Account Name: usr-aa0

Account Domain: ANCHYS

Рисунок 3.49 – Перегляд детальної інформації про подію

Подібним чином створимо елемент збору подій Windows Defender з рівнем сповіщення «Середня» на події з ID 1116 (Загроза була виявлена), 1117 (Загроза була заблокована), 1118 (Загроза була успішно видалена), рис.3.50.

Рисунок 3.50 – Створення елемента даних на збір подій з ID 1116, 1117, 1118
І налаштуємо тригер на сповіщення виявлених подій з ID 1116, 1117, 1118 з рівнем сповіщення «Середня», рис.3.51.

Рисунок 3.51 – Налаштування тригера на події з ID 1116

Далі на робочій станції завантажимо тестовий зловмисний файл «eicar.com.txt». Відразу після цього з'явилось сповіщення від Windows Defender про виявлення та видалення загрози, рис.3.52

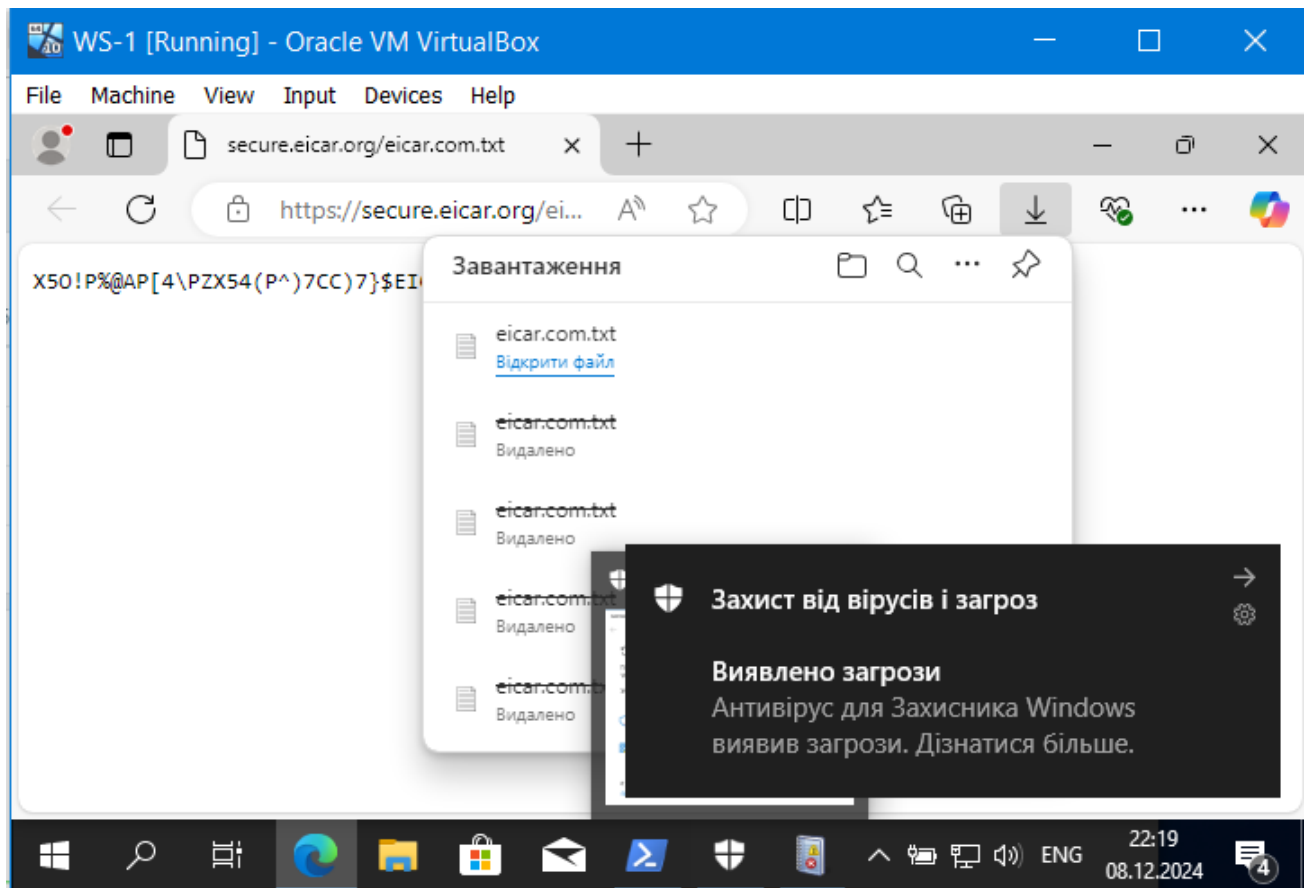


Рисунок 3.52 – Завантаження тестового зловмисного файлу

Повернемося до серверу Zabbix. Тут одразу з'явилась нова проблем «Windows Defender Events», рис.3.53.

☐	Час ▾	Важливість	Час відновлення	Стан	Інфо	Вузел мережі	Проблема	Тривалість	Оновити	Дії	Теги
☐	22:19:50	Середня		ПРОБЛЕМА		WS-1	Windows Defender events	3хв 26с	Оновити		class: os target: windows

Рисунок 3.53 – Виявлена проблема

Можемо дослідити детальніше дану проблему та з'ясувати, який завантажений файл спровокував даний тригер та хто це зробив, рис.3.54.

08.12.2024 22:19:50	08.12.2024 22:19:44	Microsoft-Windows-Windows Defender	Попередження	1116	Microsoft Defender Antivirus has detected malware or other potential unwanted software. For more information please see the following: https://go.microsoft.com/fwlink/?linkid=37020&name=Virus:DOS/EICAR_Test_File&threatid=2147519003&enterprise=0 Name: Virus:DOS/EICAR_Test_File ID: 2147519003 Severity: Критична Category: Bipyс Path: file:_C:\Users\usr-aao\AppData\Local\Temp\7d9fa3dc-4e74-4f5a-a3d3-741364b80875.tmp; webfile: C:\Users\usr-aao\AppData\Local\Temp\7d9fa3dc-4e74-4f5a-a3d3-741364b80875.tmp https://secure.eicar.org/eicar.com.txt pid:5968,ProcessStart:133781627838055780 Detection Origin: Internet Detection Type: Concrete Detection Source: Downloads and attachments User: ANCHYS\usr-aao Process Name: Unknown
---------------------	---------------------	------------------------------------	--------------	------	--

Рисунок 3.54 – Отримана подія виявлення загрози

Відправка сповіщень.

Zabbix надає гнучкі можливості для відправки сповіщень про проблеми в системі моніторингу. Він підтримує різноманітні медіа-типи, такі як електронна пошта, SMS, месенджери (Telegram, Slack, Microsoft Teams), вебхуки та інші сервіси в залежності від потреб організації. В даному проєкті виконаємо налаштування відправки сповіщень всіх рівнів на електронну пошту Gmail. Для цього вказуємо пошту, яка буде використовуватись для відправлень та пароль додатку від неї, який необхідно згенерувати заздалегідь, рис.3.55.

?

×

Тип засобу передачі

Тип засобу передачі

Message templates 5

Опції

* Ім'я

Gmail

Тип

Email

Постачальник пошти

Gmail

* Email

andrianchys2024@gmail.com

* Пароль

Формат повідомлення

HTML

Текст

Опис

Активовано

☒

Рисунок 3.55 – Налаштування передачі сповіщень через Gmail

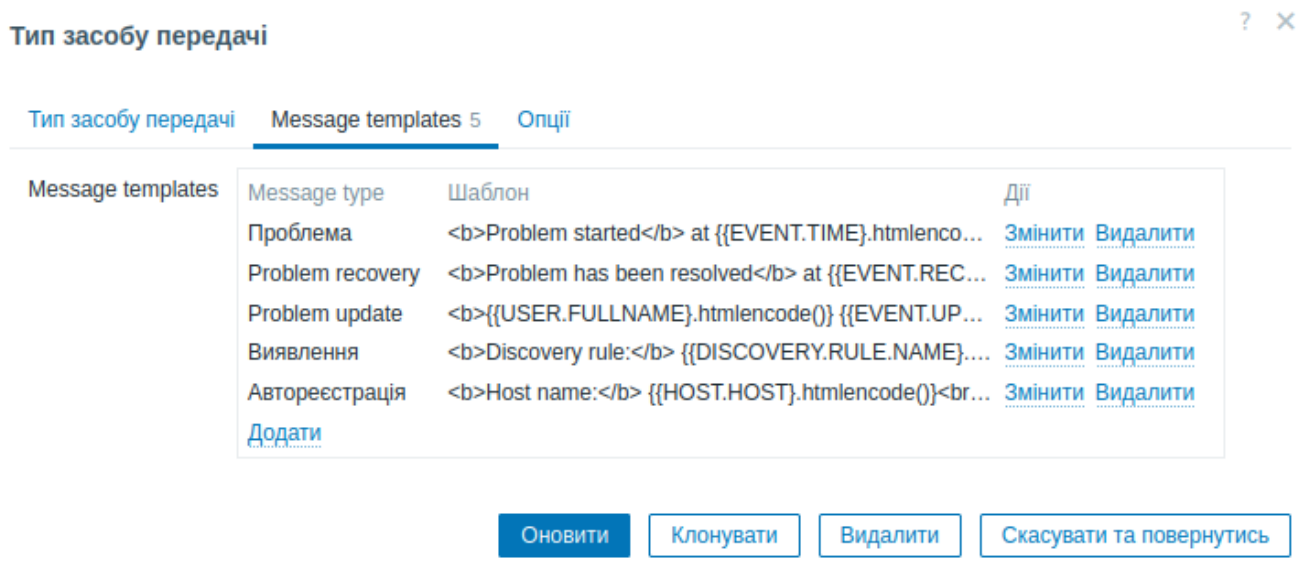


Рисунок 3.56 – Налаштування передачі сповіщень через Gmail

Далі в налаштуваннях користувача налаштовуємо спосіб передачі сповіщень з використанням Gmail. Тут вказується адреса отримувача, дні та години коли може здійснюватися відправка повідомлень та рівні важливості. Однак в реальному середовищі доцільно залишати сповіщення лише рівня середні та вище, аби пошта адміністратора не була переповнена повідомленнями, рис.3.57.

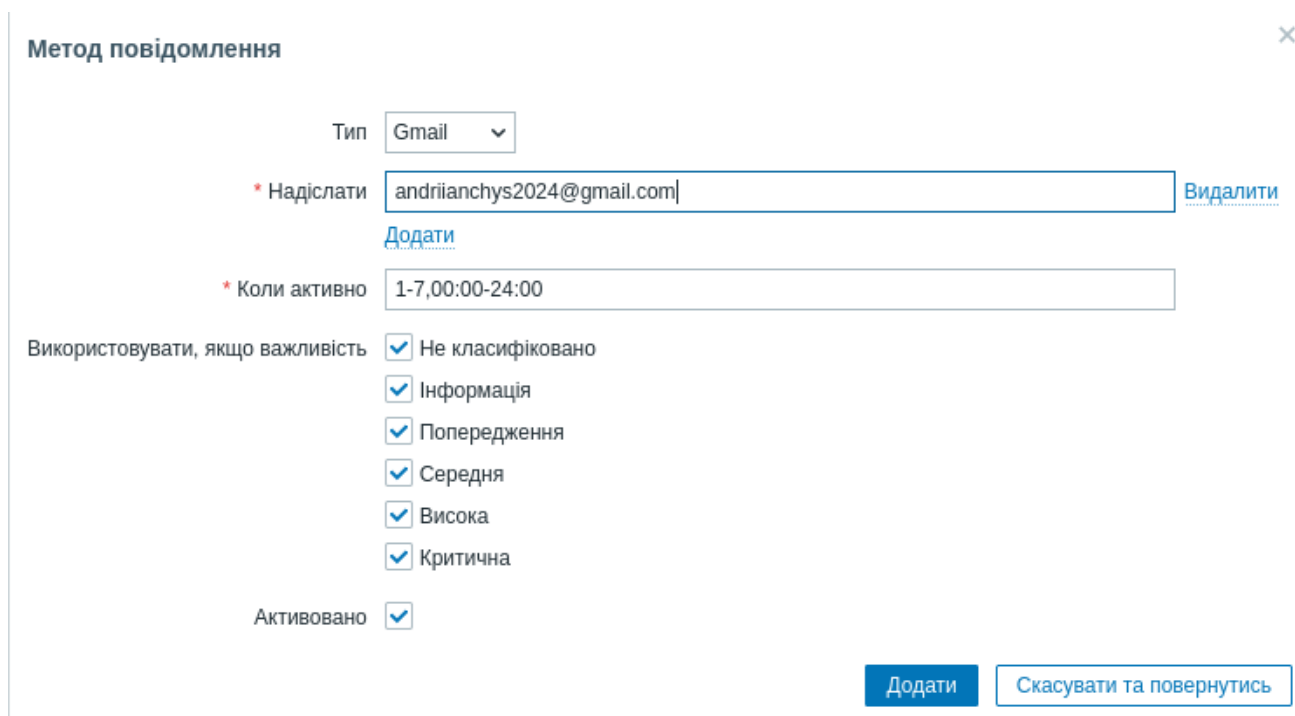


Рисунок 3.57 – Налаштування передачі сповіщень через Gmail

І якщо все було виконано правильно, то на вказану пошту почнуть надходити повідомлення про проблеми різного рівня, рис.3.58-3.59.

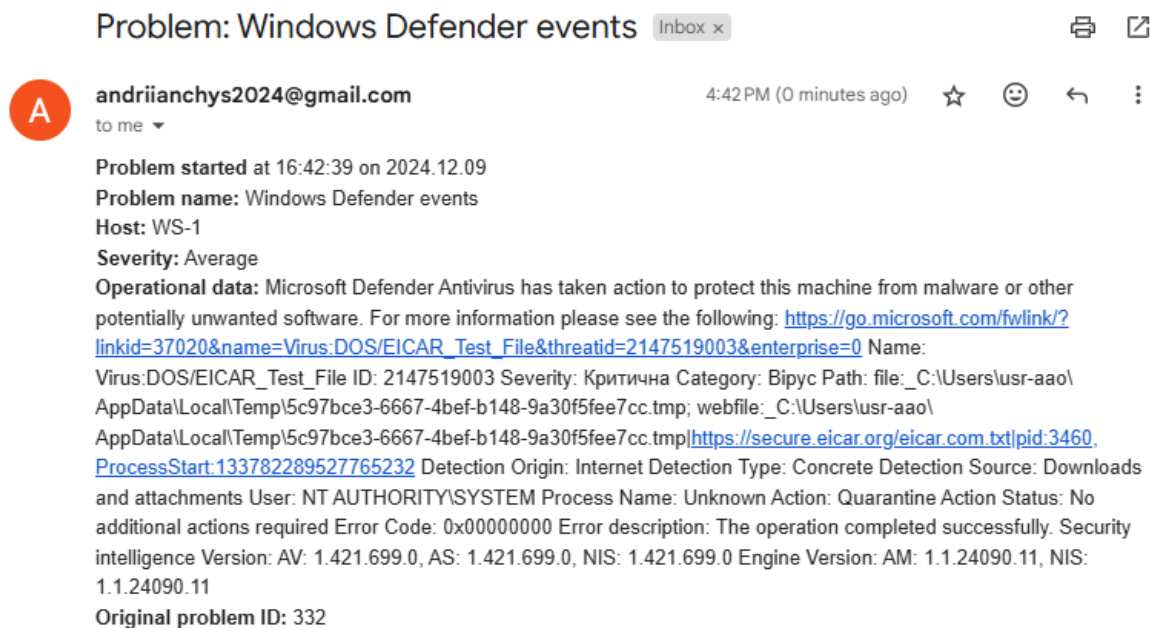


Рисунок 3.58 – Отримане повідомлення про проблему з виявленою загрозою

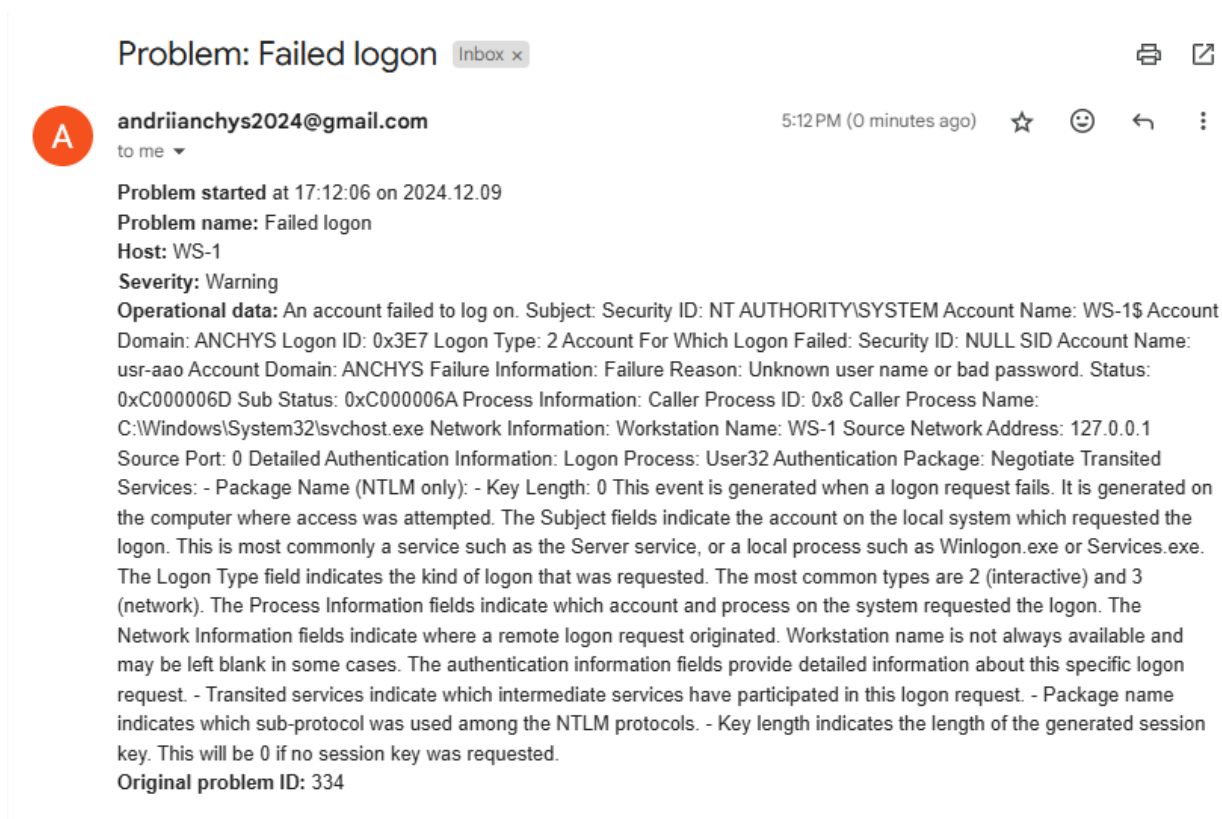


Рисунок 3.59 – Отримане повідомлення про хибно введений пароль користувача

Створення карти мережі дозволить візуалізувати взаємозв'язки між елементами інфраструктури. Це сприятиме швидкому виявленню проблемних вузлів, розумінню впливу інцидентів, полегшенню аналізу залежностей і тим самим підвищує ефективність моніторингу і швидкість реагування, рис.3.60.

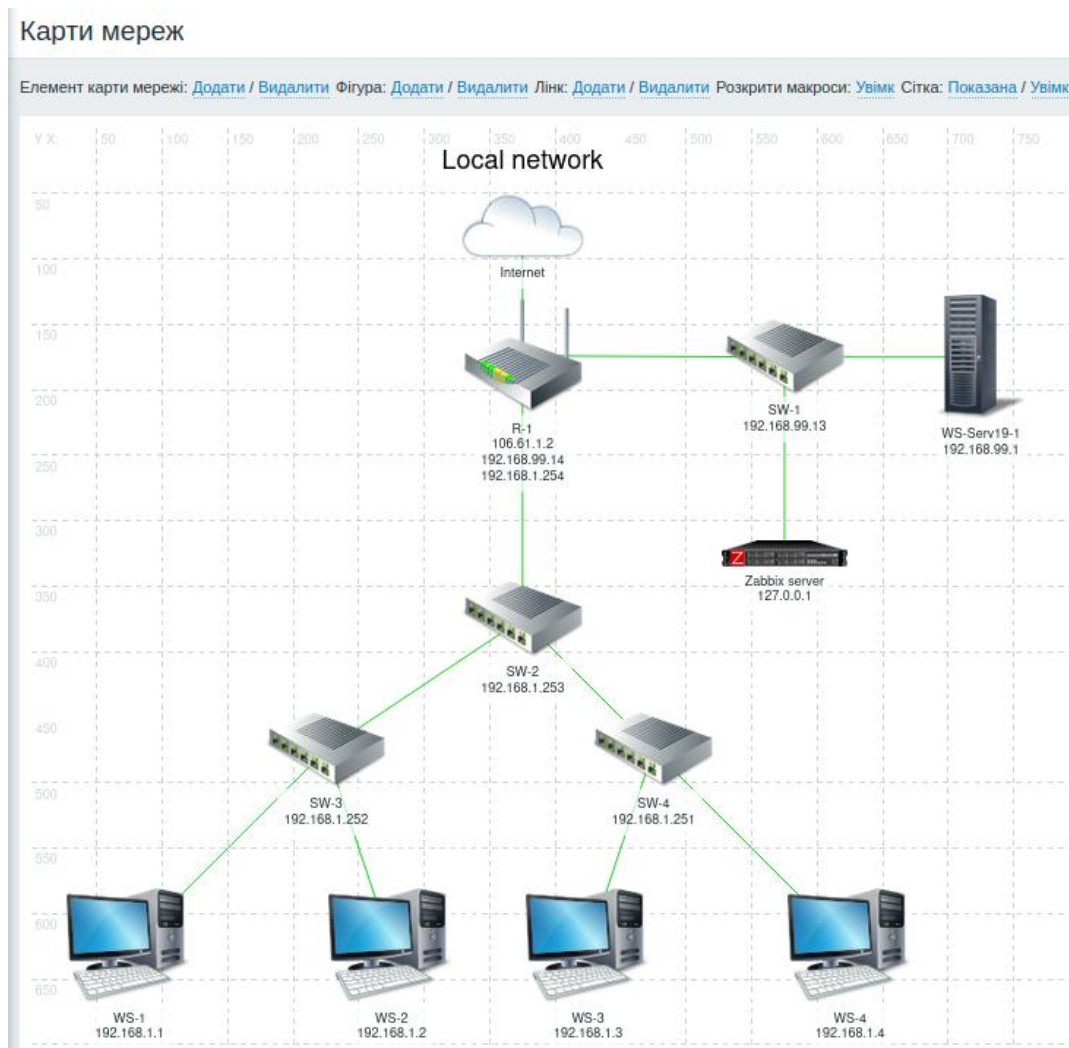


Рисунок 3.60 – Створена карта топології мережі в Zabbix

Інвентаризація в Zabbix - це функція для зберігання та управління даними про обладнання, програмне забезпечення та інші ресурси мережі. Вона дозволяє автоматично або вручну заповнювати інформацію про вузли, таку як серійні номери, IP-адреси, розташування, власників, тип і багато іншого. Це корисно для ведення обліку активів, спрощення процесів управління інфраструктурою, швидкого пошуку інформації про пристрої. Нижче наведено приклад заповнених даних для сервера WS-Serv19-1, рис.3.61.

Рисунок 3.61 – Заповнення додаткових даних для вузла

Також Zabbix надає широкі можливості щодо персоналізації панелі моніторингу для різних потреб і ролей. Можна легко додавати та налагоджувати віджети для відображення графіків, таблиць, подій, тригерів, карт та іншого. Це забезпечує високу гнучкість у візуалізації даних, допомагає швидко отримувати потрібну інформацію та покращує швидкість прийняття рішень, рис.3.62.

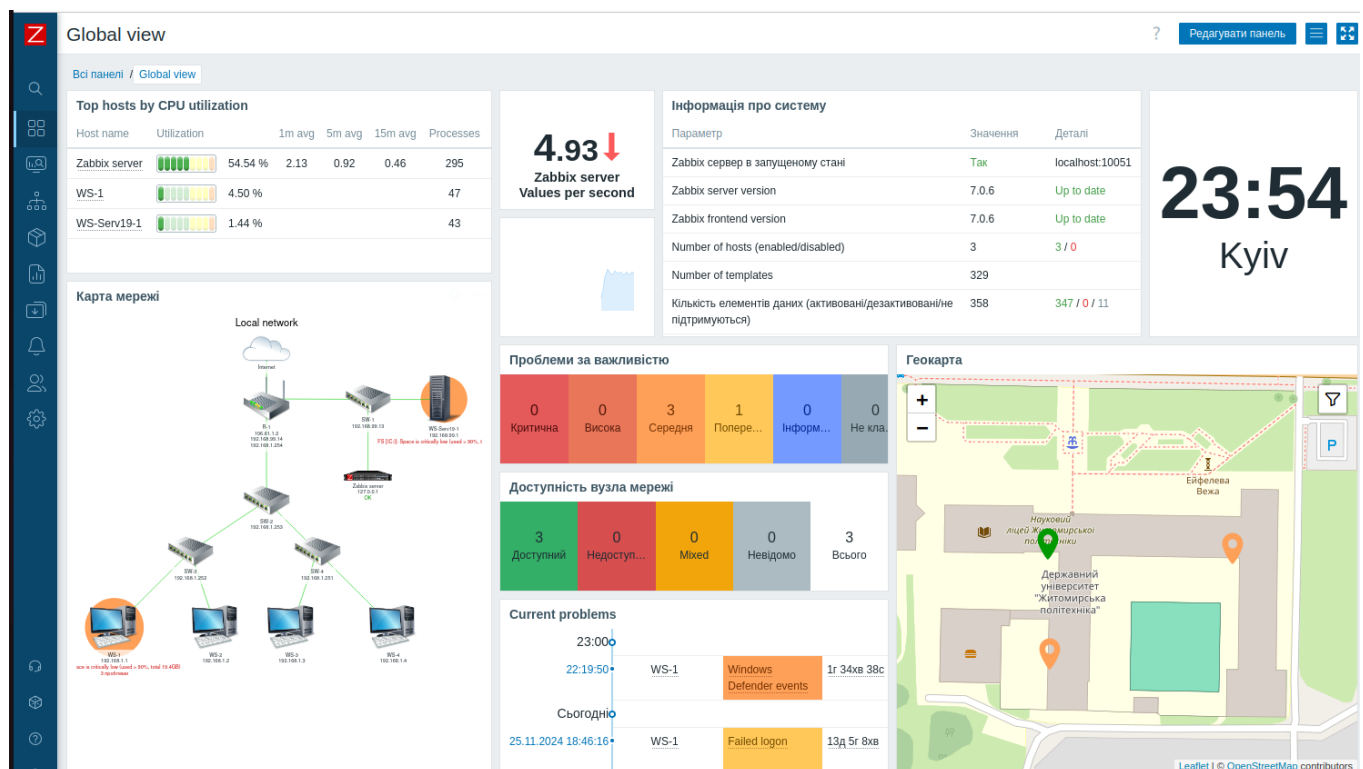


Рисунок 3.62 – Налагоджена панель моніторингу

Висновки до третього розділу

В ході розробки розділу було виконано базове налаштування мережі з використанням емулятора GNS3, що включало маршрутизатор, комутатор, Windows Server 2019 та кінцеві пристрої користувачів. Для безпеки на всіх пристроях ввімкнено роботу SSH і налаштовано базову аутентифікацію з використанням логіну та паролю. На сервері WS-Serv19-1 встановлено ролі Active Directory, DNS і DHCP, створено домен anchys.net. Робочі станції приєднано до домену, налаштовано динамічне отримання IP-адрес, перевірено зв'язок між вузлами мережі та віддалений доступ через SSH до мережевих пристроїв.

Встановлення Zabbix розпочалося з налаштування статичної адреси для станції Serv-Zabbix, після чого було виконано безпосереднє встановлення Zabbix.

Налаштування та перевірка роботи системи кібербезпекового моніторингу Zabbix включало встановлення агента на всі сервери та кінцеві станції. Після чого з відповідного меню на Zabbix була перевірена активність роботи агентів та виконано аналіз отриманих даних у вигляді метрик та графіків. Налагоджено шифрування передачі даних між агентами та сервером з використанням згенерованого ключа PSK. Було виконано налаштування сповіщення на основі подій з неправильним входом в систему та спрацюванням Windows Defender на ОС Windows. Налаштована карта мережі допоможе візуалізувати інфраструктуру і покращить видимість, а функція інвентаризації дозволить зберігати детальні відомості про кожен пристрій мережі. Налагоджена панель моніторингу під власні потреби дозволить швидше та ефективніше реагувати та виправляти отримані проблеми в мережі. Таким чином робота системи Zabbix була успішно протестована на предмет виявлення кібербезпекових загроз різного характеру.

ВИСНОВКИ

У першому розділі дипломної роботи було розглянуто теоретичні аспекти систем моніторингу, їхнє призначення та основні функціональні можливості. Проведено аналіз існуючих рішень, що дало змогу виявити переваги й недоліки популярних платформ моніторингу. Особливу увагу приділено системі кібербезпекового моніторингу Zabbix: її структурі, архітектурі, основним компонентам і функціональним можливостям. Дослідження дозволило визначити, що Zabbix є багатофункціональним рішенням яке забезпечує безпековий моніторинг мережевих пристроїв, серверів і додатків із високою гнучкістю у налаштуванні та можливістю інтеграції із зовнішніми системами.

В другому розділі було розроблено топологію мережі з використанням трирівневої ієрархічної моделі, виконано розподіл адресації між пристроями, а також обрано оптимальне обладнання для реалізації проєкту мережі. Для моніторингу мережевих ресурсів передбачено використання Zabbix із встановленими агентами на кінцевих пристроях. Обране мережеве обладнання та сервери відповідають сучасним вимогам до продуктивності, масштабованості та надійності, що гарантує стабільну роботу всієї мережі.

В третьому розділі було налаштовано базову інфраструктуру мережі з використанням GNS3, що включала маршрутизатор, комутатор, сервер Windows Server 2019 і кінцеві пристрої. Налаштовано безпечний доступ через SSH, створено домен на сервері, приєднано робочі станції та налаштовано отримання динамічної адресації на кінцевих пристроях. Встановлено та налаштовано систему моніторингу Zabbix, включно з агентами на всіх пристроях, перевірено їхню активність і налаштовано надсилення сповіщень про події. Створено тригери та налагоджено автоматичну відправку повідомлень про виявлені проблеми на електронну пошту адміністратора. Карта мережі й панель моніторингу Zabbix забезпечують зручну візуалізацію, ефективне виявлення й усунення проблем, а функція інвентаризації дозволяє зберігати детальну інформацію про пристрої.

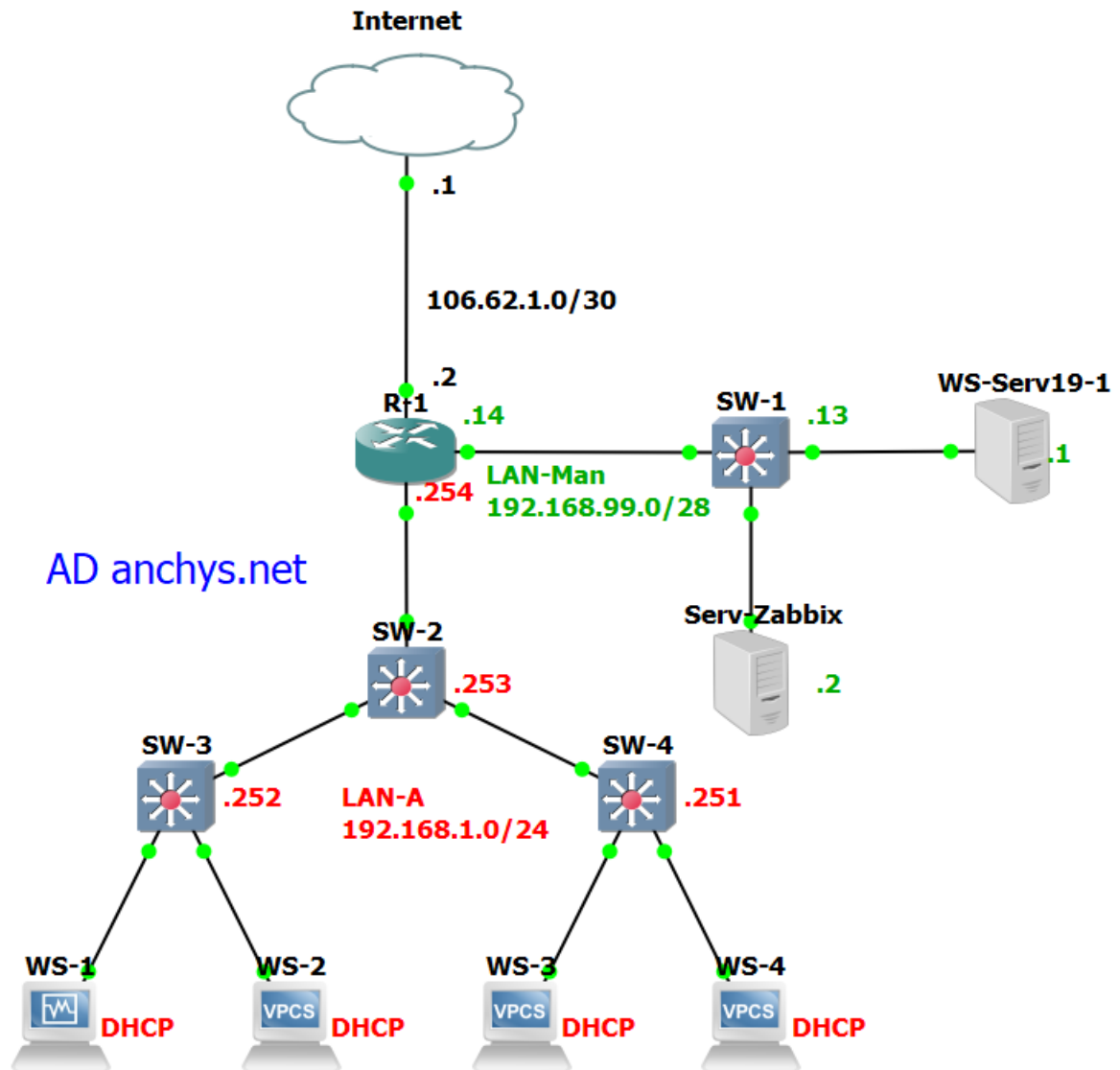
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Інструменти моніторингу та аналізу даних [Електронний ресурс]. Режим доступу до ресурсу: <https://learn.microsoft.com/uk-ua/power-platform/well-architected/security/monitor-threats>
2. Рекомендації щодо моніторингу та виявлення загроз [Електронний ресурс]. Режим доступу до ресурсу: <https://it-solutions.ua/servisi/sistemi-monitoringu-ta-keruvannya/>
3. Анчис А.О, Бродський Ю.Б. Технологія захисту кінцевих вузлів // Тези доповідей VI Всеукраїнської науково-технічної конференції «Комп'ютерні технології: інновації, проблеми, рішення», м. Житомир, 29-30 листопада 2023 року – Житомир: Житомирська політехніка, 2023.
4. Система моніторингу Zabbix [Електронний ресурс]. Режим доступу до ресурсу: <https://www.zabbix.com/>
5. Документація Zabbix [Електронний ресурс]. Режим доступу до ресурсу: https://www.zabbix.com/documentation/1.8/en/manual/about/overview_of_zabbix
6. Система моніторингу Nagios [Електронний ресурс]. Режим доступу до ресурсу: <https://www.nagios.org/>
7. Опис можливостей системи моніторингу Nagios [Електронний ресурс]. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Nagios>
8. Система моніторингу Prometheus [Електронний ресурс]. Режим доступу до ресурсу: <https://prometheus.io/docs/introduction/overview/>
9. Система моніторингу SolarWinds [Електронний ресурс]. Режим доступу до ресурсу: <https://www.solarwinds.com/>
10. Документація системи моніторингу SolarWinds [Електронний ресурс]. Режим доступу до ресурсу: https://documentation.solarwinds.com/en/success_center/orionplatform/content/core-how-orion-works-sw1625.htm

11. Система моніторингу Datadog [Електронний ресурс]. Режим доступу до ресурсу: <https://www.datadoghq.com/product/>
12. Опис системи моніторингу Datadog [Електронний ресурс]. Режим доступу до ресурсу: <https://proit.ua/datadog-rozshiriue-sfieru-diyi-ta-okhopliennia-platformi-spostieriezhiennia/>
13. Анчис А.О, Головня О.С. Структура та функціональні можливості системи моніторингу Zabbix // Тези доповідей VII Всеукраїнської науково-технічної конференції «Комп'ютерні технології: інновації, проблеми, рішення», м. Житомир, 02-03 грудня 2024 року – Житомир: Житомирська політехніка, 2024.
14. Огляд функцій Zabbix [Електронний ресурс]. Режим доступу до ресурсу: <https://www.zabbix.com/features>
15. Моніторинг серверів за допомогою Zabbix [Електронний ресурс]. Режим доступу до ресурсу: https://www.zabbix.com/server_monitoring
16. Моніторинг, пов'язаний із безпекою, за допомогою Zabbix [Електронний ресурс]. Режим доступу до ресурсу: <https://blog.zabbix.com/security-related-monitoring-with-zabbix/8659/>
17. Моніторинг мережі за допомогою Zabbix [Електронний ресурс]. Режим доступу до ресурсу: https://www.zabbix.com/network_monitoring
18. What is Hierarchical Network Design? [Електронний ресурс]. Режим доступу до ресурсу: <https://www.auvik.com/franklyit/blog/hierarchical-network-design/>
19. Cisco 4331 Integrated Services Router [Електронний ресурс]. Режим доступу до ресурсу: <https://www.cisco.com/c/en/us/support/routers/4331-integrated-services-router-isr/model.html>
20. Cisco Catalyst 2960-X Series Switches Router [Електронний ресурс]. Режим доступу до ресурсу: <https://www.cisco.com/c/en/us/support/switches/catalyst-2960-x-series-switches/series.html>

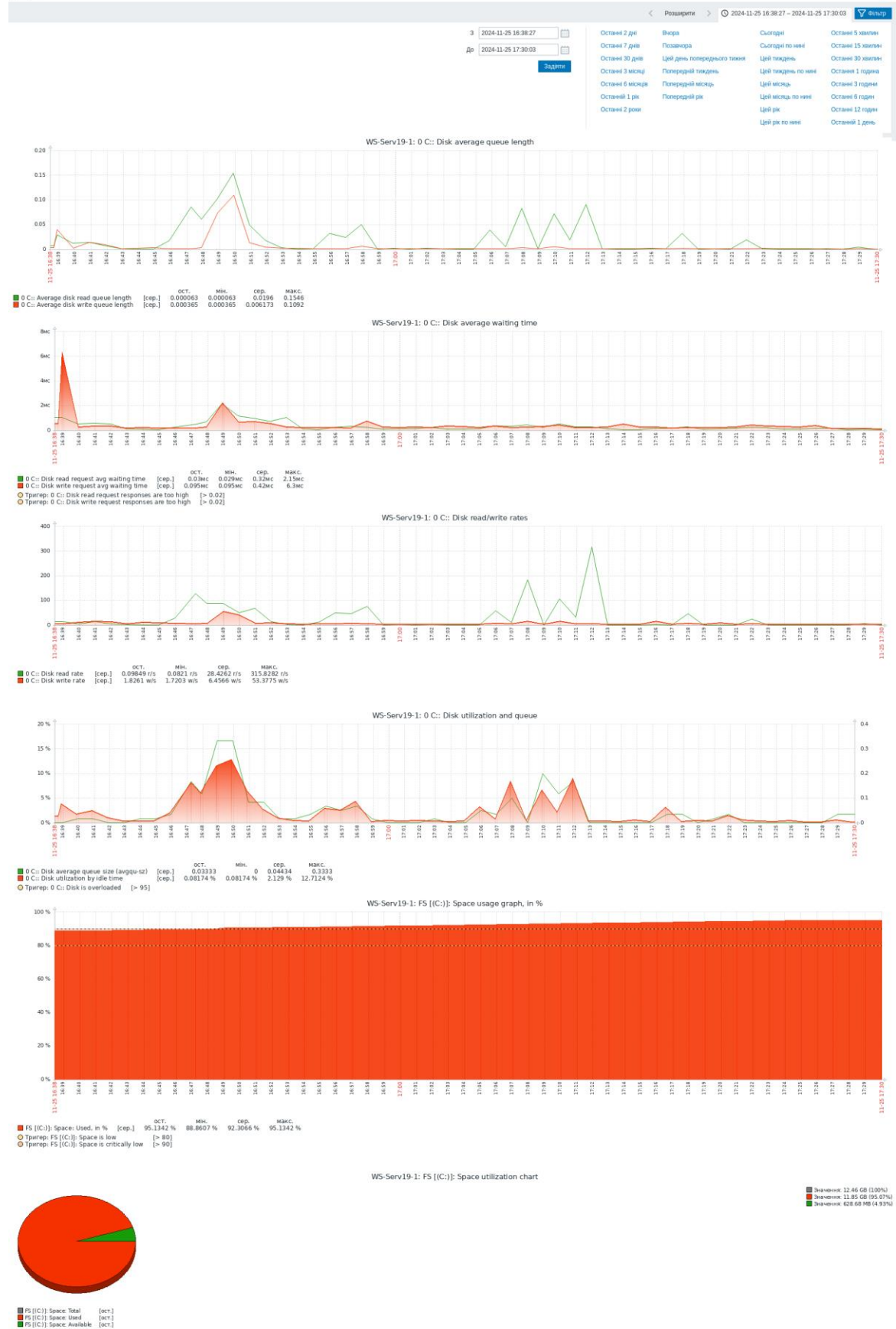
ДОДАТКИ

Додаток А – Схема проєкту мережі в емуляторі GNS3

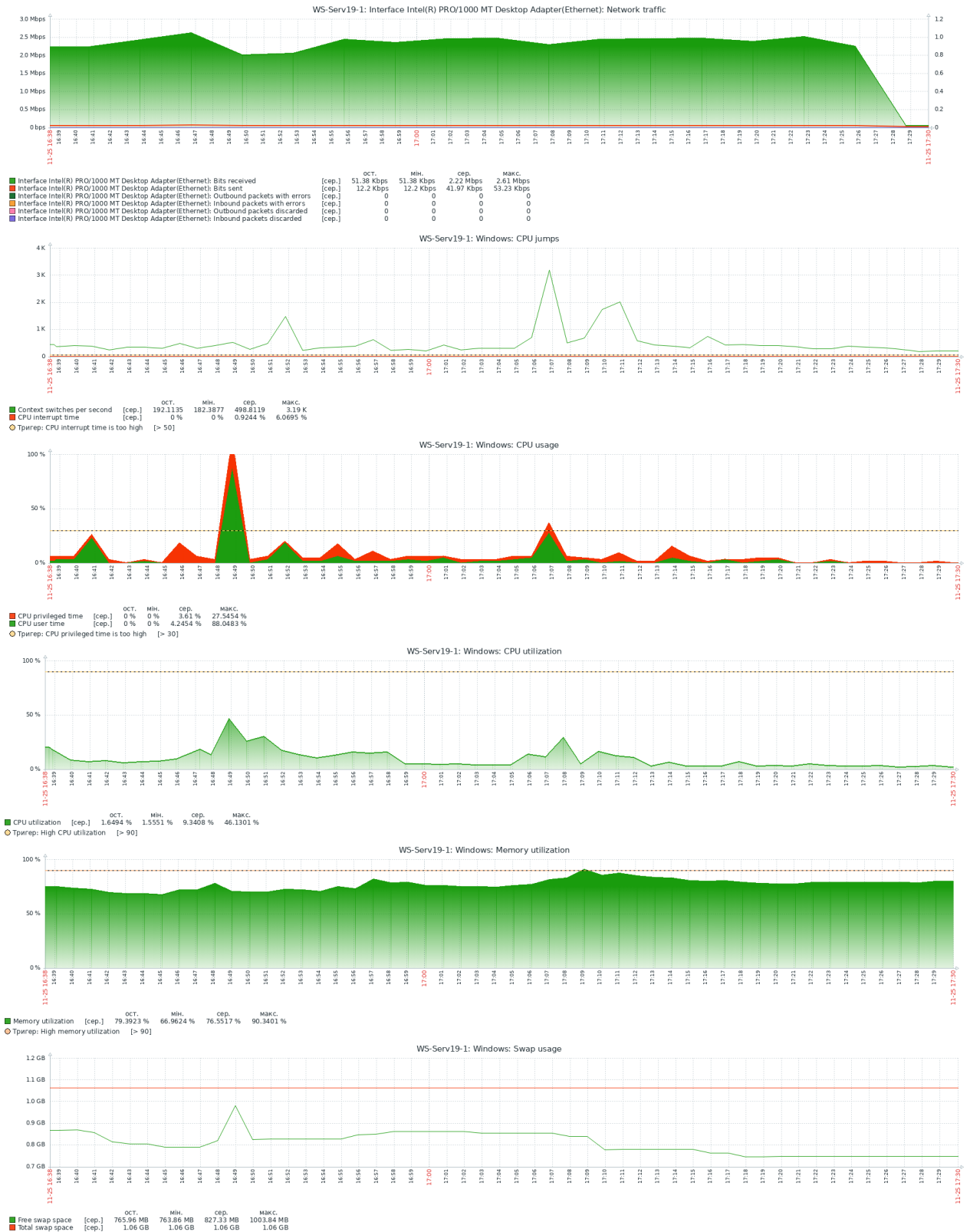


Додаток Б – Графік отриманих метрик з WS-Serv19-1

Графіки



Додаток В – Графік отриманих метрик з WS-Serv19-1



R-1:

```

upgrade fpd auto
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R-1
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
enable secret 5
$1$FU/5$KCdSH8xHb694NCT1Sq2Og/
!
no aaa new-model
ip source-route
no ip icmp rate-limit unreachable
ip cef
!
!
ip domain name anchys.net
ip name-server 192.168.99.1
ip name-server 8.8.8.8
no ipv6 cef
!
multilink bundle-name authenticated
!
username admin privilege 15 secret 5
$1$NsY0$Wc87K5Guo/NNANQ6A9VtO/
archive
 log config
  hidekeys
!
!
ip tcp synwait-time 5
ip ssh time-out 30
ip ssh version 2
!
interface FastEthernet0/0
 description link-to-internet
 ip address dhcp
 ip nat outside
 ip virtual-reassembly
 duplex half
!
interface FastEthernet1/0

```

```

 description link-to-sw-1
 ip address 192.168.99.14 255.255.255.240
 ip nat inside
 ip virtual-reassembly
 duplex auto
 speed auto
!
interface FastEthernet1/1
 description link-to-lan
 ip address 192.168.1.254 255.255.255.0
 ip helper-address 192.168.99.1
 ip nat inside
 ip virtual-reassembly
 duplex auto
 speed auto
!
ip forward-protocol nd
no ip http server
no ip http secure-server
!
ip dns server
ip nat inside source list 1 interface
FastEthernet0/0 overload
!
access-list 1 permit 192.168.0.0 0.0.255.255
no cdp log mismatch duplex
control-plane
!
gatekeeper
 shutdown
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 login local
 stopbits 1
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 stopbits 1
line vty 0 4
 login local
 transport input ssh
!
End

```

SW-2:

```
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
service compress-config
!
hostname SW-2
!
boot-start-marker
boot-end-marker
!
!
logging discriminator EXCESS severity drops
6 msg-body drops EXCESSCOLL
logging buffered 50000
logging console discriminator EXCESS
enable secret 5
$1$epRd$kbw.A7/4XGRQePxZA7ZWd/
!
username admin privilege 15 secret 5
$1$pHzU$GN7LDEnWtORrD.7UTufuQ/
no aaa new-model
!
!
no ip routing
no ip icmp rate-limit unreachable
!
!
ip domain-name anchys.net
ip name-server 192.168.1.254
ip name-server 192.168.99.1
no ip cef
no ipv6 cef
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
!
interface Ethernet0/0
!
interface Ethernet0/1
!
...
interface Ethernet3/2
!
interface Ethernet3/3
!
interface Vlan1
```

```
ip address 192.168.1.253 255.255.255.0
no ip route-cache
!
ip default-gateway 192.168.1.254
ip forward-protocol nd
!
ip tcp synwait-time 5
ip http server
!
ip ssh time-out 30
ip ssh version 2
ip ssh server algorithm encryption aes128-ctr
aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr
aes192-ctr aes256-ctr
!
!
control-plane
!
!
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
login local
line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
line vty 0 4
login local
transport input ssh
!
!
!
end
```