МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
Факультет кібербезпеки, комп'ютерної та програмної інженерії
Кафедра комп'ютерних інформаційних технологій

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри
_____Аліна САВЧЕНКО
"___"_____2021 р.

ДИПЛОМНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ

«МАГІСТРА»

ЗА ОСВІТНЬО-ПРОФЕСІЙНОЮ ПРОГРАМОЮ «ІНФОРМАЦІЙНІ УПРАВЛЯЮЧІ СИСТЕМИ ТА ТЕХНОЛОГІЇ»

Тема: «Саморозгортувана віртуальна машина агент-серверної телеметрії ентерпрайз класу»

Виконавець: Орлов Ілля Олексійович

Керівник: професор Зіатдінов Юрій Кашафович

Нормоконтролер: _____ Ігор РАЙЧЕВ

Київ 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра Комп'ютерних інформаційних технологій

Галузь знань, спеціальність, освітньо-професійна програма: 12 "Інформаційні технології", 122 "Комп'ютерні науки", "Інформаційні управляючі системи та технології"

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Аліна САВЧЕНКО

«____»_____2021р.

ЗАВДАННЯ

на виконання дипломної роботи студента

Орлова Іллі Олексійовича

1. Тема роботи: "Саморозгортувана віртуальна машина агент-серверної телеметрії ентерпрайз класу"

Затверджена наказом ректора №1891/ст. від 02.10.2021р..

2. Термін виконання роботи: з 05.10.2021р. до 21.12.2021р.

3. Вихідні данні до роботи: агент серверна телеметрія, створення працездатної віртуальної машини.

4. Зміст пояснювальної записки: 1)Проаналізовано потребу телеметрії. 2) Створено віртуальну машину. 3)Проаналізовано найпопулярніші системі моніторингу.

5. Перелік обов'язкового ілюстративного матеріалу: рисунки, діаграма, а також слайди презентації доповіді у PowerPoint.

6. Календарний план-графік

№ з/п	Завдання	Термін виконання	Підпис керівника
1.	Формування теми дипломної роботи, постановка задачі та узгодження з дипломним керівником.	05.10.21	
2	Формування структури розділів дипломної роботи	06.10.21– 10.10.21	
3	Формування та оформлення першої частини дипломної роботи	11.10.21 – 15.11.21	
4	Збір науково-технічного матеріалу до другої частини дипломної роботи	16.10.21 – 20.10.21	
5	Формування та оформлення другої частини дипломної роботи	08.11.21 – 16.11.21	
6	Розробка програмної частини згідно з задачу дипломної роботи	16.11.21 – 31.11.21	
7	Формування та оформлення третьої частини дипломної роботи	19.11.21 – 31.11.21	
8	Формування звіту та графічних матеріалів	23.11.21 – 08.12.21	
9	Підписання необхідних документів	09.12.21 – 15.12.21	
10.	Підготовка до захисту дипломної роботи	09.12.21 – 21.12.21	

7. Дата видачі завдання: 05.10.2021р.

Керівник дипломної роботи _____ Юрій ЗІАТДІНОВ
(підпис керівника)

Завдання прийняв до виконання _____ Ілля ОРЛОВ
(підпис випускника)

РЕФЕРАТ

Пояснювальна записка до дипломної роботи "Саморозгортувана віртуальна машина агент-серверної телеметрії ентерпрайз класу " містить 83 сторінки, 43 рисунка, 4 таблиці, 6 використаних джерел.

Ключові слова – МОНІТОРИНГ ТЕЛЕМЕТРІЯ, АВТОМАТИЗАЦІЯ РОБОТИ ІТ ВІДДІЛУ, СТВОРЕННЯ ВІРТУАЛЬНОЇ МАШИНИ, НАЛАШТУВАНЯ СИСТЕМИ.

Мета дипломної роботи – використання новаційних методів, для досягнення економії ресурсів ІТ-відділу в компаніях.

Об’єкт дослідження – середні та великі ІТ компанії.

Предмет – інтеграція віртуальної машини серверного програмного забезпечення.

Метод дослідження – аналіз та моніторинг ресурсів при обслуговуванні серверного обладнання.

Результат роботи – диск віртуальної машини з налаштуваннями серверної системи.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

PCM – процес перетворення аналогового сигналу на цифровий сигнал.

Bash – вдосконалена варіація командної оболонки Bourne shell.

Troubleshooting – форма вирішення проблем, для ремонту несправних продуктів чи процесів машини або системи.

Traceroute – програма для визначення маршрутів слідування даних.

Active Directory – інтелектуальна служба каталогів корпорації Microsoft.

Ping – службова комп'ютерна програма, що використовується для перевірки з'єднань.

ВСТУП

Бурхливе зростання комп'ютерних мереж призводить до частих збоїв у їхній працездатності. Тому потрібно покращувати існуючі засоби контролю за функціонуванням локальних обчислювальних мереж.

Особливо гостро постає це задачу, коли спочатку при проєктуванні мережі було закладено менші розміри та навантаження. Ця ситуація призводить до неможливості користувачів отримати своєчасний доступ до потрібної інформації.

Крім цього, необхідність моніторингу виникає, коли в рамках однієї мережі об'єднується кілька підмереж, спочатку спроєктованих для вирішення різних завдань. У цьому випадку в мережі збільшується обсяг трафіку, а також зростає навантаження на сервери додатків і особливо на сервери доменів.

Все вищесказане підкреслює особливу актуальність вирішення задачу постійного контролю над функціонуванням мережі для гарантованої (передбачуваної) роботи мережі загалом і доступність користувачів її компонент. Подібний контроль забезпечує збереження цілісності та доступності даних, допомагає запобігти несанкціонованому доступу до них.

РОЗДІЛ 1
ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Телеметрія

1.1.1 Понятя телеметрії

Телеметрією називається область науки і техніки, що займається питаннями розробки та експлуатації комплексу автоматизованих засобів, що забезпечують отримання, перетворення, передачу по каналу зв'язку, прийом, обробку та реєстрацію вимірювальної інформації та інформації про події з метою контролю на відстані стану та функціонування технічних та біологічних систем різних об'єктів та вивчення явищ природи [4].

Вужче телеметрію визначають як вимір на відстані фізичних величин, що характеризують технологічний процес, явище природи, стан живого організму.

Телеметрія дозволяє задовольнити дуже важливу потребу вченого, інженера, медичного експерта або іншого користувача даних про віддалені об'єкти.

Одне з важливих застосування телеметрії можна назвати льотні випробування нової моделі літака або іншого літального апарату. Для оцінки працездатності конструкції та льотних характеристик літака потрібно вимірювати витрати пального, характеристики роботи двигунів, механічні навантаження, що випробовуються фюзеляжем та крилами, вібрації та температури критично важливих елементів літального апарату, параметри електронного обладнання літака, траєкторні дані. Засоби телеметрії стежать за вимірами у безлічі точок, кількість яких становить від кількох сотень до кількох

тисяч, і надають результати вимірів конструкторам на їх комп'ютери або дисплейні термінали.

Найскладніші сучасні системи телеметрії застосовують у аерокосмічних дослідженнях. До часто застосовуваних датчиків відносяться датчики (перетворювачі) тиску і витрати, генераторні датчики, термопари, термометри опору, мости та потенціометри. Для сигнальних параметрів характерна стрибкоподібна зміна в часі, наприклад, пов'язана з переходом з одного дискретного стану в інший - до них відносяться сигнали: включено-вимкнено, так-ні і т.д.

1.1.2 Історія створення

Передача інформації з проводів бере свій початок у XIX столітті. Борис Семенович Якобі (Моріц Герман Якобі) (1801–1874 р.р.), російський учений, винахідник та експериментатор створив телеграфний апарат і з його допомогою організував лінію зв'язку між Зимовим палацом російського імператора та Царським селом (1839 р.), а в 1845 року між Зимовим палацом і Головним штабом.

У 1839 році в Росії пущена в експлуатацію лінія семафорного телеграфу між Санкт-Петербургом і Варшавою, що стала надалі однією з найдовших у світі (1200 км). Сигнал нею через 149 проміжних станцій йшов 15 хв. З використанням електричного телеграфу дію семафорного телеграфа 1854 р. припинено.

24 травня 1844 року Семюель Морзе і Альфред Вейл закінчили будівництво першої експериментальної телеграфної лінії між Вашингтоном і Балтімором (США). Морзе надіслав перше повідомлення.

Антоніо Меучі (1808-1896 р.р.) винайшов у 1849 році телефонний зв'язок. Через відсутність коштів на реєстрацію не зміг запатентувати пристрій. Організував телефонний зв'язок у власному будинку для того, щоб хвора дружина могла викликати його, коли їй ставало погано. Провів демонстрацію (1860 р.) пристрою передачі голосу по проводах, під час якої голос співака

передавався по телефонним проводам на відстань кількох миль. Опублікував малюнки свого винаходу в 1870, на 6 років раніше Белла (1876).

У 1874 році французькі інженери встановили систему датчиків визначення погоди та глибини снігу на Монблані для передачі інформації в режимі реального часу до Парижа.

У 1876 Олександр Грейм Белл (1847-1922 р.р.) запропонував свій винахід Телеграфної компанії («Telegraph Company). Із висновків експертів: «…Ми встановили, що голос дуже слабкий і неясний… Технічно, ми не бачимо (перспектив), що цей пристрій буде колись здатний до посилки розбірливої мови на відстань кількох миль».

1.1.3 Галузі використання

Сільське господарство.

Від своєчасного надання даних залежить благополучний стан сільськогосподарських культур та отримання хороших урожаїв, якісне зберігання овочів у овочесховищах.

Для керування зберіганням овочів потрібно контролювати кілька параметрів: контроль температури, вологості, кратність повітрообміну (контроль вуглекислого газу) та циркуляцію повітря в камері зберігання.

Для моніторингу захворювань рослин та для пропорційного зрошення бездротові метеостанції відіграють важливу роль, вони передають на базову станцію інформацію про важливі параметри, необхідні для прийняття рішень. Основні параметри, що вимірюються: температура і відносна вологість повітря, випадання опадів і вологість листя (для побудови моделей профілактики захворювань), сонячна радіація, швидкість вітру (для розрахунку випаровування), зволоженість грунту (для прийняття рішень про зрошення). Оскільки місцеві мікроклімати можуть суттєво відрізнятися, таку інформацію потрібно отримувати прямо від сільськогосподарських культур. Зазвичай станції моніторингу використовують сонячні батареї для забезпечення

енергонезалежності станцій від місцевої інфраструктури і передають дані, використовуючи наземне радіо або супутникові системи.

Водопостачання та водовідведення.

Телеметрія застосовується для оцінки якості води та вимірювання показників потоку: в автоматичних водолічильниках, моніторингу підводних вод, визначенні витоків у розподільчих трубопроводах. Дані виходять майже реальному часі і дозволяють негайно реагувати на обставини.

Медицина.

Телеметрія (біотелеметрія) використовується для спостереження за пацієнтами, які перебувають під загрозою виникнення патологічної серцевої діяльності, які переважно перебувають у кардіологічних диспансерах. До таких пацієнтів підключаються вимірювальні, записувальні та передавальні пристрої. Зареєстровані дані можуть бути використані лікарями у діагностиці стану пацієнта. Завдяки функціям сигналу тривоги медичні сестри можуть бути сповіщені у разі виникнення різких загострень або небезпечних станів для пацієнта.

Оборона та космос, ракетна техніка.

У ракетній техніці телеметричне обладнання стає невід'ємною частиною обладнання ракет, що використовуються при спостереженні за процесом ракетного запуску, для отримання інформації про параметри зовнішнього середовища (температури, прискорень, вібрацій), про енергопостачання, точне вирівнювання антени та (на довгих дистанціях, наприклад, при космічному польоті) про час розповсюдження сигналу.

Керування польотом космічного апарату здійснюється автоматизованою системою керування, основним задачум якої є керування орієнтацією космічного апарату та рухом його центру мас. Для цього необхідні системи передачі командно-програмної інформації на космічний апарат та телеметричної інформації від нього.

Такі космічні агенції як NASA, ESA та інші використовують телеметричні, телекеровані системи для збору даних із діючих космічних апаратів та супутників.

Авто- та мотоспорт.

Телеметрія є ключовим фактором у сучасному автоспорті. Інженери можуть обробляти величезну кількість даних, що збираються в ході пробного заїзду та використовувати їх для відповідної модернізації автомобіля та досягненні при цьому оптимальних властивостей. Системи, що використовуються в серіях гонок Формула-1, дозволяють вирахувати можливий час проходження кола, що є потрібною інформацією для пілота. Інші приклади необхідних вимірювань включають прискорення (сили тяжіння) по трьох осях, графіки температур, швидкість обертання коліс та усунення підвіски. У Формулі-1 також записуються дії пілота, що дозволяє команді оцінити його продуктивність і за нещасного випадку Міжнародна автомобільна федерація може визначити або виключити роль помилки пілота як можливий випадок.

У Формулі-1 двоколійна телеметрія з'явилася на початку 90-х років (TAG-electronics) і реалізовувалась через дисплей повідомлень на щитку, повідомлення на якому команда могла оновлювати. Його розвиток тривало до травня 2001 року, коли вперше було отримано дозвіл встановлювати цю систему на автомобілях. З 2002 року команди вже могли змінювати режими роботи двигуна і відключати окремі моторні датчики з піт-стопів, коли машина знаходилася на трасі. Починаючи з сезону 2003 року двоколійна телеметрія була заборонена на Формулі-1, проте дана технологія все ще продовжує існувати і зрештою знаходить своє застосування в інших видах гоночних або дорожніх автомобілів.

Енергетика.

На фабриках, будівлях та в будинках проводиться спостереження в багатьох місцях за енергоспоживанням таких систем як клімат-контроль разом із пов'язаними параметрами (наприклад, температурою) за допомогою бездротової телеметрії на одну центральну точку. Інформація збирається та обробляється,

дозволяючи приймати найрозумніші рішення щодо ефективних шляхів використання енергії. Такі системи дозволяють здійснювати профілактичне технічне обслуговування.

Дослідження дикої природи.

Телеметрія використовується вивчення дикої природи, зокрема спостереження за видами, що під загрозою на індивідуальному рівні. Піддослідні тварини можуть бути оснащені інструментарієм, починаючи від простих бірок і закінчуючи камерами, пакетами GPS та передавачами для забезпечення інформацією вчених та керівників.

Телеметрія використовується в гідроакустичних оцінках риби, які традиційно використовуються під час мобільного обстеження з човнів для оцінки біомаси риб та просторового розподілу. І навпаки, є технічне обладнання, що розміщується в стаціонарних місцях, воно використовує стаціонарні перетворювачі для контролю проходження риби.

Оцінки проходження риби проводяться 24 години на добу протягом року, визначається швидкість проходження риби, її розмір, просторовий та тимчасовий розподіл.

В останні 35 років у всьому світі використовуються десятки тисяч мобільних або стаціонарних апаратів гідроакустичної оцінки для оцінок переміщень риби в завихрення водяної течії та вивчення міграцій риб у річках.

Розумні будинки.

За допомогою високотехнологічних пристроїв для зручності проживання людей створюються розумні будинки, тобто системи, що вміють розпізнавати конкретні ситуації, що відбуваються в будівлі, і відповідним чином на них реагувати (одна із систем може керувати поведінкою інших за виробленими алгоритмами).

В єдину систему розумного будинку зазвичай входять: система опалення, вентиляції та кондиціювання, охоронно-пожежна сигналізація, система

контролю доступу в приміщення, контроль протікання води, витоків газу, система відеоспостереження, система освітлення, механізація будівлі (відкриття/закриття воріт, шлагбаумів, електропідігрів) щаблів тощо), керування з одного місця аудіо-, відеотехнікою, домашнім кінотеатром.

Центри обробки даних (ЦОД).

Центр обробки даних є спеціалізованою будівлею для розміщення серверного та комунікаційного обладнання та підключення до каналів мережі Інтернет, виконує функції обробки, зберігання та розповсюдження інформації.

ЦОД зазвичай складається з інформаційної інфраструктури, що включає серверне обладнання і забезпечує обробку і зберігання інформації, телекомунікаційної інфраструктури, що забезпечує взаємозв'язок елементів ЦОД і передачу даних між ЦОД і користувачами, інженерної інфраструктури, що забезпечує нормальне функціонування основних систем центру обробки даних.

Інженерна інфраструктура включає: кондиціювання для підтримки температури і рівня вологості в заданих параметрах, охоронно-пожежну сигналізацію та система газового пожежогасіння, системи віддаленого контролю.

Автомобілі.

Телеметричні системи дозволяють визначати склад паливної суміші для підвищення ккд двигуна та екологічності вихлопу, склад вихлопних газів, температуру контрольних точок двигуна (олії, охолоджуючої рідини), тиск масла в двигуні, обороти двигуна, тиск у гідроканалах гальмівної системи та зчеплення, тиск у шинах великовантажних машин, здійснювати вібродіагностику, контроль складу олії на наявність металевої стружки, контроль клімату.

Також телеметричні системи використовуються при бурінні похилих та горизонтальних свердловин, в системах глобального позиціонування, системах безпеки (сигналізація, відеоспостереження).

Телеметрією називається область науки і техніки, що займається питаннями розробки та експлуатації комплексу автоматизованих засобів, що забезпечують отримання, перетворення, передачу по каналу зв'язку, прийом,

обробку та реєстрацію вимірювальної інформації та інформації про події з метою контролю на відстані стану та функціонування технічних та біологічних систем різних об'єктів. та вивчення явищ природи [4].

Вужче телеметрію визначають як вимір на відстані фізичних величин, що характеризують технологічний процес, явище природи, стан живого організму.

Радіотелеметрія своєю чергою є телеметрією, що використовує радіоканали зв'язку.

Телеметрична система є сукупність пристроїв, що забезпечують збирання сигналів із засобів первинного перетворення, формування телеметричних сигналів, передачу їх по каналу зв'язку, реєстрацію та відображення телеметричних повідомлень на приймальній стороні [4].

Передається інформація включає інформацію про результати вимірювання фізичних параметрів, стану контрольованих об'єктів, явищах, що вивчаються або події, а також інформацію, що забезпечує роботу наземних засобів телеметричної системи. Інформація, що надходить з об'єктів телеконтролю, згідно [5], поділяється на такі групи:

- інформація про стан систем та агрегатів контрольованого об'єкта, а також про роботу різної апаратури;
- інформація про параметри навколишнього простору;
- інформація про медико-біологічні параметри людини та тварин.

До складу представлених груп входять різноманітні телеметровані параметри (показники фізичного процесу, події або явища, що вивчається, значення або поведінка якого підлягають вимірюванню або контролю).

Параметри, що телеметруються, залежно від характеру зміни в часі діляться на функціональні та сигнальні. Функціональні параметри є безперервними функціями часу і, своєю чергою, поділяються на повільно мінливі (спектр частот від нуля до 20 – 50 Гц) і мінливі (спектр частот до 2…3 кГц і більше). Для сигнальних параметрів характерна стрибкоподібна зміна часу.

До параметрів, що повільно змінюються, відносяться температури, тиску, швидкості потоків рідин і газів, лінійні і кутові переміщення, швидкості

прискорення і т.д. Вібрації та акустичні шуми є представниками параметрів, що швидко змінюються. Прикладом сигнальних параметрів служить сигнал увімкнено - вимкнено.

Сукупність пристроїв та (або) складових частин з одним входом та одним виходом, що забезпечують передачу групових телеметричних сигналів на відстань та їх прийом називається каналом зв'язку [4].

Лінія зв'язку, на відміну від каналу, може обслуговувати кілька джерел повідомлень, утворюючи багатоканальну лінію. Сучасні телеметричні системи та комплекси, будучи багатоканальними, забезпечують одночасну передачу великої кількості вимірюваних величин на одній несучій частоті.

Нижче наведено основні показники телеметричних систем.

Пропускна здатність - максимально можливий обсяг повідомлень, який телеметрична система здатна передавати, приймати та реєструвати в одиницю часу [4].

Дальність зв'язку характеризує максимальну відстань, у якому забезпечується необхідна можливість зв'язку (можливість отримання протягом сеансу зв'язку телеметричних повідомлень).

Похибка телеметрування – це відхилення оцінки параметра, що телеметрується, або будь-якої його характеристики від справжнього значення [4].

Вимоги, що пред'являються при проєктуванні телеметричної системи, та особливості її побудови залежать від призначення та умов застосування цієї системи. Наприклад, якщо телеметрична система входить до складу системи телекерування як інформаційна ланка, вирішальну роль грає швидкість та достовірність отримання даних телевимірювань. У разі передачі на великі відстані найважливішого значення набуває енергетична ефективність радіолінії, визначається витратами енергії однією двійкову одиницю інформації.

Загальні вимоги до телеметричних систем:

- можливість одночасної передачі великої кількості різноманітних параметрів (від кількох десятків до кількох сотень);

- забезпечення заданої точності, яка оцінюється для систем середньої, високої та дуже високої точності середньоквадратичними похибками 3…5 %, 1…2 % та 0,1…0,5 % відповідно;
- можливість оперативного зміни складу вимірюваних параметрів з урахуванням їх найважливіших особливостей (швидкості виміру часу, швидкості передачі та інших.);
- високий рівень автоматизації процесів збору, передачі та обробки даних;
- забезпечення високої надійності телеметричної апаратури та низки інших експлуатаційних вимог (мала вага, обсяг, вартість у розрахунку на один вимір або одну двійкову одиницю інформації) [5]. Системи рухомого моніторингу об'єктів дозволяють контролювати переміщення будь-яких об'єктів, що рухаються, як транспортних засобів, так і людей.

Основним задачум моніторингу є контроль у режимі реального часу розташування об'єкта та маршруту його руху. Система моніторингу дозволяє зберігати маршрути руху об'єкта, створювати звіти про рух об'єкта, його швидкість, просте, про технічний стан транспортного засобу за допомогою аналогового підключення до датчиків автомобіля. Існує можливість створення маршруту руху та контролю його проходження.

На рухомому об'єкті розміщується мобільний навігаційний контролер із приймачем GPS, gsm/gprs приймачем та різними датчиками. Приймачі GPS приймають сигнали з видимих супутників. Потім інформація про географічне розташування об'єкта, точний час, дані з датчиків передаються в центр керування (web-server+ PC зі спеціалізованим програмним забезпеченням) по gsm каналу. Центр керування приймає та обробляє ці дані та відображає інформацію про положення кожного рухомого об'єкта на карті в реальному часі. Центр керування може надсилати команди на мобільний навігаційний контролер, наприклад, включати звуковий сигнал, зупиняти двигун, змінювати напрямок руху, доставляти повідомлення і т.д.

Перевагою використання системи моніторингу є не лише можливість контролювати переміщення транспортного засобу та його стан, а й значно оптимізувати витрати на його експлуатацію, витрати на керування автопарком загалом.

Моніторинг може використовуватися як протиугінна система, і як система пошуку автомобіля у разі його угону.

Моніторинг приватних осіб дозволяє контролювати місцезнаходження дітей, осіб похилого віку, а також працівників, які мають роз'їзний характер роботи. Ефективна система моніторингу рухомих об'єктів та пошуку домашніх тварин.

- Сфери застосування системи моніторингу:
- Корпоративний транспорт
- Муніципальний транспорт
- Таксі
- Авіаційний транспорт
- Пожежні служби
- Рятівні бригади
- Інкасатори
- Приватні особи
- Приватний транспорт
- Рідкісні та дорогі тварини

Бездротові мережі можуть бути використані в різних галузях народного господарства: у медицині – передача інформації від машин швидкої допомоги до шпиталю, у правоохоронних органах – стеження за рухомими об'єктами з можливістю документування подій; організації зв'язку при стихійних лихах, дистанційного контролю над об'єктами (передача телевізійного сигналу, передача кодів керування різне устаткування - наприклад на камери лімба і об'єкта під час відстеження траєкторії польоту ракет-носіїв у космонавтиці);

встановлення різноманітних датчиків (зокрема мобільних - наприклад, спеціальні браслети для дітей), змонтованих у загальну систему оповіщення;

Апаратна реалізація заснована на використанні бездротових систем передачі інформації, так звані mesh-мережі, тобто осередкові мережі бездротової передачі даних, що самоорганізуються, смуга пропускання в яких може забезпечувати гарантовану якість каналу і високу швидкість передачі.

У мережах радіопередач використовуються як вузькоспрямовані антени, так і антени з ширшим сектором охоплення, аж до всеспрямованих (кругових). Для з'єднання типу точка-точка використовуються дві націлені одна на одну (вузько) спрямовані антени; так будуються, наприклад, радіорелейні лінії передач, у яких відстань між сусідніми релейними вежами може обчислюватися десятками кілометрів. Вузконаправлена антена фокусує радіопромінь, збільшуючи щільність його енергії; таким чином передавач даної потужності "прострілює" на більшу відстань.

Інший тип зв'язку вийде при використанні лише всеспрямованих антен. У цьому випадку буде досягнуто можливості з'єднання кожного з кожним. Таку топологію зазвичай мають невеликі установчі мережі, розгорнуті на обмеженій території.

Нарешті, якщо в центрі "осередку" помістити базову станцію (БС) з всенаправленою антеною і забезпечити всіх обслуговуваних нею абонентів сфокусованими на неї спрямованими антенами, то отримаємо топологію "точка-багатоточка". Якщо ще з'єднати між собою базові станції в деякій ієрархії (або радіорелейними лініями або просто радіо-з'єднаннями на кшталт "крапка-крапка", або кабельними каналами), то отримаємо вже цілу стільникову мережу.

За цим принципом будуються системи бездротового широкосмугового доступу (БШД). У центрі зони обслуговування встановлюється БС із секторними антенами, але в віддалених майданчиках - абонентські термінали з спрямованими антенами (рис.2). Для роботи сервісів реального часу, передачі трафіку голосу та відео сучасні системи БШД підтримують якість обслуговування з виділенням гарантованої смуги пропускання.

Пропускна здатність такої мережі, що розподіляється між абонентськими терміналами, що обслуговуються одним сектором БС, залежатиме від числа терміналів. Ефективна продуктивність одного сектора БС відомих виробників БШД лежить у межах 10-43 Мбіт/с, чого цілком достатньо для організації більшості інфокомунікаційних сервісів.

1.2 Моніторинг системи

1.2.1 Необхідність моніторинг системи

Інформаційні системи ускладняються завдяки розвитку IT, а саме: накопичення все більшої кількості інформації та вдосконалення збору та аналізу алгоритми.

Пандемія викликала революційні зміни в інфраструктурі ділового світу і багато компаній все ще намагаються адаптуватися. Стрімкі зміни обставин вимагають швидких дій, якщо ми хочемо, щоб компанії вижили в нинішній ситуації. Ті, хто мав можливість перемістити свої операції в режим онлайн, щоб продовжувати вести бізнес, вважаються щасливчиками, але вони були обмежені в часі і наявних ресурсах, щоб залишитися на плаву. Отже, одним із найбільш затребуваних інструментів у віддалених компаніях, безумовно, є програмне забезпечення для моніторингу співробітників. Незважаючи на те, що на ринку сьогодні їх багато, їх характеристики, опції, можливості інтеграції та цінові діапазони дуже різняться, тому важливо провести ретельне дослідження, перш ніж приймати на себе зобов'язання щодо довгострокового виконання одного рішення.

Ефективне керування проєктами є дуже важливою частиною встановлення стандартизованого рівня продуктивності праці працівників в організації. Інструмент керування проєктами з відстеженням часу може бути особливо корисним, коли ваша компанія жонглює кількома важливими проєктами

одночасно. Наприклад, маркетингові агентства чи адвокатські контори мають безліч клієнтів, із якими працюють і котрим використовують різні ресурси. Щоб мати можливість стежити за часом, що витрачається на кожен сегмент, та успішно дотримуватися графіка, таким компаніям, як вони, було б розумно впроваджувати сучасні технології, які надають точну та актуальну інформацію, необхідну для завершення проєктів у встановлені терміни та з очікуваними результатами. Складні проєкти, як правило, розбиваються на безліч завдань для кращого розуміння прогресу та причинно-наслідкового зв'язку між конкретними задачуми, але це також може ускладнити їх виконання у хронологічному порядку. Саме тут програмне забезпечення для відстеження проєктів має важливе значення, так як воно відображає всі складні деталі індивідуально та у великому масштабі та дозволяє малі та великі зрушення у стратегії середнього проєкту. Більше того, такі інструменти дозволяють оптимізувати процеси на всіх рівнях, що особливо важливо за часів, коли стабільність компанії ставиться під сумнів як внутрішніми, так і зовнішніми силами. Якщо компанія прагне або заявляє про свою гнучкість у веденні бізнесу, наявність коштів для оптимізації операційних процесів є обов'язковою.

Комплексний захист комп'ютерної мережі сучасного рівня потребує використання різноманітних засобів безпеки, таких як системи виявлення мережевих атак, системи захисту від спаму, антивіруси, міжмережеві екрани (firewall), сканери безпеки тощо. При цьому зростання кількості апаратних та програмних засобів захисту мережі значно збільшує обсяг аналізованої інформації, необхідний контролю безпеки. Як наслідок - адміністратори мережі повинні приділяти значний час аналізу рутинної інформації, що знижує продуктивність роботи та, відповідно, впливає на оперативне прийняття рішень щодо підтримки функціонування комп'ютерної мережі. Таким чином, виникає суперечність між збільшенням обсягу інформації, яку доцільно аналізувати для запобігання загрозам, та оперативністю керування мережею.

Під загрозою з точки зору безпеки слід розуміти сукупність умов і факторів, що потенційно призводять до порушення функціонування

комп'ютерної мережі в цілому, у тому числі контрольованих мереж активів (даних), а також окремих користувачів [49]. Загрози зазвичай поділяються на: навмисні (усвідомлене заподіяння шкоди) та природні. До перших відносяться несанкціоновані підключення, витоку інформації, порушення функціонування мережі тощо, до других – форс-мажорні обставини та нещасні випадки, а також помилки внаслідок збоїв апаратури [41]. При цьому значна частина загроз безпеці є наслідком людського фактора – відсутності у користувачів необхідних компетенцій, а також ігнорування службових інструкцій (наприклад, недбале поводження з паролями). Аналіз безпеки комп'ютерної мережі вимагає враховувати всі види загроз, проте якщо природні загрози досить легко формалізуються в плані ризиків і захист від них досить лінійний, то загрози, що мають причину людський фактор, вимагають особливої уваги через непередбачуваність дій навіть за відсутності наміру заподіяння шкоди. З іншого боку, людський фактор має значення у тих випадках, коли дисфункція системи будь-яким чином загрожує людині, що особливо важливо з футуристичного погляду – час, коли робототехніка стане звичною у побуті, відноситься до найближчого майбутнього [46].

1.2.2 Характеристики систем моніторингу

Основним задачум системи моніторингу є надання актуальної інформації для аналізу стану IT-інфраструктури та швидкого виявлення виниклої несправності та її оперативне усунення. Системи моніторингу продуктивності дозволяють IT-фахівцям вчасно помітити зниження продуктивності та визначити "вузькі місця" в IT-інфраструктурі. Постійний моніторинг допомагає уникнути простоїв у її роботі, підтримувати всі IT-сервіси в робочому стані та зберігати необхідний рівень їхньої якості, а також спланувати її модернізацію.

Раніше роль моніторингу здійснювали адміністратори, а інформація про стан систем у кращому разі збиралася ними ж у будь-яких неспеціалізованих програмах (через їхню відсутність), у гіршому ж взагалі ніяк не накопичувалася

і не агрегувалась. Усі відомості про систему були прив'язані до практичного досвіду роботи з інфраструктурою у конкретного спеціаліста та повністю губилися під час його відходу. Зараз з'явилося безліч напів- та повністю автоматизованих систем для моніторингу, які аналізують стан систем, збирають інформацію в колекції, які теж згодом можна вивчити за потрібності.

Існують досить специфічні види моніторингу, наприклад, від імені кінцевого користувача, коли задані проміжки часу циклічно емулюються його дії. Зазвичай це робот, планувальник завдань, що запускає спеціальний, заздалегідь визначений скрипт-сценарій, а потім рапортує про успіх виконання дій або про помилки. Для зберігання отриманої інформації зазвичай використовується конфігураційна база даних під різними СУБД: інформація про об'єкти моніторингу представлена як набір конфігураційних одиниць.

Моніторинг локальної мережі є безперервним процесом, пов'язаним із спостереженням за робочою мережею. Процес виконує такі функції:

Своєчасне виявлення помилок та несправностей.

Адекватна та швидка реакція на помилки та несправності.

Здійснює моніторинг стану мережі системний адміністратор. Для зручності роботи застосовують різноманітні програмні засоби оповіщення. Одним з таких програм є наступна розробка - Total Network Monitor від Softinventive Lab.

Системи моніторингу

Виділяються основні вимоги, які мають бути у софті з моніторингу мережі:

Підтримка всіх видів мережевих підключень, у тому числі Wi-Fi мережі.

Стеження за мережевою активністю.

Визначення детальності системних та мережевих служб.

Аналіз віддалених комп'ютерів та веб-серверів.

Системи моніторингу мають надавати звіти про події за певні періоди часу. Важливо зберігати весь лістинг активності та архівувати його у відповідному журналі.

Потрібно розрізняти засоби, які забезпечують контроль зовнішнього доступу мережі та програмного забезпечення, що важливо для контролю над мережевими процесами.

Моніторинг активності мережі визначається так:

Програма з певним періодом надсилає запити на необхідні ір адреси мережі.

При некоректному чи невдалому результаті такого запиту, відправляється сигнал сисадміну.

Автоматичне визначення дії, що регламентовані мережевим протоколом.

Методи моніторингу

Існує безліч методів та засобів для моніторингу мережевих підключень. Особливості їх використання залежить від цілей процесу, мережевої конфігурації, файлової системи тощо.

Основні методи:

Аналізатори протоколів. Дані системи необхідні виключно для контролю мережевого трафіку.

Інтегровані системи керування та аналізу. Використовують для програмного та апаратного середовища. Забезпечують контроль певних програм, відрізків комунікацій та окремих пристроїв у мережі.

Керування мережею. Сюди відноситься ПЗ, яке збирає дані про мережеві процеси та стан апаратного блоку. Відстежується весь мережевий трафік.

Кабельне встаткування. Здійснює сертифікацію та тестування кабельних мереж.

Програма Total Network Monitor зараз належить до найбільш актуальних програм моніторингу робочої мережі. Софт забезпечує своєчасне відстеження неполадок, перевіряє програмне забезпечення на актуальність і працює з антивірусними базами.

Кожен сервер кожен мережевий пристрій це певна одиниця, все це зберігається в централізованій базі даних. Таке уявлення дозволяє потім

інтегрувати систему моніторингу з візуальними уявленнями: діаграмами, графіками та ін.

Сама структура моніторингу значно видозмінюється з часом. Наприклад, одна з тонкощів виникла при появі та великому поширенні віртуалізації: якщо раніше була необхідність відстежувати стан лише фізичних серверів, то тепер на кожному з них може бути ще кілька віртуальних. Також системи моніторингу можна налаштувати виконання будь-яких стандартних сервісних дій. Наприклад, очищати кошик при заповненні або активувати архівування для будь-яких файлів, коли певний відсоток дискового простору стає зайнятим.

При виборі, розробці, впровадженні систем моніторингу спочатку потрібно визначитися з об'єктами, які будуть підлягати спостереженню, а також критичні події та показники, які визначать кількість оповіщень при поломці, частоту сканування та інші параметри та наслідки.

Для великих інфраструктур, на кшталт дата-центрів, перед фінальним використанням зазвичай розгортають тестовий майданчик, де можна оцінити доцільність зроблених рішень та визначень параметрів порогових значень. Впровадження подібних рішень особливо важливо при використанні сервісного підходу до діяльності IT-підрозділів, коли всі процеси переглядаються з точки зору IT-сервісів, що надаються підрозділом.

Кожен бізнес-сервіс корпоративної системи, по можливості, інтерпретується як IT-сервіс, задається певний рівень якості його надання. Далі він описується у системі моніторингу як набір взаємозалежних компонентів IT-інфраструктури.

Системи моніторингу можуть бути спрямовані на споживачів різного рівня. Для великих систем зазвичай використовується величезна кількість різноманітних функцій, для маленьких зазвичай досить загального аналізу вузлів та надсилання оповіщень. Серед основних функцій моніторингів можна виділити такі:

Спостереження. Основна функція, що включає періодичний збір показників з вузлів обладнання, сервісів і т.п.

Зберігання інформації. Доповнення до стеження. Здійснюється збір інформації за основними показниками кожного об'єкта моніторингу, для зберігання зазвичай використовуються бази даних.

Побудова звітів. Здійснюється як на основі поточних даних стеження, так і за довгостроковою інформацією. Наприклад, довготривалий моніторинг навантаження на сервер може попередити, що ресурси постійно збільшуються, отже потрібно збільшити доступні засоби або перенести частину завдань на інший сервер, вибір якого також можна здійснити на основі довготривалого звіту.

Візуалізація. Звіти у візуальному поданні: у вигляді графіків, підказок, діаграм. Допомагають легкому сприйняттю інформації, а також можливий вибір для відображення декількох найважливіших індикаторів, тоді як у звітах будуть представлені всі показники.

Пошук вузьких місць. На основі аналітичних даних моніторингу можна дізнатися, в якому місці інфраструктури найбільше знижує загальні показники продуктивності.

Автоматизація сценаріїв. Функція звільняє адміністраторів від рутинних завдань.

Завдяки наявності коштів для реалізації всіх цих функцій адміністратору більше не потрібно перевіряти вручну стан кожної складової системи, проблеми вирішуються та уламки усуваються оперативніше, діагностика здійснюється багатомірно і точно, а також можна планувати розширення інфраструктури.

Використання систем моніторингу та керування дозволяє:

- оптимізувати використання інформаційних ресурсів;
- підвищити якість IT-сервісів та швидкість усунення збоїв у роботі обладнання та програмного забезпечення, мінімізувати час простою сервісів;
- забезпечити надійність, безпеку та узгоджене функціонування всіх компонентів IT-інфраструктури;
- полегшити модернізацію IT-інфраструктури;
- у кілька разів підвищити ефективність роботи IT-підрозділу.

РОЗДІЛ 2
ВИБІР ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Найпопулярніші системи моніторингу

Всі системи моніторингу поділяються на кілька груп - платні і безкоштовні, масштабовані і не масштабовані, що працюють по одному-двох протоколів моніторингу або здатні забезпечувати безліч видів моніторингу. Отже, ми маємо дві платні і дві безкоштовні системи моніторингу. Розглянемо необхідні функції для нашого підприємства:

надійність: PRTG, Zabbix, Nagios;

моніторинг мережевої активності пристрою: всі системи;

перевірка стану пристрою (вентилятор, CPU, пам'ять): всі системи;

видалене керування пристроями: PRTG, Zabbix, Nagios;

автоматичні дії при тривогах: Zabbix, Nagios;

безкоштовність: Zabbix, NetXMS; PRTG, Zabbix.

Нашій компанії не хочеться платити зайві гроші, отже, ми вибираємо Opensource систему. Як видно з прикладу на рисунку 2.1 , на середньому підприємстві є більше 300 пристроїв, а в даному випадку, відстежується майже 14000 елементів даних (або сенсорів), що, безсумнівно, заощаджує нашому підприємству 16 900 $, якщо не купувати ліцензію PRTG.

За функціями всі системи однакові, але ми хочемо встановити систему з можливістю у майбутньому її розширити та вдосконалити. Невідомо, яку апаратуру чи пристрій потрібно підключити та яку інформацію збирати. Необхідна надійна та стабільна система, за якою немає потрібності стежити щодня, тому NetXMS нам не підходить.

Порівнюючи можливості систем моніторингу, їх продуктивність, складність налаштування та вартість, можна дійти висновку, що найбільш підходящою системою є Zabbix.

У таблиці 2.1 наведу невелику порівняльну таблицю існуючих на сьогоднішній день систем моніторингу.

Таблиця 2.1

Існуючі на сьогоднішній день системи моніторингу

Платні	
PRTG	Рішення для моніторингу великих корпоративних мереж. Країна виробник - Німеччина.
OSG (Orange)	Рішення для обслуговування великих корпоративних мереж. Чудова в питаннях інтеграції під технічне завдання замовника. Країна виробник -Росія.
Solar Winds	Рішення для обслуговування великих корпоративних мереж. Країна виробник - США.
SCOM	Рішення для обслуговування серверної інфраструктури Microsoft Windows Server. Країна виробник - США.
Wats UP Gold	Рішення для моніторингу невеликих мереж. Має ряд недоліків в порівнянні з іншими платними аналогами. Країна виробник - США.
Cisco Prime	ПО для обслуговування мереж, побудованих на устаткуванні Cisco. Країна виробник - США.
SNMPc	Вузькоспеціалізоване рішення для моніторингу устаткування по протоколах SNMP і ICMP. Країна виробник -Росія (Agneco SNMPc), СШФ (SNMPc Network Manager).
Безкоштовні (вільно поширювані)	
Zabbix	Універсальна, стабільна, масштабована, широкий функціонал, можливість створення своїх інструментів. Країна виробник -Латвія.
Prometheus	Універсальна, стабільна, має спеціалізовану базу даних, здатну витримати високі навантаження. Відкритий проект.
Nagios + Cacti	Своєрідна система, що запускається і підтримується в ручному режимі.
10 Strike	Обмежений функціонал, demo- версія. Розширений функціонал - у платному режимі. Підходить для невеликих мереж. Країна виробник - Росія.
Ping	Універсальний інструмент перевірки доступності мережевих вузлів, вбудований у будь-яку ОС. Підходить для вирішення украй вузького круга завдань.

2.2 Переваги та можливості Zabbix

Zabbix – система, що вільно розповсюджується для комплексного моніторингу мережевого обладнання, серверів і сервісів. Складається з чотирьох частин: Сервер моніторингу (ядро) – виконує періодичне опитування та отримання даних, обробляє їх, аналізує, також здійснює запуск скриптів для розсилки оповіщень. Може віддалено перевіряти мережеві послуги, є сховищем, у якому зберігаються всі конфігураційні, статистичні та оперативні дані. Не може розміщуватися на сервері під керуванням операційної системи сімейства Windows, а також OpenBSD.

Проксі - збирає дані про продуктивність та доступність від імені Zabbix сервера. Всі зібрані дані заносяться в буфер на локальному рівні і передаються серверу Zabbix, до якого належить проксисервер. Zabbix проксі є ідеальним рішенням для централізованого віддаленого моніторингу місць, філій, мереж, які не мають локальних адміністраторів. Він може бути також використаний для розподілу навантаження одного сервера Zabbix. У цьому випадку, проксі лише збирає дані, тим самим на сервер лягає менше навантаження на ЦПУ та на введення/виведення диска.

Агент – спеціальний демон, який запускається на об'єктах, що відстежуються, і надає дані серверу, здійснюючи контроль локальних ресурсів та додатків (таких як жорсткі диски, пам'ять, статистика процесора тощо) на мережевих системах, тобто. ці системи повинні працювати з запущеним Zabbix агентом (проте моніторинг можна проводити не тільки за допомогою нього, але й за SNMP версій 1, 2, 3, запуском зовнішніх скриптів, що видають дані, та кілька видів зумовлених вбудованих перевірок, таких як ping, запит по http, ssh, ftp та інших протоколів, а також замір часу відповіді цих сервісів Zabbix агенти є надзвичайно ефективними через використання вбудованих системних викликів для збору інформації про статистику. на AIX та Windows.

Концепція Zabbix, що відображає багатозадачність і універсальність системи, зображена на малюнку 2.1.

Рис. 2.1. Концепція Zabbix як активної системи моніторингу

За допомогою Zabbix можна здійснювати розподілений моніторинг до 1000 вузлів, де конфігурація молодших вузлів контролюється старшими в ієрархії. Також продукт включає централізований моніторинг лог-файлів, можливість створювати карти мереж (вручну за шаблоном), виконання запитів до різних баз даних, генерацію звітів та тенденцій, виконання сценаріїв на основі моніторингу, підтримку інтелектуального інтерфейсу керування платформами (IPMI).

Zabbix надає гнучкі можливості з налаштування умов-тригерів, які включаються при аваріях та неполадках, і система починає моргати лампочками (насправді червоними квадратиками), сповіщаючи адміністратора про можливу поломку. Також, при включенні тригера, веб-інтерфейс навіть починає попискувати на зразок будильника, Zabbix досить самостійний і зможе відправити повідомлення на пошту, в jabber або sms за допомогою gsm-модему, або навіть спробувати самостійно підняти сервіс, що впав, виконавши заздалегідь певні дії, які запускаються під час спрацьовування певних тригерів.

За умови відхилення одержуваних параметрів від встановлених еталонних значень (при спрацьовуванні тригерів) - спрацьовують інструменти оповіщення адміністратора системи моніторингу або сконструйовані адміністратором скрипти. Оповіщення може бути візуальним, звуковим, у вигляді електронного листа - передбачено кілька різних видів реакції на зафіксовані події (Рис. 2.2):

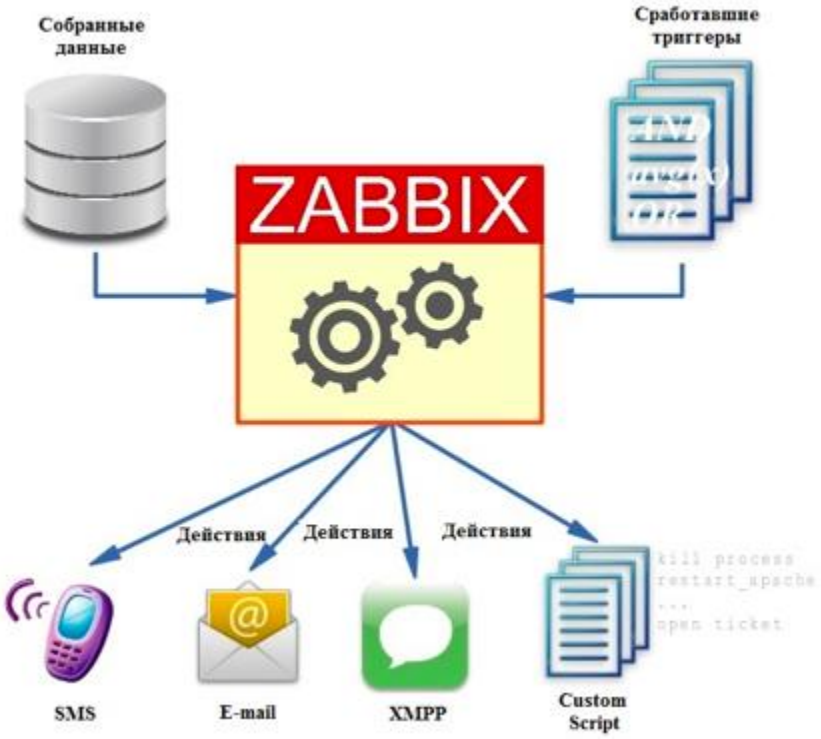

Рис. 2.2. Варіанти реакції системи на зафіксовані події

Очевидно, що система надає масу можливостей для моніторингу самих різних процесів, що відбуваються в інформаційних мережах. Розглянемо технології моніторингу, які підтримує Zabbix.

Класичний моніторинг мережевих пристроїв здійснюється за протоколами ICMP і SNMP.

Протокол ICMP потрібен для простої перевірки досутпності мережевого вузла, з доступом додаткового аналізу відсотка втрачених ICMP-пакетів і часу затримки. Протокол є основним елементом будь-якого моніторингу, і використовується повсюдно.

Протокол SNMP - створений для моніторингу та керування мережевими пристроями. Нас буде цікавити виключно моніторинг, оскільки Zabbix не є системою керування. Всі сучасні мережеві пристрої без винятку підтримують протокол SNMP. На мережному диску існують (запрограмовані при виробництві) бази об'єктів SNMP, або так звані MIB-бази (management information base). У MIB-базах містяться як статичні об'єкти, які не змінюються в процесі роботи пристрою (інвентарні номери, моделі і назви пристроїв), так і динамічні, які з плином часу змінюють свої значення (завантаження трафіком мережевого інтерфейсу, температура і завантаження процесора, рівні помилок , і багато іншого). Виробники розміщують в мережі Інтернет документацію на MIB своїх мережевих пристроїв - по суті, це структуровані текстові файли або групи файлів, які містять інформацію про всі SNMP-параметрах пристрою, до якого вони належать, і їх можливі значення. Кожен об'єкт в MIB має свій унікальний цифровий адресу OID (Object IDentificator) і ім'я Object Name. Система моніторингу звертається до інших мережних пристроїв по протоколу SNMP і запитує цифрові значення конкретних OID. Зрозуміло, потрібно знати, який OID в MIB-базі зберігає в собі ту чи іншу значення. Для цього потрібно вміти працювати з MIB-файлами (і документацією виробника), аналізуючи які, можна отримати необхідні для вирішення поставлених завдань OID. Протокол SNMP представляє велику кількість можливостей для отримання корисної інформації при опитуванні мережевих пристроїв. Наприклад, під час опитування джерел безперебійного живлення (якщо вони мають мережеві інтерфейси) - можливо отримати дані про температурнйи стан термодатчика і значення напруги в мережі електроживлення. Під час опитування комутаторів, можна отримувати таку специфічну інформацію, як рівень помилок на мережевому інтерфейсі, а при роботі з маршрутизаторами - отримувати статистику IP SLA тестів (технологія моніторингу якості послуг зв'язку). Важливим та незамінним при моніторингу мережевого обладнання є протокол SNMP .

Доступність мережевих сервісів здійснюється за допомогою відкриття TCP-сесій на порт тестованого мережевого сервісу (мова йде про так званих безагентних перевірках, коли на стороні тестованого сервісу немає додаткового ПЗ, і аналізується лише можливість встановлення TCP-з'єднання). Для розгорнутого моніторингу серверів ОС Windows і Linux існує технологія так званих агентів, коли на об'єкт моніторингу (сервер) встановлюється додаткове ПО (Zabbix-агент), яке і відправляє серверу моніторингу Zabbix необхідні дані.

На малюнку 2.3 приведена схема навчального стенду і об'єктів, моніторинг яких буде організований і вивчений в рамках даного курсу:

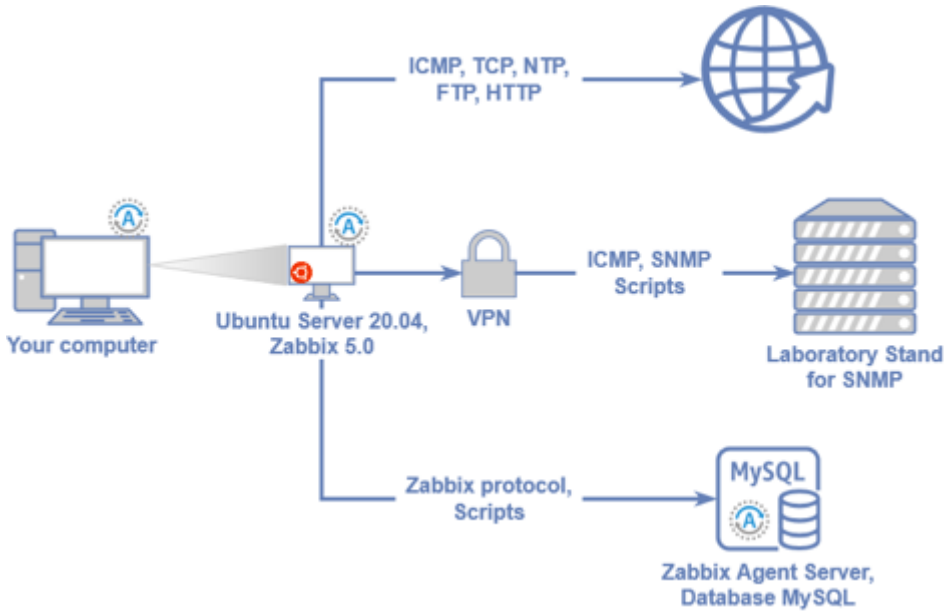

Рис. 2.3. Схема навчального стенду технології моніторингу

Потрібно пам'ятати, що використання будь-який, навіть самої просунутої системи моніторингу при обслуговуванні мережі не вирішує всіх задач, що стоять перед адміністратором.

Автоматичне виявлення:

- автоматичне виявлення по діапазону IP-адрес, доступним сервісам та SNMP перевірка;
- автоматичний моніторинг виявлених пристроїв;
- автоматичне видалення відсутніх хостів;

розподіл за групами і шаблонами залежно від результату, що

У запасі у Zabbix є ще півдесятка функцій, які дозволяють ще більше спростити спостереження за мережею, такі як моніторинг стану веб-сайту за допомогою автоматичного виконання сценарію на кшталт імітації дії користувача на сайті. У результаті це одна з найпотужніших і найширших систем моніторингу. У результаті ми отримуємо найбільш підходящу для наших цілей систему, яку також можна використовувати як скелет до своїх власних скриптів моніторингу.

Однак у черговий раз варто відзначити громіздкість сервісу, відсутність повної документованості можливостей проєкту, а також необхідність встановлення агентського програмного забезпечення на всі машини.

Як додатковий мінус варто відзначити складність делегування прав - машина з сервісом часто управляється операційною системою сімейства *nix, що робить трудомісткою взаємодію з доменними користувачами та правами з Active Directory (Windows системи).

Далі перерахуємо інструменти, присутні в Zabbix і деяких платних системах моніторингу, але не увійшли в роботу через екзотичність, складності реалізації саме в Zabbix, або неможливості організувати лабораторні умови для їх вивчення. Сюди відносяться веб-моніторинг по протоколу HTTP (аналіз веб-сторінки на наявність певного текстового контенту), виконання віддалених команд по протоколам Telnet і SSH, зі збереженням результатів виконання цих команд в базу даних Zabbix, моніторинг серверів за технологією IPMI, Java-додатків, і платформ VmWare.

РОЗДІЛ 3
РЕАЛІЗАЦІЯ ПРОЄКТУ

3.1. Налаштування системи

3.1.1. Встановлення Ubuntu

Zabbix складається з:

- сервера моніторингу, який виконує періодичне отримання даних, обробку, аналіз та запуск скриптів оповіщення;
- бази даних (MySQL, PostgreSQL, SQLite чи Oracle);
- веб-інтерфейсу на PHP;
- агента-демона, який запускається на об'єктах, що відстежуються і надає дані серверу.

Агент опціональний, моніторинг можна проводити не тільки за допомогою нього, але й за SNMP (версій 1, 2, 3), запуском зовнішніх скриптів, що видають дані, та кілька видів зумовлених вбудованих перевірок, таких як ping, запит на http, ssh, ftp та іншим протоколам, а також замір часу відповіді цих сервісів.

У Zabbix є три способи встановлення:

а) Установка з пакетів – Zabbix постачає офіційні RPM та DEB пакети для Red Hat Enterprise Linux, Debian та Ubuntu LTS. Також доступні репозиторії yum та apt.

б) Завантаження архіву з вихідними кодами та самостійне їх складання.

в) Завантаження готового рішення - повністю встановлена і налаштована система на віртуальному диску або настановний образ CD.

Готове рішення Zabbix доступне у таких форматах:

- vmdk (VMware/Virtualbox);
- OVF (Open Virtualisation Format);
- KVM;
- 23HDD/flash image, USB stick;
- Live CD/DVD;
- Xen guest;
- Microsoft VHD (Azure);
- Microsoft VHD (Hyper-V).

Найшвидше і просте рішення - встановити готове рішення, але це передбачає встановлення зайвих компонентів, які нам не потрібні. Ми ж хочемо знати, що встановлюємо. Тому встановлення з пакетів ми також відмовимося. Будемо встановлювати з вихідних кодів, контролюючи кожен крок установки.

Базу даних, виходячи з міркування поширеності та простоти використання, візьмемо MySQL.

Операційна система для нашого сервера Zabbix не відіграє особливої ролі, тому візьмемо останню стабільну версію Ubuntu, доступну на сайті http://ubuntu.ru/get: Ubuntu 14.04.5 server 32 bit. 32-хбитную систему візьмемо тому, як для Zabbix не потрібно об'єму RAM пам'яті більше 4Gb.

Достоїнства віртуалізації сервера – економія місця у стійках, зниження енергоспоживання та тепловиділення, спрощене адміністрування та широкі можливості з автоматизації розгортання та керування серверами.

Для почтаку роботи потрібно встановити на свій ПК з офіційного сайту virtualbox.org актуальну версію середовища віртуалізації VirtualBox. Аналогічно дистрибутив Ubuntu Server 20.04 LTS 64 bit (Рис.3.1), з офіційного ресурсу проєкту Ubuntu - http://ubuntu.com. Працездатність при використанні інших ресурсвів не досліджена, тому не може гарантуватися.

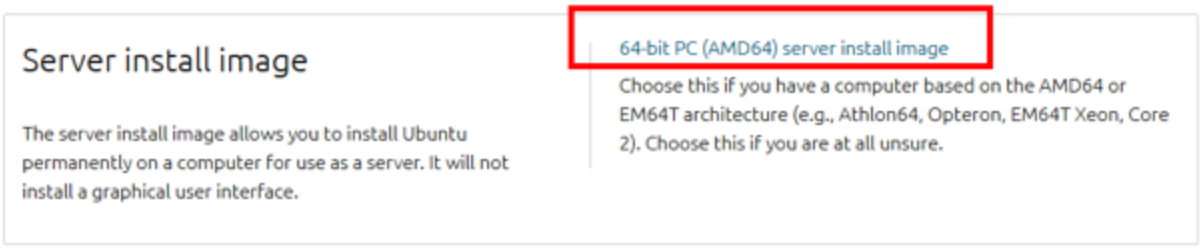

Рис. 3.1. Вибір дистрибутива ОС Linux Ubuntu

На офіційному сайті проєкту, zabbix.com, в керівництві по установці системи наведені чіткі відповідності версій Zabbix дистрибутивам ОС сімейства Linux, які можна використовувати. Для створення даного проєкту була обрана зв'язка ОС Ubuntu Server 20.04 + Zabbix 5.0, оскільки дистрибутив Ubuntu Server зарекомендував себе як надійна серверна платформа з широким базовим функціоналом, а Zabbix 5.0 - стабільний пакет актуальної на сьогоднішній день версії.

Приступимо до створення віртуальної машини. Запустив VirtualBox, і створив нову віртуальну машину (Рис. 3.2).

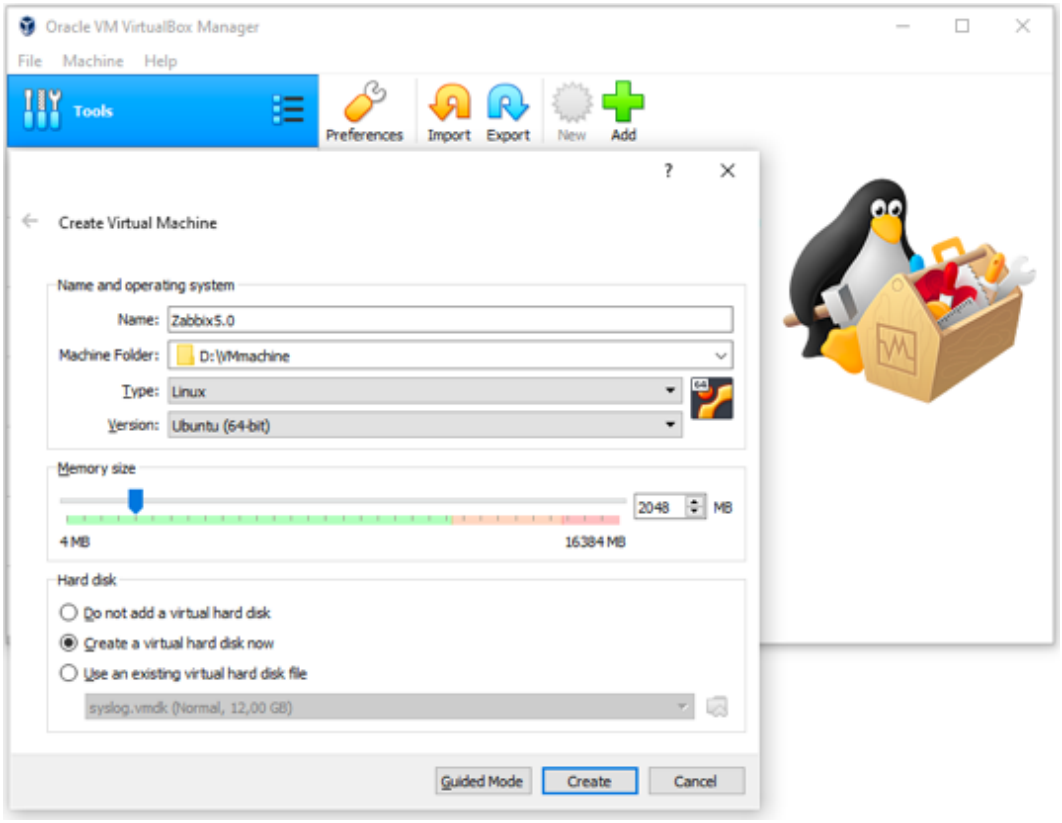

Рис. 3.2. Створення віртуальної машини

Потрібно вказати об'єм ОП, яку буде використовувати віртуальна машина. В основному використовувати системі ресурси будуть база даних, яка обробляє всі запити і впорядковує одержувані системою моніторингу відомості, і сам Zabbix, які генерує запити до віддалених об'єктів і передає зібрані дані в базу даних.

Важливим є надання віртуальній машині достатній обсяг оперативної пам'яті, інакше система при високих навантаженнях не буде працювати.

Відповідно до рекмендації Zabbix, для спостереження за мережевими вузлами в кількості 500 одиниць, досить 2 ГБ оперативної пам'яті.

Далі потрібно створити новий віртуальнийжорсткий диск.

У роботі було обрано тип жорсткого диска VMDK, з фіксованим розміром, обсягом 10 ГБ (Рис. 3.3).

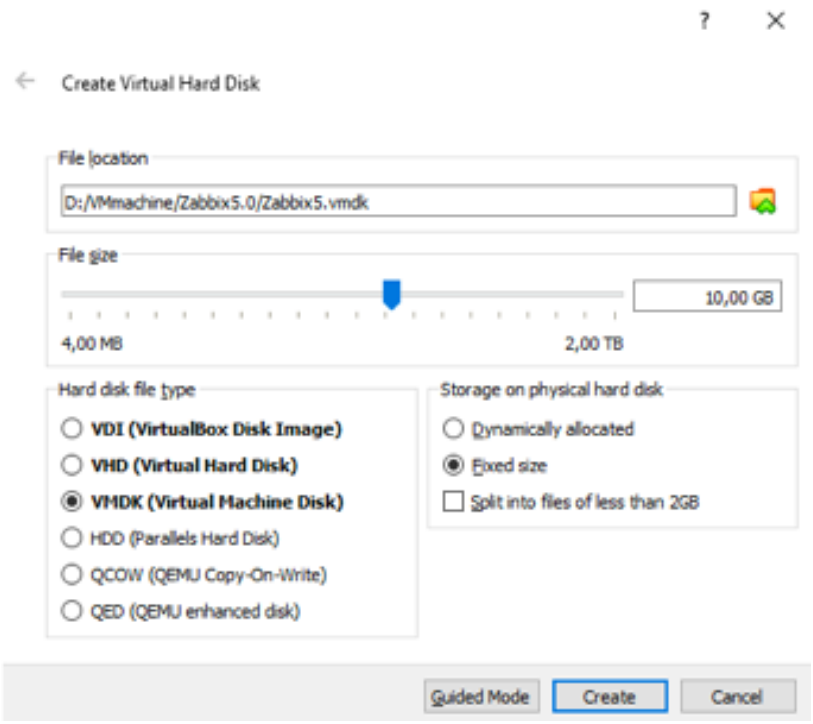

Рис. 3.3. Створення жорсткого диска віртуальної машини

Такий обсяг потрібен для того, щоб його була достатня кількість для вирішення всіх завдань, які будуть розглядатися в нашому курсі. Завершіть

створення віртуальної машини. У Таблиці 3.1 наведені різниці основних типів жорстких дисків, пропонованих VirtualBox:

Таблиця 3.1

Відмінності основних типів жорстких дисків

Тип	Пояснение
VDI (VirtualBox Disk Image)	Собственный формат виртуальных дисков VirtualBox
VHD (Virtual Hard Disk)	Формат, применяемый в среде виртуализации Microsoft, а именно - HyperV
VMDK (Virtual Machine Disk)	Открытый формат, применяемый в среде виртуализации VmWare

У Zabbix є методика розрахунку потрібного розміру бази даних, і її потрібно взяти до уваги при створенні віртуальної машини: розмір бази даних Zabbix залежить від кількості оброблюваних запитів в секунду і налаштування очищення історії в базі даних.

Кількість оброблюваних запитів в секунду - середня кількість нових значень, які Zabbix-сервер отримує кожну секунду. Наприклад: Якщо є 1000 елементів даних з інтервалом перевірки 100 секунд, то кількість оброблюваних запитів за секунду розраховується 1000/100 = 10.

Отже, кожну секунду база даних Zabbix поповнюється десятьма новими записами.

Zabbix зберігає отримані значення деякий момент часу в залежності від налаштувань. Потрібно також врахувати, що кожне нове значення вимагає деякий шматок дискового простору для даних і індексів. Наприклад, якщо потрібно збереження 90 днів історії і кожну секунду в базу даних додається 10 нових записів, загальна кількість значень дорівнюватиме приблизно:

$$\text{Общее количество} = \left(90_{\text{дней}} \cdot 24_{\text{часа}} \cdot 3600_{\text{сек}}\right) \cdot 10 = 77760000_{\text{значений}}$$

Відповідно до типу бази даних, типу отриманих значень (з плаваючою точкою, цілочисельний, рядки, файли журналів і т.д.) може бути потріюно від 40 до кількох сотень байт дискового простору, для зберігання одного значення. Зазвичай одне значення займає, в середньому, 90 байт по числовим елементів даних. Для нас це означає, що для зберігання всіх значень потрібно:

$$77760000_{\text{значений}} \cdot 90_{\text{байт}} = 6998400000_{\text{байт}} = \frac{6998400000_{\text{байт}}}{1024} =$$
$$= 6834375_{\text{МБ}} = \frac{6834375_{\text{МБ}}}{1024} \approx 6.7\ \text{ГБ}$$

дискового простору.

Потрібно відзначити, що розмір бази даних Zabbix спочатку буде невеликим, але стане поступово збільшуватися, і зупиниться після досягнення певного моменту, залежного від налаштувань очищення бази даних (терміну зберігання значень, інструмент HouseKeeper). При впровадженні системи моніторингу Zabbix в діючій мережі - обов'язково потрібно провести розрахунок потрібної вільного простору, виходячи із завдань моніторингу і кількості опитуваних пристроїв.

Повернемося до віртуальної машини - перед запуском залишилося вказати образ Ubuntu для завантаження, і тип мережевого підключення. Потрібно зайти в налаштування віртуальної машини, і в розділі «Носії», додати в контролер IDE, скачаний раніше ISO-образ Ubuntu Server 20.04 (Рис.3.4):

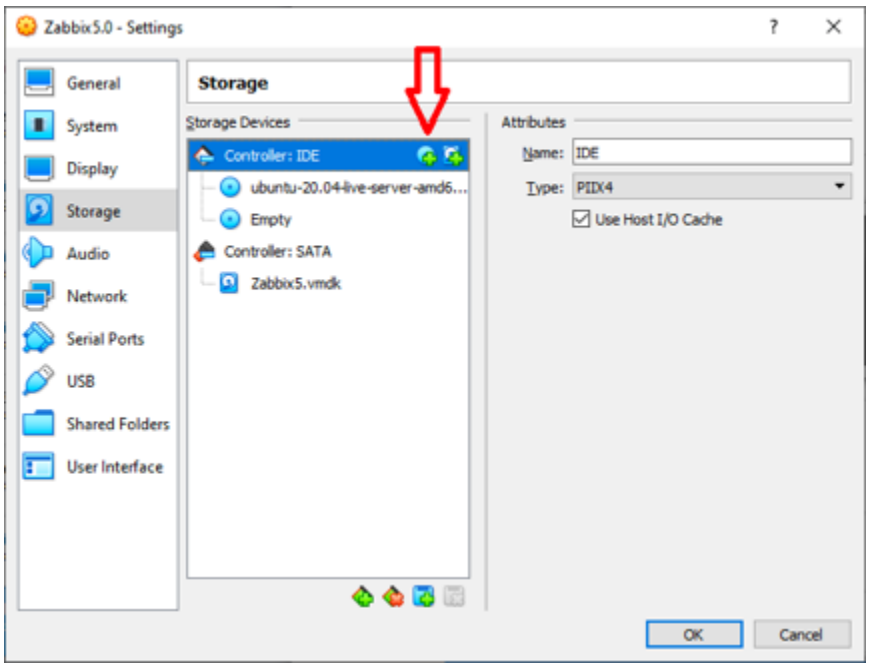

Рис.3.4. Додавання образу Ubuntu для установки

Останній крок перед початком установки Ubuntu на віртуальну машину - організація доступу до мережі Інтернет (він обов'язково знадобиться в подальшому). Вибрав тип підключення «Мережевий міст», або Bridge, як показано на малюнку 3.5. Це означає, що віртуальний мережевий адаптер віртуальної машини буде отримувати IP-адресу з однієї підмережі з Вашим ПК, і на веб-інтерфейс Zabbix буде легко потрапити, вказуючи в браузері ПК IP-адреса віртуального хоста. Так само не буде потрібності додавати правила в віртуальний NAT віртуальної машини, для роботи з різними мережевими інструментами.

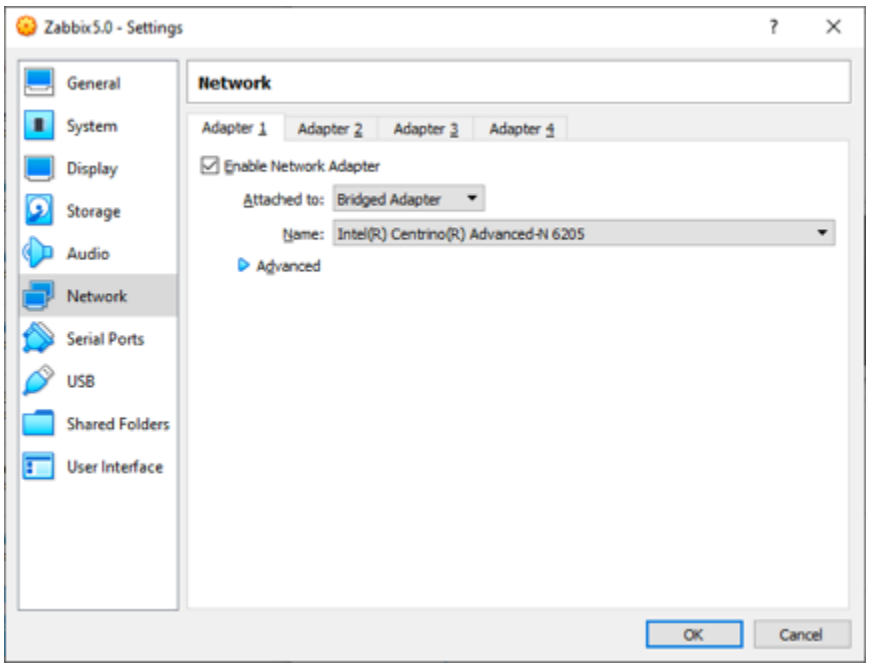

Рис. 3.5. Тип мережевого підключення віртуальної машини

Для спрощення подальшого виконання проєкту, я використав підключення до мережі Інтернет через домашній роутер. Підключення ПК до мережі Інтернет безпосередньо не рекомендується, оскільки для віртуальної машини доведеться використовувати тип підключення NAT і надалі налаштовувати «Проброс портів».

Запустивши віртуальну машину і приступив до встановлення Ubuntu Server 20.04. Не буду детально зупинятися на процесі установки - виберіть анлійську мову установки, розкладку клавіатури English, дія Install Ubuntu, мережевий інтерфейс, проксі, мережеву адресу дзеркала дистрибутивів, настройку використання диска Use an Entire Disk (і потім сам фізичний диск, а після – підтвердження налаштування диска). Далі установник запропонує створити відповідні налаштування профілю - ім'я машини і юзера, а так же пароль. Дані для проєктного профілю наведені в Таблиці 3.2, і на Рис.3.6:

Таблиця 3.2

Дані для проєктного профілю

Параметр	Значение
Ваше имя	zabbix
Имя сервера	zabbix_machine
Имя пользователя	zabbix
Пароль (и подтверждение)	zabbix

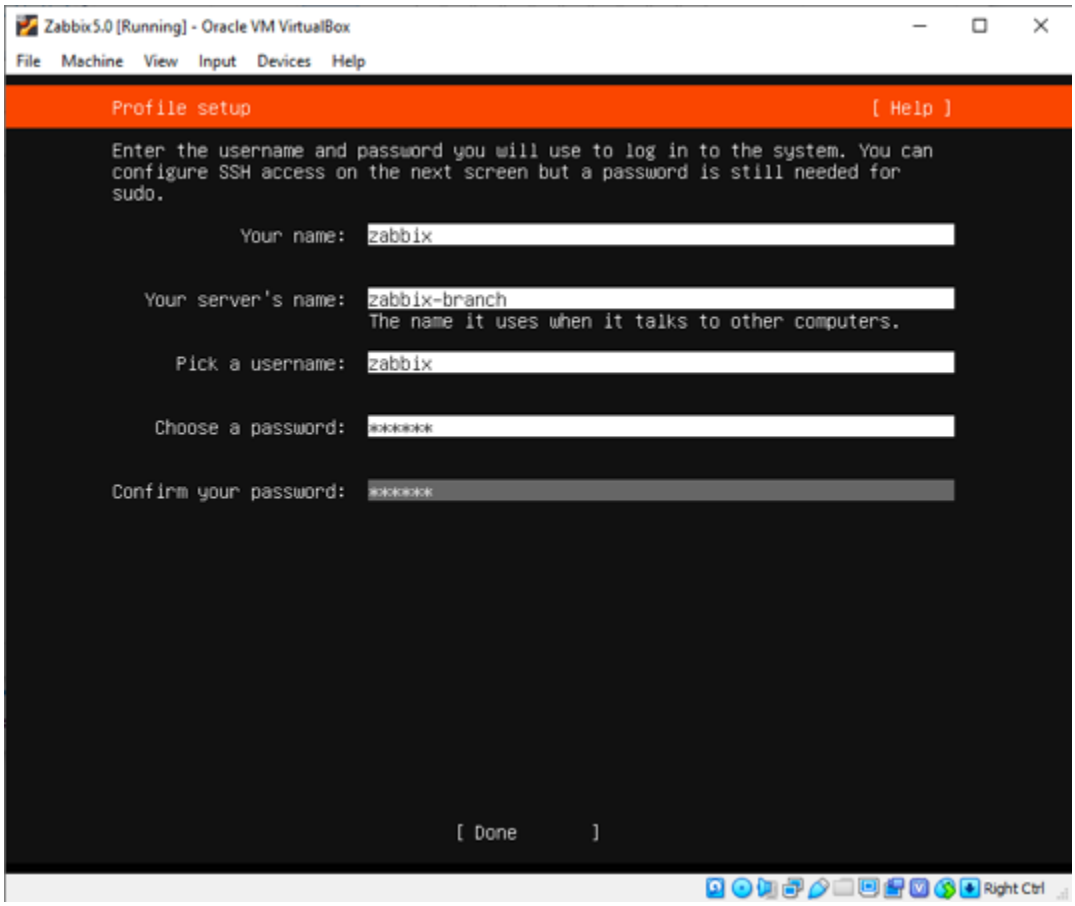

Рис. 3.6. Створення профілю

Далі установник запропонує включити в процес установки SSH-сервер. Потрібно поставити відмітку погодитися. Далі піде вибір популярних інструментів – я пропустив цей крок. Наступним кроком запуститься процес установки Ubuntu Server. Після цього встановлення можно подивитися лог або перезавантажити віртуальну машину. Вибрав перезавантаження (в процесі підготовки до перезавантаження система попросить натиснути Enter).

Після перезавантаження - віртуальна машина готова до роботи. Потрібно розібратися з мережевим підключенням. В ОС Ubuntu 20.04 застосований новий

підхід до конфігурації мережевого інтерфейсу (за допомогою утиліти netplan, вже з ОС Ubuntu Server 18.04 - видна відмова від утиліти net-tools). Тепер файл налаштувань мережевої карти розташований по шляху / etc / netplan /, і має розширення .yaml (в різних інсталяціях він буде називатися по-різному). Переглянемо вміст цього каталогу (Рис.. 3.7):

ls -la / etc / netplan

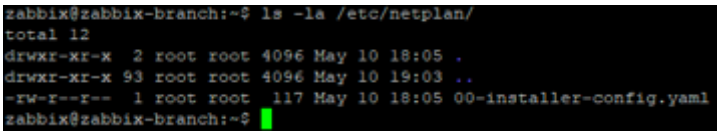

Рис. 3.7. Вміст каталогу /etc/netplan

Якщо у локальної мережі наявний DHCP-сервер, то віртуальна машина отримає IP-адресу автоматично. Перевірити наявність IP-адреси можна командою ifconfig, а доступ в мережу, наприклад, командою ping 8.8.8.8.

Відредагувавши файл з розширенням .yaml - в моєму випадку, відповідно до малюнка 3.7, це /etc/netplan/50-cloud-init.yaml.

Переглянути вміст каталогу можна за допомогою команди dir (або «dir -a», так консоль виведе всі приховані файли і папки), а можна скористатися командою «ls -la», тоді файли відображуу вигляді списку з описом прав, розміру, і дати створення.

Для того, щоб змінити права доступу використовуйте команду sudo, або увійдіть в режим root, за допомогою команди sudo su:

sudo nano /etc/netplan/50-cloud-init.yaml

Конфігурації DHCP виглядає так:

network:

ethernets:

enp0s3:

addresses: []

dhcp4: true

version: 2

Конфігурації статичної IP-адреси виглядає інакше:

network:

ethernets:

enp0s3:

dhcp4: no

dhcp6: no

addresses: [192.168.1.1/24,]

gateway4: 192.168.1.254

nameservers:

addresses: [8.8.8.8, 8.8.4.4]

version: 2

Розглянемо інтерфейси. Перший інтерфейс, для роботи з сервером Ubuntu - інтерфейс командного рядка, або Command Line Interface (CLI). Саме в ньому буде налаштовуватися пакети, працювати зі скриптами, і взаємодіяти з сервісами системи в цілому. Другий інтерфейс, графічний інтерфейс юзера, або Graphical user interface (GUI) - інтерфейс системи моніторингу Zabbix, саме в ньому буду освоювати матеріали курсу.

Різниця між CLI і GUI полягає в тому, що вони призначені для вирішення різних завдань. CLI гнучкий при роботі з різними програмними засобами, майже не вимагає апаратних ресурсів, дозволяє оперувати командами майже як текстовий редактор, і добре підходить для обслуговування сервісів, налаштування конфігурацій, роботи з масивами даних. GUI потрібен для відображення інтерфейсу, дає можливість вирішувати власні задачу по роботі з різними прикладними програмами. В ОС Windows яскравий приклад відмінностей між GUI і CLI - звичний нам віконний інтерфейс і командний рядок CMD.

Для роботи з CLI можна користуватися терміналом середовища віртуалізації. Крім того, якщо доведеться працювати одночасно з різними пристроями (наприклад, декількома серверами і активним мережевим

обладнанням), користуватися терміналами буде складно. Більш комфортне використання при адмініструванні серверів користуватися програмними клієнтами для віддаленого доступу.

Доволі простий, доступний та поширений клієнт віддаленого доступу - PuTTY. Уснує чимало інших програмних клієнтів, що є складнішими у використанні або вимагають абонентської плати. У них є ряд корисних функцій, наприклад, зберігання паролів, зручний інтерфейс для збереження сесій до серверів, багатовіконний режим роботи з декількома пристроями, додаткові надбудови для шифрованого передачі файлів, і багато іншого. Наприклад, це програмні продукти SecureCRT або Xshell.

Можна закантажити PuTTY і віддаленодолучатися до віртуальної машини з його допомогою (Рис. 3.8):

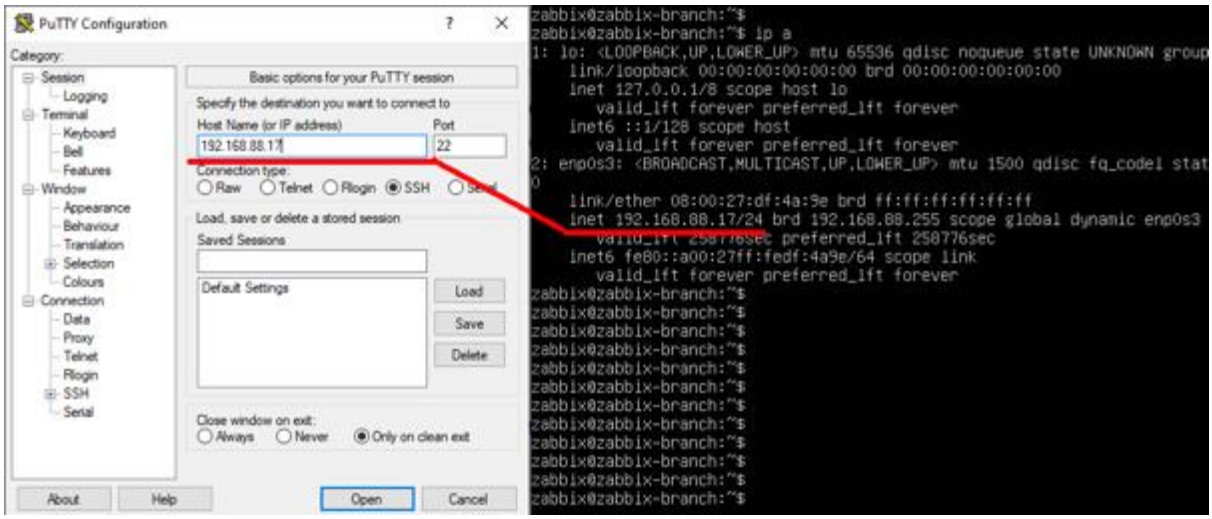

Рис. 3.8. Віддалене підключення до сервера

За допомогою команди «ip a» в терміналі сервера ввести IP-адресу мережевого інтерфейсу, через протокол віддаленого керування SSH, підключаєтеся до сервера клієнтом віддаленого доступу PuTTY.

Після виконаних дій, програма видає повідомлення на запит логіну та пароля.

Провідні віддалені клієнти потрібні для організації логування на стороні адміністратора і керується відповідно до сесій пристроїв (в великих

корпоративних мережах кількість мережевих пристроїв може досягати десятків тисяч, і кожен раз створювати сесію і налаштовувати параметри підключення незручно).

SSH є головним протоколом віддаленого керування на сьогоднішній день, в ОС Ubuntu він включений і преднастроен за замовчуванням. Його основна відмінність від попередника, першого протоколу віддаленого керування, Telnet - в шифруванні всього трафіку між клієнтом і сервером. Telnet передає всі дані - логіни і паролі, що вводяться команди і висновок на них, у відкритому вигляді (так званий plain text), і на сьогоднішній день його використання в принципі не рекомендується.

Далі потрібно оновити всі пакети, що завантажені на ОС. Пакети для ОС сімейства Linux зберігаються в репозиторіях.

Список основних репозиторіїв, з яких ОС Ubuntu викачує різні пакети, знаходиться в файлі /etc/apt/sources.list. Подивитися список можна за допомогою команди cat /etc/apt/sources.list.

Спочатку виконаємо команду, оновлюючу інформацію про пакети, що містяться в репозиторіях, а потім команду, обновляющую пакети

sudo apt-get update

sudo apt-get upgrade

Після закінчення оновлення - система готова до встановлення Zabbix.

3.1.2 Встановлення Zabbix

Приступаючи до встановлення Zabbix, потрібно пам’ятати, що для конкретних збірок Zabbix підходять строго певні ОС сімейства Linux. На офіційному сайті проєкту, zabbix.com, наведені чіткі інструкції, з якого сховища встановлювати пакет для певної ОС.

Якщо використати в якості ОС дистрибутив, відмінний від використовуваного в рекомендаціях- обов'язково потрібно отримати інформацію і правильний репозиторій, і підходящі для нього версії Zabbix.

Заручившись інформацією з zabbix.com, почну установку. Спочатку зазначу репозиторій, з якого потрібно встановлювати пакет. Для проєкту обрана актуальна версія Zabbix 5.0 (Рис. 3.9-3.10).

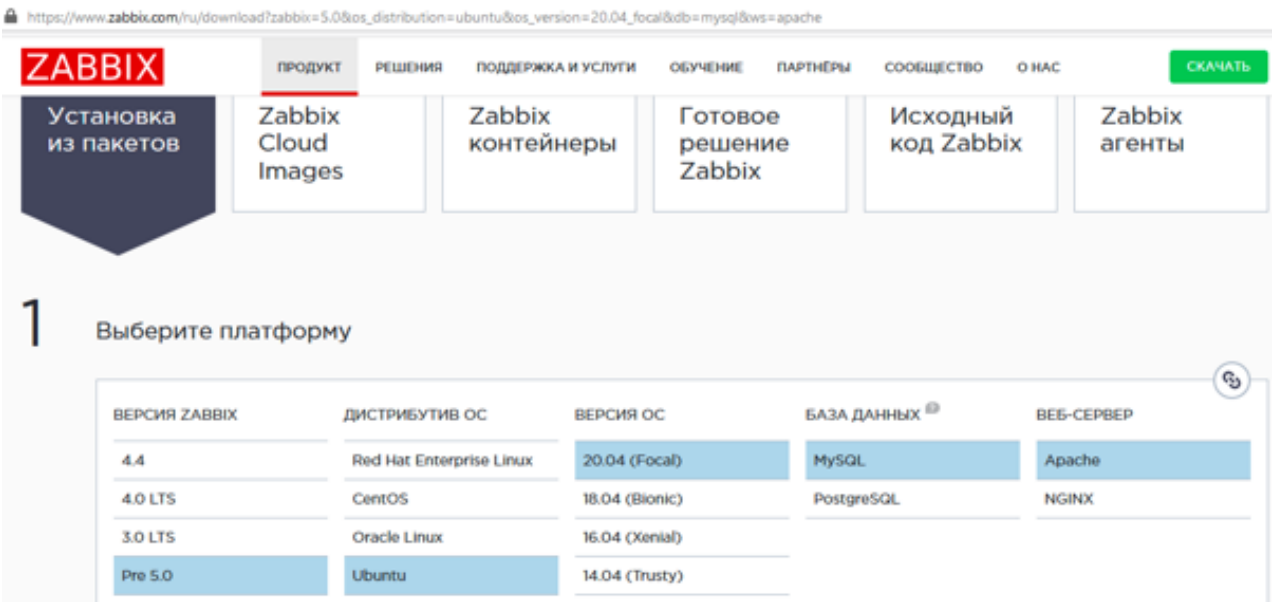

Рис. 3.9. Вибір правильного дистрибутива

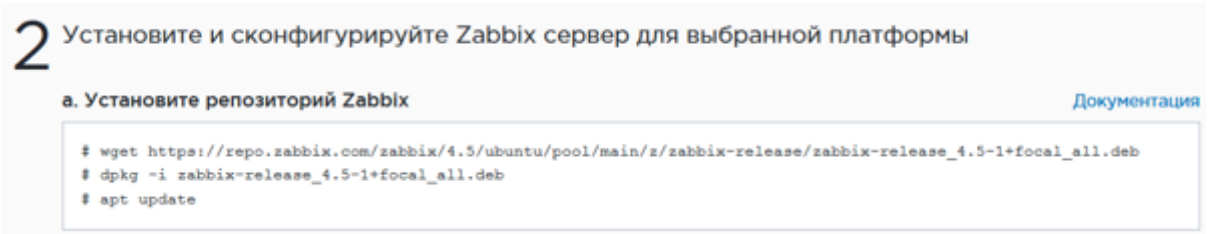

Рис. 3.10. Отримання сховища

Команда wget помістить в файл /etc/apt/sources.list, що має список репозиторіїв за замовчуванням, новий шлях, який вказує на пакет zabbix, відповідний до моєї системи ОС Ubuntu.

wget https://repo.zabbix.com/zabbix/4.5/ubuntu/pool/main/z/zabbix-release/zabbix-release_4.5-1+focal_all.deb

Команда dpkg -i витягує інформацію про складові релізу Zabbix версії 4.2.

dpkg -i zabbix-release_4.5-1 + focal_all.deb

Оновити інформацію можна за допомогою команди:

sudo apt-get update

Список додаткових репозиторіїв, які додаються користувачем та є повноцінними сховищами великих продуктів, знаходяться в файлі /etc/apt/sources.list.d/zabbix.list.

Команда dpkg -i оновила відомості про доступні для установки модулів пакета zabbix. Після введення в консолі команди «sudo apt-get install zabbix-» та натиснувши клавішу TAB - система запропонує можливі варіанти подальшого введення. (Рис. 3.11).

Рис. 3.11. Інструмент швидкого введення команд

Після всього цього поновив інформацію про репозиторії і встановив необхідні пакети для Zabbix - «back-end», веб-інтерфейс, і агент (знадобиться пізніше):

sudo apt-get install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-agent

Після встановлення програми переходимо до наступного етапу установки— створення бази даних для Zabbix в сервісі баз даних MySQL: Потрібно виконати скрипт (готовий набір команд) для створення бази даних Zabbix.

Для проєкту досить простий БД MySQL. При впровадженні Zabbix для моніторингу, скажімо, від 500 об'єктів (наприклад, велике підприємство), буде потрібно БД PostgreSQL.

Перед початком роботи потрібно встановити mysql-server, а тільки після цього можна зробити вхід в БД MySQL

sudo apt-get install mysql-server

sudo mysql -uroot -proot

Створимо базу даних zabbix_db, з кодуванням utf-8:

mysql> create database zabbix_db character set utf8 collate utf8_bin;

mysql> create user zabbix_us @ localhost identified by 'zabbix_pw';

mysql> grant all privileges on zabbix_db. * to zabbix_us @ localhost;

Дана команда створює юзера і код для доступу до конкретної бази.

mysql> quit;

В таблиці 3.3 показані параметри доступу до бази даних.

cd / usr / share / doc / zabbix-server-mysql

sudo zcat create.sql.gz | mysql -uzabbix_us -pzabbix_pw zabbix_db

Ця операція може зайняти кілька хвилин, дочекаймося її закінчення.

При виконанні цього пункту потрібно дотримуватися послідовності дій і орієнтуйтеся на дані Таблиці 3.1.2.1, для розуміння скоєних дій.

Наступний етап установки - зміна конфігураційного файлу zabbix_server.conf. У ньому потрібно правильно зазначити назву бази даних для Zabbix, і важливо внести логін та пароль.

У файлі zabbix_server.conf:

sudo nano /etc/zabbix/zabbix_server.conf (табл. 3.3):

Таблиця 3.3

Параметри доступу до бази даних

Параметр базы данных	zabbix_server.conf до редактирования	zabbix_server.conf после редактирования
	#DBHost=localhost	DBHost=localhost
Название базы zabbix_db	#DBName=zabbix	DBName=zabbix_db
Пользователь zabbix_us	#DBUser=zabbix	DBUser=zabbix_us
Пароль к базе zabbix_db	#DBPassword=zabbix	DBPassword=zabbix_pw

Процес Zabbix сервера:

sudo service zabbix-server start

Для того, щоб процеси запускалися автоматично:

sudo update-rc.d zabbix-server enable

sudo service zabbix-agent start

sudo update-rc.d zabbix-agent enable

Майже закінчив. Залишилося небагато - налаштувати конфігурацію PHP для веб-інтерфейсу Zabbix. Файл конфігурації Apache для веб-інтерфейсу Zabbix розміщується в /etc/apache2/conf.d/zabbix або /etc/apache2/conf-enabled/zabbix.conf. Майже всі параметри конфігурації PHP вже задані.

Відреагую файл zabbix.conf для веб-сервера:

sudo nano /etc/apache2/conf-enabled/zabbix.conf

В розділі:

<IfModule mod_php5.c>

Розкоментував рядок:

php_value date.timezone Europe / Moscow

В розділі:

<IfModule mod_php7.c>

Розкоментував рядок:

php_value date.timezone Europe / Moscow

Після зміни файлу конфігурації перезапустив веб-сервер apache:

sudo service apache2 restart

Отже, установка закінчена!

3.2. Перший запуск веб інтерфейсу

Інформація про користувачів знаходиться в «Адміністрування → Користувачі». Спочатку в Zabbix тільки два попередньо встановлених користувача: «Admin» – суперкористувач Zabbix, який має всі привілеї та «Guest» – спеціальний користувач за замовчуванням. Якщо користувач не увійшов до системи, він отримає доступ з привілеями користувача «guest». За замовчуванням, «guest» не має дозволів на об'єкти Zabbix. Для додавання нового користувача натиснемо «Створити користувача». білої теми. Додаємо на наступній вкладці способи оповіщення користувача та вводимо e-mail. Можна вказати період часу, коли цей спосіб буде активним, за замовчуванням спосіб активний завжди. Також можна настроїти важливість тригера, для якого спосіб сповіщення буде використовуватись. І на останній вкладці додамо прав новому користувачеві. Можна також керувати правами доступу користувачів через групи користувачів.

У випадку IP-адреса інтерфейсу eth0 - 192.168.88.17.

В адресну строку необхідно ввести http: // IP-адрес_віртуальной_машіни / zabbix.

У нашому випадку це http://192.168.88.17/zabbix - результат на малюнку 3.12:

Рис. 3.12. Перший запуск системи Zabbix

Після заупуску сервера, йому необхідно пройти самотестування. Переходимо до наступного кроку потрапляємо в меню перевірки налаштувань (Рис. 3.13).

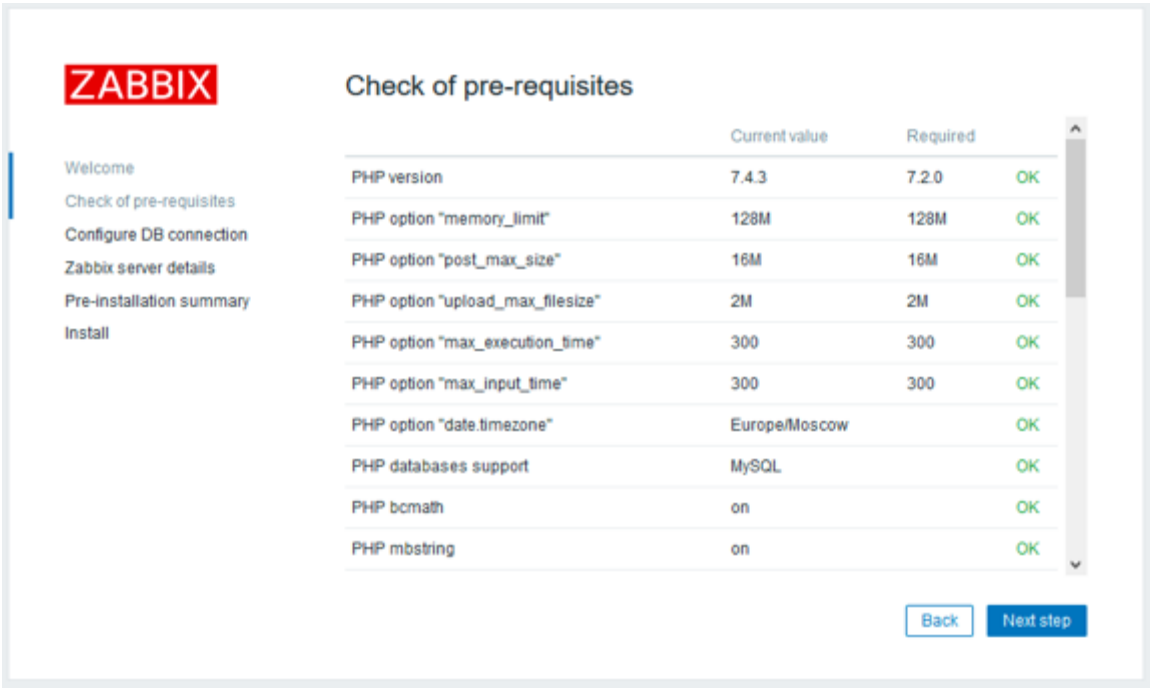

Рис. 3.13. Перевірка системи

Для запуску необхідно, щоб кожен пункт працював коректно. Переходимо в меню налаштувань доступу до БД (Рисунок 3.14).

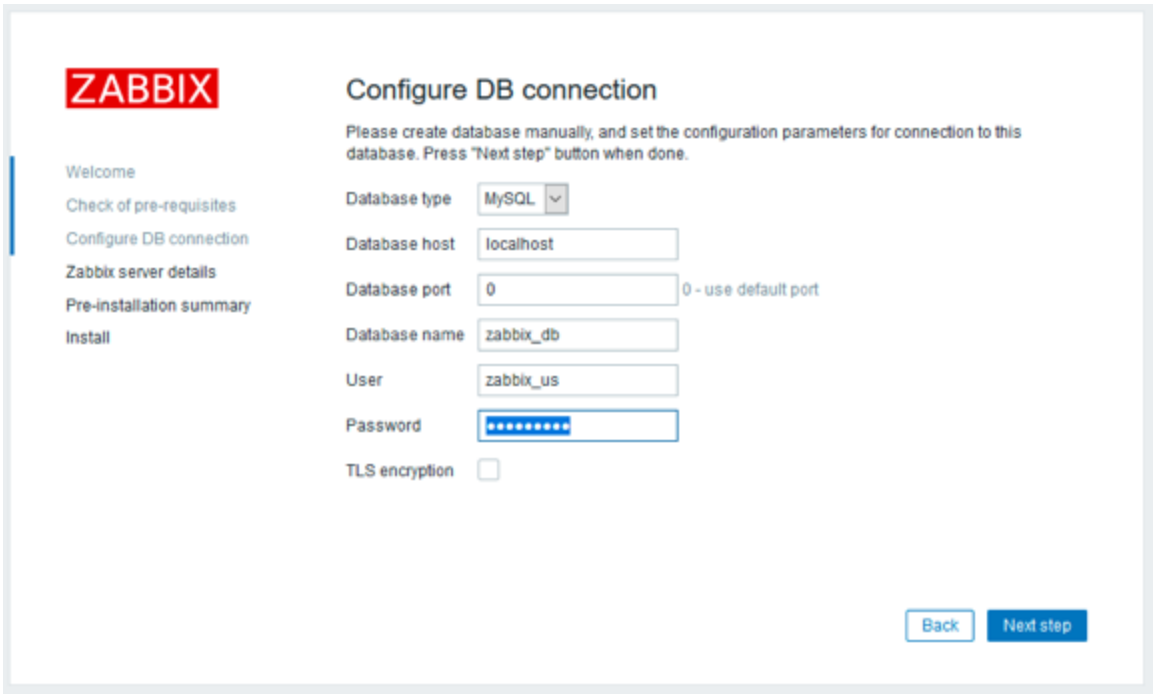

Рис. 3.14. Налаштування БД

Далі переходимо в меню налаштувань імені (Рис. 3.15). Вказуємо ім’я свого хосту.

Рис. 3.15. Налаштування імені сервера

Далі переходимо до перевірки налаштувань. Після наступного натискання переходимо до фінального меню (Рис. 3.16).

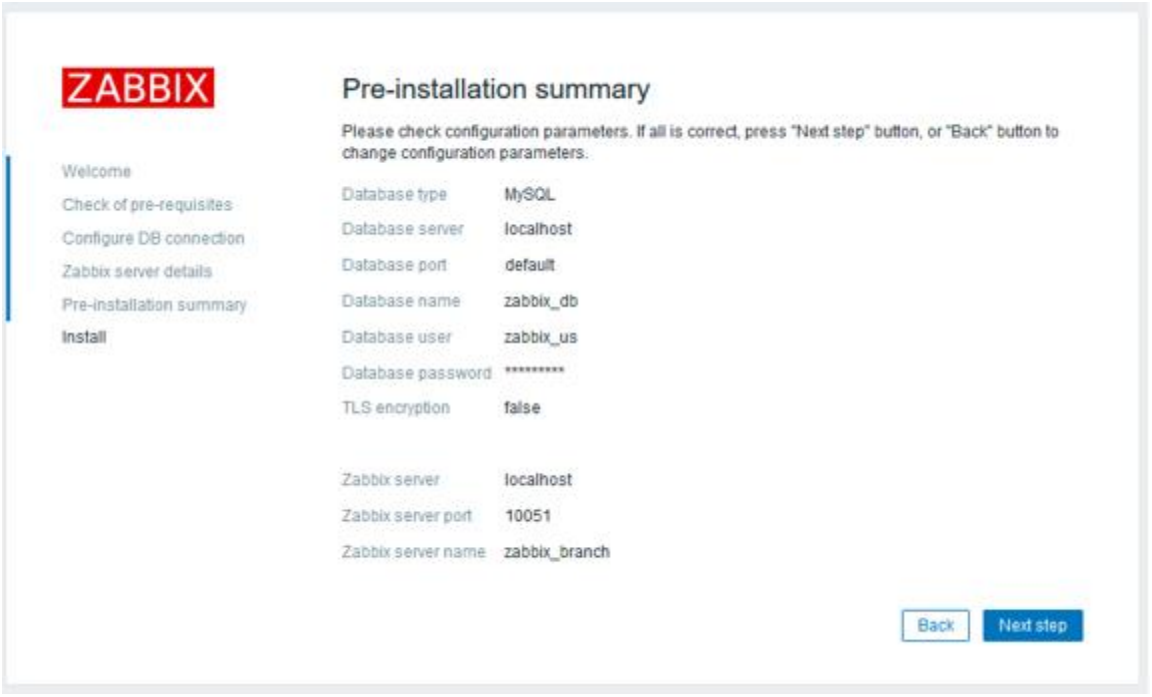

Рис. 3.16. Завершення установки

Фінальне вікно запуску системи представлене на рисунку 3.17.

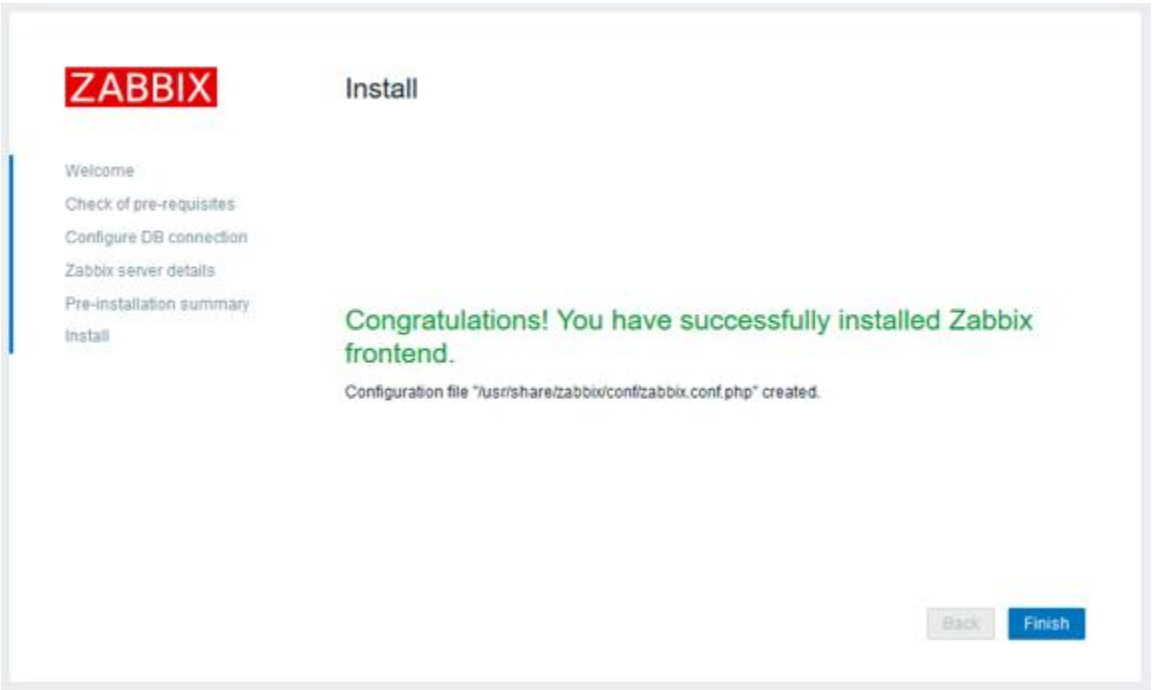

Рис. 3.17. Запуск системи

Отже, після натискання кнопки «Finish» потрапив в меню введення логіна і пароля (Рис.3.18).

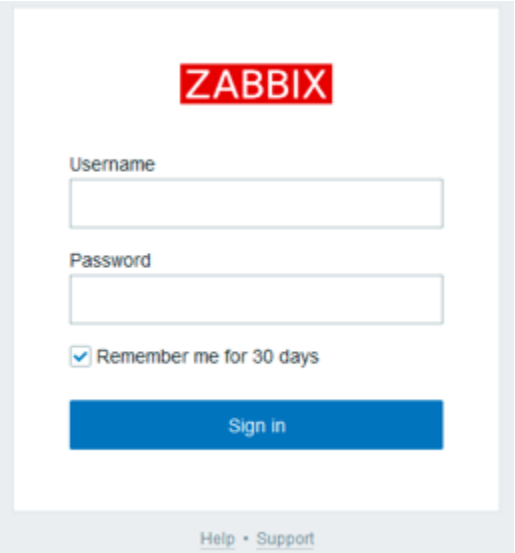

Рис. 3.18. Запрошення для входу в систему

Це перше меню налаштованої і готової до роботи системи моніторингу Zabbix. Логін і пароль за замовчуванням - Admin / zabbix. При першому вході можна побачити основну панель Zabbix (Рис. 3.19):

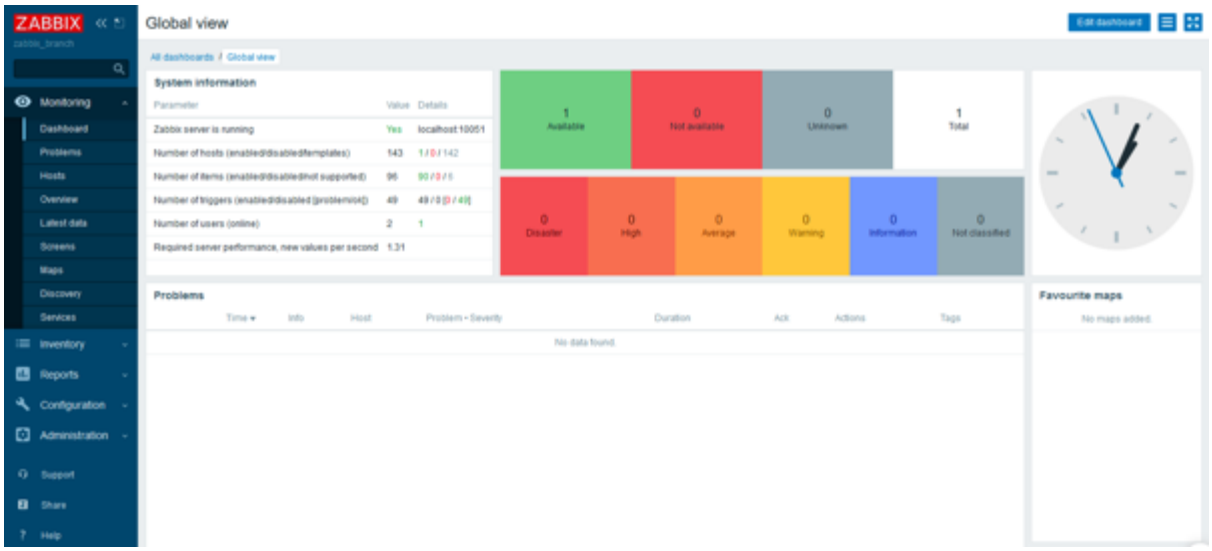

Рис. 3.19. Основна панель Zabbix

Ось і успішне розгортанням системи моніторингу! Взагалі, на офіційному сайті проєкту можна скачати готові інсталяції, образи, докери - але я вважаю, що потрібно усвідомлювати і розуміти етапи установки і налаштування елементів системи моніторингу. А також готові інсталяції тільки на CentOs, а деякі додаткові програмні забезпечення є тільки в Debian системах, такі як libphone та esim

Коли вирішимо закінчити роботу, "виключення" ОС Ubuntu (аналог завершення роботи в Windows) здійснюється командою:

sudo shutdown -H now -P

При такому відключенні буде коректно завершуватися робота всієї віртуальної машини. Щоб подивитися можливі параметри команди shutdown, запускайте її в вигляді «shutdown --help».

Результатом виконання даного розділу повинен став скріншот веб-інтерфейсу успішно запущеної системи .моніторингу, з панеллю System Information, в якій статус запуску системи має значення «Yes» (Рис. 3.20).

Якщо буде проблема з підключенням до БД, веб-інтерфейс буде працювати, але реально система моніторингу не працюватиме, і статус запуску системи матиме значення «No».

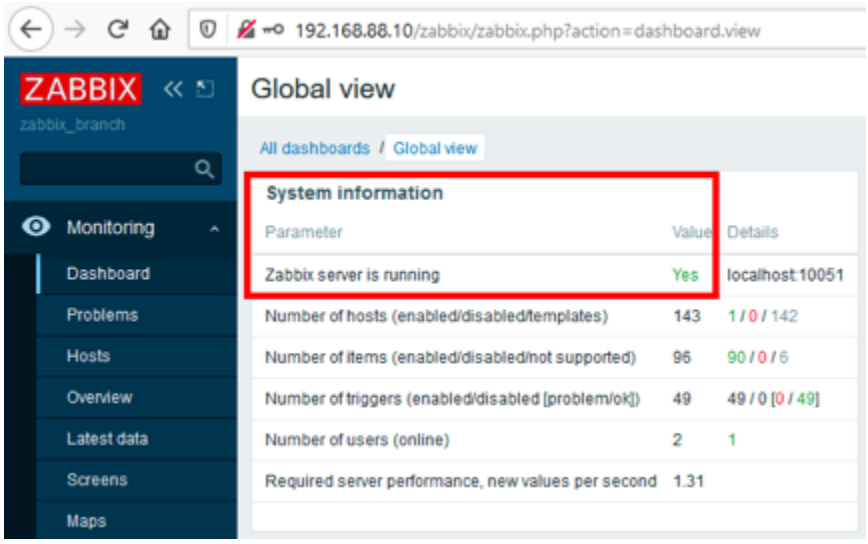

Рис. 3.20. Звіт про виконану роботу

3.3 Налаштування Zabbix та знайомство с системою

Отже, в минулому підрозділі розгорнули готову для роботи систему моніторингу. Запустивши середу віртуалізації Oracle VM VirtualBox, запустив віртуальну машину і дочекався завантаження ОС. Після завантаження ОС упевніться, що з віртуальної машини доступна мережа Інтернет (при введенні команди ifconfig - інтерфейс eth0 має IP-адресу, ping 8.8.8.8 показує доступність сервера 8.8.8.8).

У браузері ПК ввів http: // IP-адрес_віртуальной_машіни / zabbix, логін і пароль за замовчуванням - Admin / zabbix. Перевірте, що веб-інтерфейс доступний (Рис. 3.21):

Рис. 3.21. Запрошення для входу в систему

Далі необхідно виконати невеликий тюнинг доступу до веб-інтерфейсу. Потрібно перейти за IP-адресою віртуальної машини доступна сторінка-заглушка веб-сервера Apache2 (Рис.3.22):

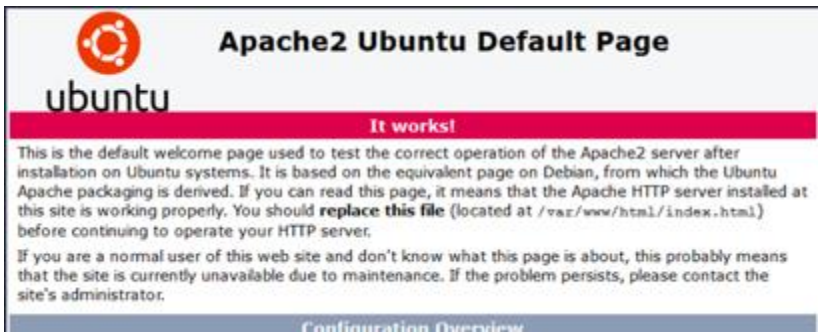

Рис. 3.22. Сторінка-заглушка веб-сервера Apache2

Її необхідно змінити, щоб при зверненні на IP-адресу віртуальної машини, попадати відразу в веб-інтерфейс Zabbix. Після редагування фвйлу налаштувань 000-default.conf, що відповідає за роботу з віртуальними хостами:

sudo nano /etc/apache2/sites-available/000-default.conf

змінив рядок

DocumentRoot / var / www / html

Перетворивши його:

DocumentRoot / usr / share / zabbix

Зберіг зміни, і перезапустив сервіс apache2

sudo service apache2 restart

Після цього веб-інтерфейс Zabbix став доступний безпосередньо по IP-адресою.

Після входу в систему — інтерфейс Zabbix англійською мовою. Перекладемо інтерфейс російською (української мови немає в Zabbix).

Переглянемо в консолі віртуальної машини список доступних мов та оберемо потрібну та зрозумілу для нас. (Рис. 3.23):

sudo locale -a

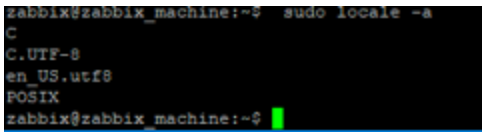

Рис. 3.23. Перегляд «локалей»

Очевидно, російська мова не встановлено. Щоб подивитися, які мови (так звані «локалі» - мова і кодування. Можливо встановити на дану ОС, виконайте команду:

cat / usr / share / i18n / SUPPORTED | grep ru_

Команда cat (Рис. 3.24) служить для простого відображення вмісту текстового файлу на екран. Приставка «|grep»вказує вивести тільки ті рядки, в яких є збіг із зазначеними після «| grep» символами (вибірка в даному випадку відбувається за випадковим збігом з «ru_»).

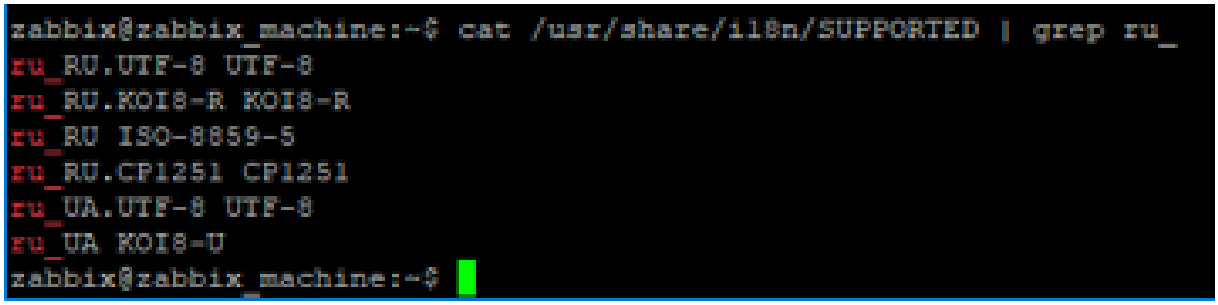

Рис. 3.24 Перегляд «локалей»

Для інсталювання вибраної мови необхідно виконати наступні дії:

sudo locale-gen ru_RU

sudo locale-gen ru_RU.UTF-8

sudo dpkg-reconfigure locales

Після того, як ввести команди на зміну налаштувань буде запропоновано вибрати з великого списку, де потрібно ввести зміни — можна вибрати все (що займе багато часу), або тільки en і ru для UTF-8. Просто двічі виберіть Ok в з'являться меню — і будуть налаштовані конфігурації для російської мови.

Після того як консоль встановлена і ініціалізована, необхідно перезавантажити веб-сервер Apache, щоб мова змінилась і став доступний в Zabbix (веб-інтерфейс Zabbix працює на базі веб-сервера Apache):

sudo service apache2 restart

Останнім кроком залишилося — включити в настройках Zabbix російську мову (Рис. 3.25):

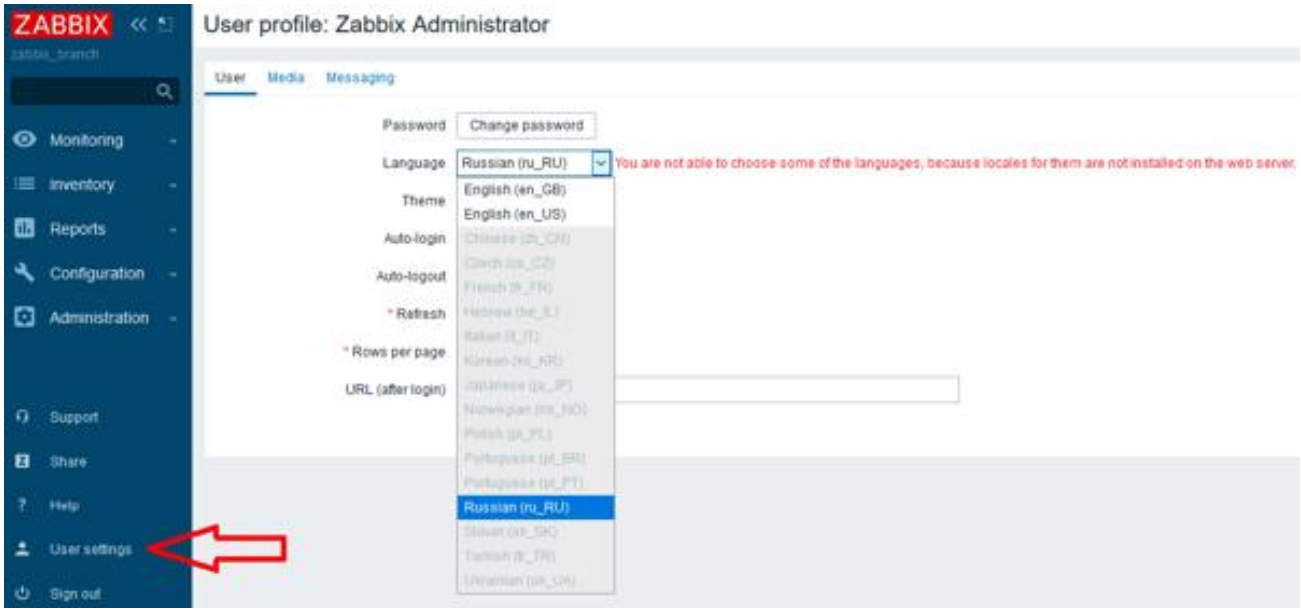

Рис. 3.25. Встановлення російської мови

Щоб додати пристрої до системи Zabbix, необхідно налаштувати вузли мережі в програмі. Вузол мережі в Zabbix – спостерігається об'єкт мережі (фізичний або віртуальний). Визначення того, що може бути «вузлом мережі» в Zabbix, дуже гнучке. Це може бути фізичний пристрій, мережевий комутатор, віртуальна машина або якась програма.

Додавання вузла мережі.

Інформація про налаштовані вузли мережі в Zabbix доступна в «Налаштування → Вузли мережі». Для додавання нового вузла мережі натискаємо "Створити". Додамо пристрій NetPing2/PWR-220 v3/ETH – це пристрій дистанційного керування розетками електроживлення по мережі Ethernet, також має чотири входи/виходи для підключення датчиків.

Ім'я вузла мережі – унікальне ім'я вузла мережі. Дозволено буквено-цифрові символи, пробіли, точки, тире та підкреслення. Мабуть ім'я – це ім'я буде видиме у списках, картах та інше. Цей атрибут має підтримку UTF-8.

Групи – вибір груп вузлів мережі, до яких належатиме вузол мережі. Вузол мережі повинен належати принаймні до однієї групи вузлів мережі. Нова група

вузлів мережі –можна створити нову групу та приєднати до неї вузол мережі. Ігнорується, якщо поле пусте.

Інтерфейси - інтерфейс, за допомогою якого Zabbix зв'язується з вузлом мережі. Підтримуються кілька типів інтерфейсів: агент, SNMP, JMX та IPMI. Натискаємо "додати", вводимо IP-адресу нашого пристрою в поле "інтерфейси агента" та "інтерфейси SNMP".

Щоб визначити проміжок часу зберігання зібраних даних в БД, потрібно увійти в меню Адміністрування - Загальні - Очищення історії (Рис. 3.26). Я вибрав період зберігання даних - 30 днів.

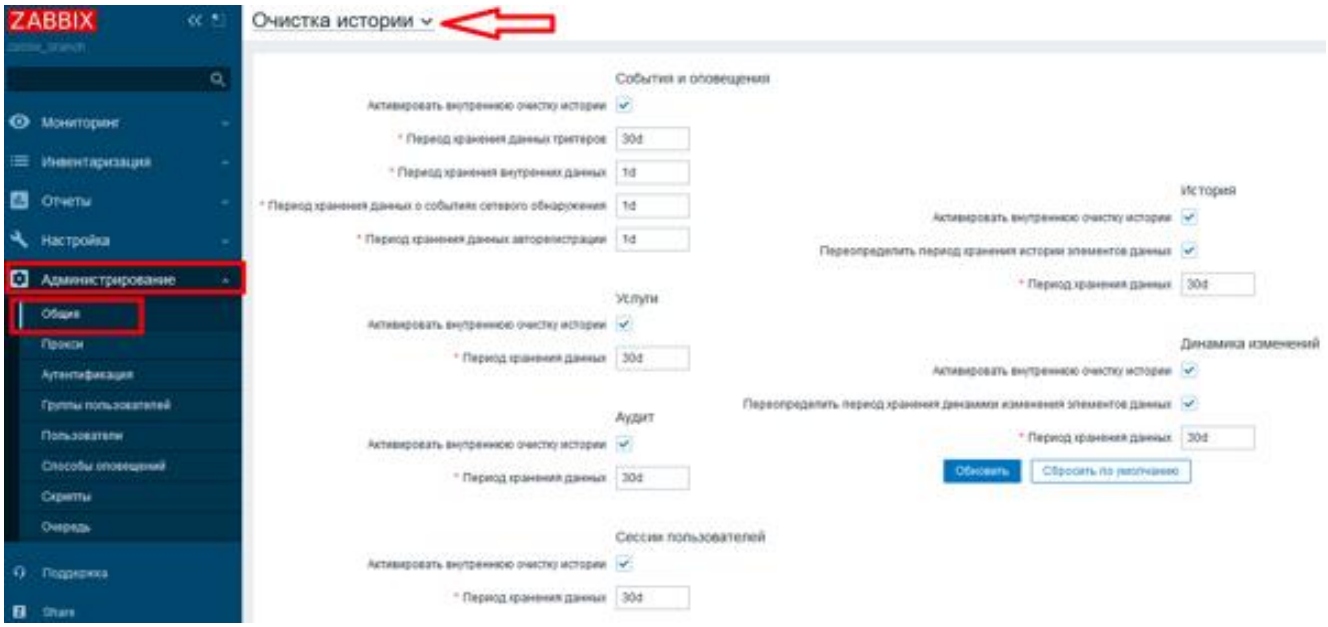

Рис. 3.26. Настройка часу зберігання даних в БД

Фіксувати дані потрібно в тому часовому поясі, який стане точкою відліку для кореляції подій.

Якщо мова йде про моніторинг обладнання компанії, що знаходиться, наприклад, в одному місті, зручніше використовувати місцевий часовий пояс. Якщо ж об'єкти моніторингу знаходяться по всьому світу, краще використовувати 0 за грінвичем час, або час, прийнятиій в компанії. Це допоможе уникнути різночитань при аналізі подій, встановлюючи для всіх єдину точку відліку фіксації і відображення цих подій.

Єдина точка відліку часу повинна бути встановлена і в системі моніторингу, та на яку обслуговує мережевому і серверному обладнанні (необхідна кореляція зафіксованих подій в системі моніторингу та, наприклад, в логах syslog-сервера).

Ще одним перспективним напрямком удосконалення систем моніторингу комп'ютерних мереж є оптимізація розмежування потоків даних, низька ефективність якої дає нині до чверті вразливостей мережі [13]. При цьому необхідно інтегрувати системи контролю безпеки та розмежування потоків даних.

Зокрема, типові стандартні рішення безпеки рідко передбачають механізм адаптації до різних конфігурацій мереж, унаслідок чого унеможливлюється оптимальне використання вільних ресурсів мережі. Понад те, часто відбувається конкуренція за обчислювальні ресурси з додатками, що працюють у мережі у той час. Для оптимізації роботи потрібен поділ інформаційних потоків у реальному часі, а також скорочення часу опитування робочих станцій, що здійснюється системою моніторингу; загалом - динамічний багатопотоковий збір інформації системою моніторингу [42].

У найбільш загальному вигляді система моніторингу має здійснювати два стратегічні види контролю.

Контроль цілісності системи, тобто. стану мережі, при якому комп'ютерна система функціонує як логічно єдина система апаратних та програмних засобів (елементів системи), що повноцінно забезпечують роботу захисних механізмів, включаючи логічну коректність роботи та нормальне функціонування у плані нейтралізації загроз безпеці. Також сучасні системи контролю за цілісністю повинні відстежувати розподілені зміни мережі і мати захист від несанкціонованої модифікації потоків даних між вузлами мережі.

Контроль захищеності системи, тобто. спроби санкціонованого «зламування» інформаційної системи, що здійснюється організацією-власником з метою виявлення наявних уразливостей у захисті мережі, які доцільно виявляти раніше зловмисників. Особливо актуальний цей метод при введенні нового

програмного забезпечення (оновлення версій, що істотно відрізняються), а також у разі зміни кадрового складу співробітників, що працюють з відповідним вузлом мережі.

Системи моніторингу комп'ютерних мереж є необхідною складовою загального забезпечення інформаційної безпеки організації. Ефективність забезпечення інформаційної безпеки залежить від того, наскільки однозначно сформульовані вимоги до оперативних (поточних), тактичних та стратегічних завдань, і наскільки цілісно при цьому забезпечено їхній взаємозв'язок [48]. Системи моніторингу як частина системи інформаційної безпеки призначені для забезпечення вирішення оперативних та частково тактичних завдань, сприяючи досягненню стратегічних цілей безпеки.

Необхідно враховувати, що побудова мереж може відповідати різним системам стандартів, включаючи міжнародні [47], а сучасні стандарти освіти далеко не завжди відповідають набору компетенцій [22] вимогам для фахівців з безпеки [27], особливо якщо враховувати специфіку конкретних інформаційних систем. Як наслідок, доцільно заздалегідь залучати студентів вищих навчальних закладів [15] до відповідних організацій регіону [39] у рамках соціального партнерства [6], з подальшим працевлаштуванням [37], що підвищить впевненість студентів [17] та дасть мотивацію до якісного навчання [4]], при цьому молоді фахівці вже будуть в змозі виконувати задачу щодо забезпечення безпеки інформаційних систем організації [14] без додаткового періоду навчання [21]. Таким чином, для залучення кваліфікованих молодих спеціалістів має сенс здійснювати моніторинг [8] вищих навчальних закладів регіонів [28], відстежуючи їх відповідність до сучасного рівня освіти, наявності організаційної культури [30]. Загалом доцільно інтегрувати вищий навчальний заклад [20] із соціокультурним простором [23] регіону [18] – ефективність такого підходу підтверджується зарубіжним досвідом [19].

Меню *Моніторинг* містить всі можливі види відображення даних, що збираються. Будь-яка інформація, яку агрегує Zabbix (опитуючи різні об'єкти), відображається в загальній статистиці, а події що відбуваються поділяються за

рівнями важливості, візуалізація зібраних даних передбачена у вигляді графіків, карт мереж, і всіляких звітів. Всі ці дані відображаються в різних розділах Моніторингу.

Меню *Моніторинг* - *Панель* показує короткий підсумок всієї інформації. Тут гнучко налаштовуються панелі відображення зібраних даних, є можливість розміщувати прямі посилання на обрані графіки, комплексні екрани і карти мереж (обрані - найважливіші для адміністратора системи, по суті, швидкі посилання на важливі об'єкти).

Можна створити свою панель моніторингу. Залишимо на панелі 4 віджета (Рис. 3.27).

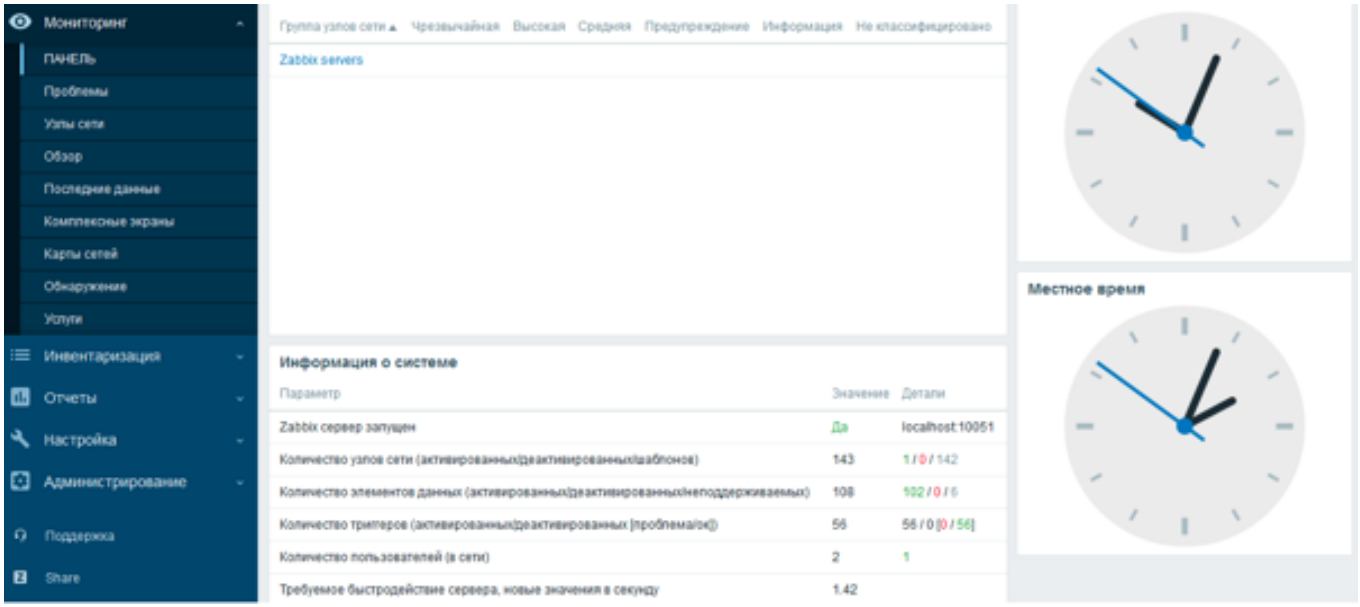

Рис. 3.27. Створення панелі моніторингу

Далі організуємо моніторинг першого об'єкта, за допомогою Zabbix-агента, який встановлю як пакет.

Поправили файл налаштувань агента:

sudo nano /etc/zabbix/zabbix_agentd.conf

Надав демону права root для збору даних:

AllowRoot = 1

Перезавантажити Zabbix-agent:

sudo /etc/init.d/zabbix-agent restart

У Zabbix ініціалізовано вузол мережі Zabbix server, що використовується для моніторингу власних параметрів. Після того, як дав агенту права на моніторинг системи, де він встановлений, він почне збирати дані системі Zabbix і ОС Ubuntu Server.

Вибравши *Проблеми* і *Огляд* надають лістинг зареєстрованих подій (спрацювали тригерів). Вкладка *Вузли* мережі надає детальний огляд станів вузлів мережі (об'єктів моніторингу), звідси здійснюється доступ до корисних інструментів відображення даних. Найголовніше - звідси здійснюється доступ до графіків (Рис. 3.28):

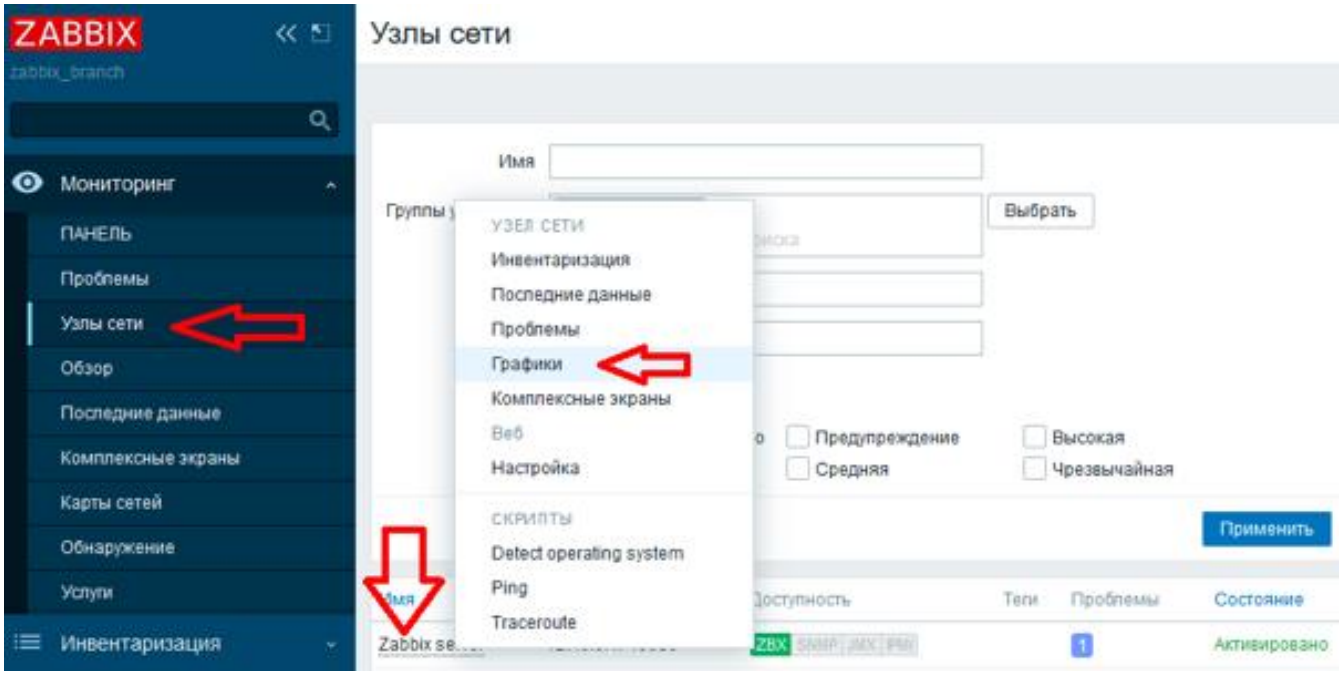

Рис. 3.28. Вкладка Моніторинг - Вузли мережі

Доступ до графіків в попередніх версіях здійснювався через окремий пункт меню Моніторингу. Зараз він організований через вкладку Вузли мережі. На мою думку, це зроблено для того, щоб підкреслити, що графіки не є єдиним інструментом відображення і аналізу даних. Графіки будуються на основі зібраних даних (Рис. 3.29). Кожен графік потрібно «спроєктувати» на етапі створення шаблону опитування для об'єкта моніторингу. Це меню дуже

популярно, оскільки саме тут можна візуалізувати події, що відбулися деякий час назад і зафіксовані системою моніторингу в числових значеннях. Масштабованість графіків і можливість виділення певного часового відрізку дозволяють швидко отримати необхідну інформацію.

Рис. 3.29. Інструмент моніторингу Графіки

Меню *Моніторинг* - *Останні дані* відображає безпосередньо зібрані значення метрик, або елементів даних. Інструмент корисний для перевірки працездатності метрик як таких, і, в разі потрібності, їх налагодження, в значенні troubleshooting. Так само, тут можна за допомогою фільтрів візуалізувати отримувані значення від одного або декількох вузлів (Рис. 3.30):

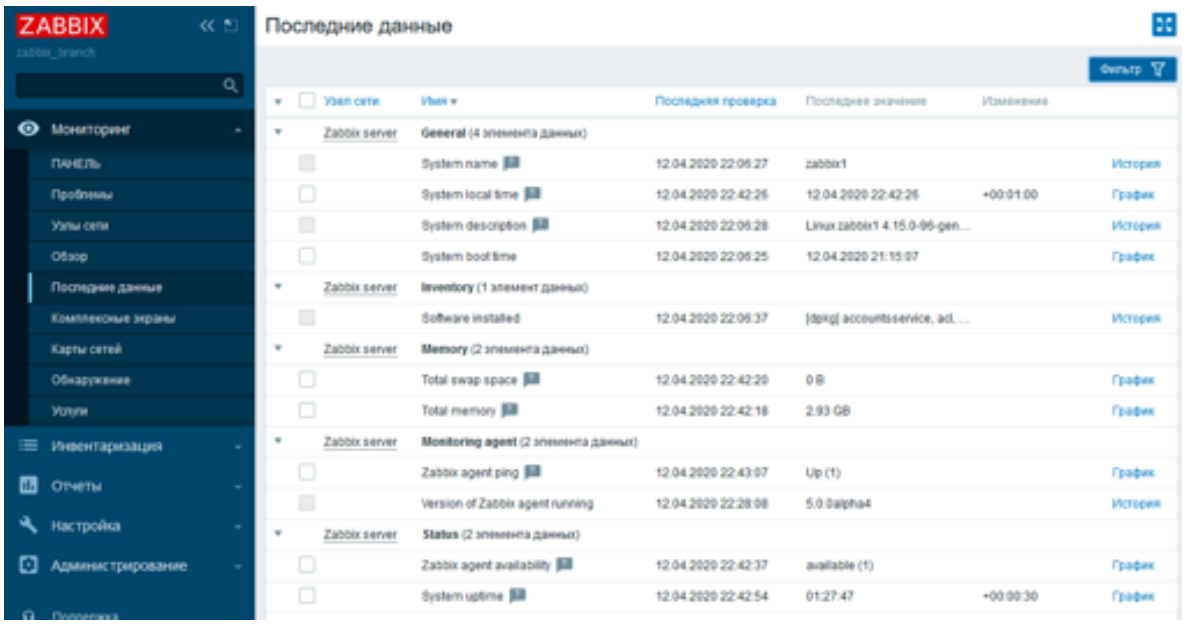

Рис. 3.30. Вкладка Моніторинг - Останні дані

Меню *Моніторинг - Комплексні екрани* призначене для розміщення відразу декількох об'єктів на одній веб-сторінці, це можуть бути карти, графіки, годинник (чомусь це нагадує основну панель). Комплексні екрани - інструмент відображення необхідних параметрів для спостережуваного процесу або об'єкта, зібраних в одному місці (наприклад, при здійсненні моніторингу сервера на відповідному комплексному екрані відображаються графіки завантаження процесора, температури, зайнятої оперативної пам'яті, і вільного місця на жорстких дисках). Однак, в Zabbix є серйозності відмінності щодо доступу та використання комплексних екранів, створюваних вручну користувачем, і автоматично при роботі шаблону опитування. Для малюнка 3.31 узятий готовий комплексний екран, встановлений в системі, і доповнений вручну.

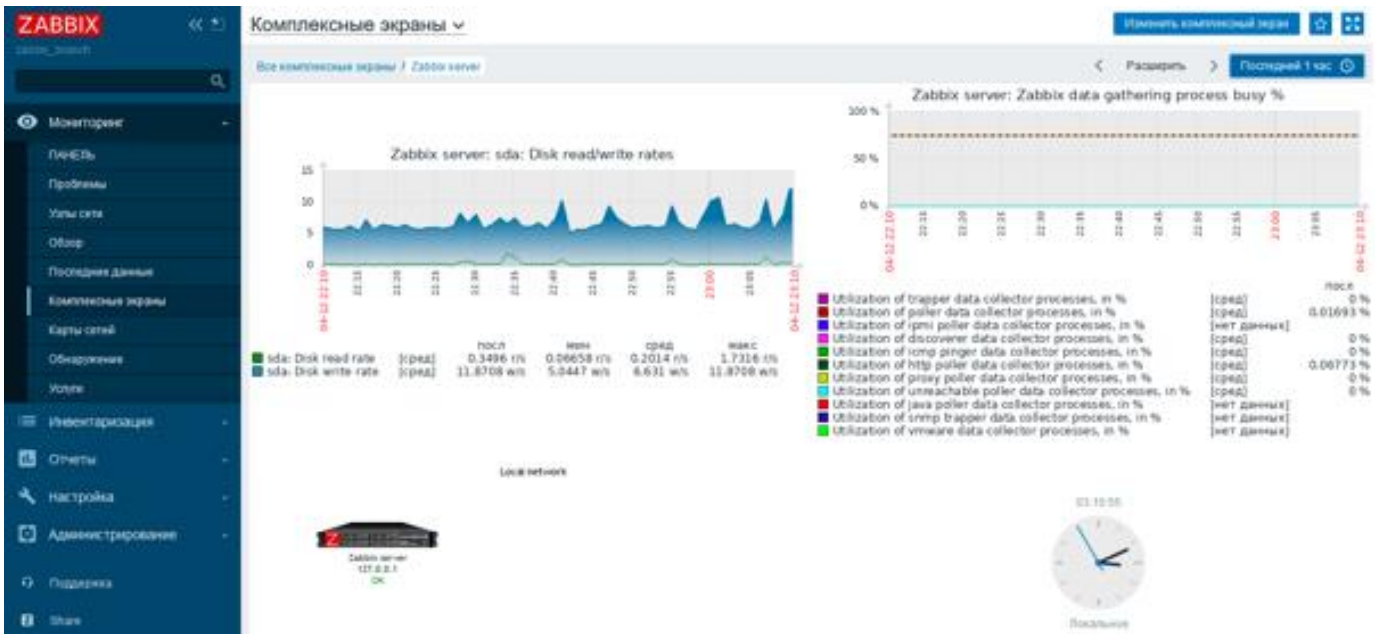

Рис. 3.31. Приклад комплексного екрану

Меню *Моніторинг* - *Карти мережі* відображає карти мережі для перегляду і редагування. Інструмент *Карти мережі* в якості головного засобу моніторингу часто використовує черговий персонал на підприємствах, де здійснюється цілодобовий моніторинг будь-яких об'єктів або процесів. Хоча, при добре продуманій системі збору різних даних з спостерігається мережі, найоб'єктивнішим інструментом моніторингу є, як не дивно, основна Панель (*Моніторинг* - *Панель*). На Картах мережі події відображаються неповноцінно, але карти зручні для візуалізацій топологій мережевого обладнання, датчиків розумних будинків, або рознесення обладнання по місцевості, з підкладкою картинки.

Меню *Моніторинг* - *Виявлення* відображає вузли мережі, додані в Zabbix за допомогою автоматичного виявлення.

Меню *Моніторинг* - *По*слуги візуалізує переднастроєні SLA (Service Level Agreement), якщо це потрібно.

Пункт меню *Інвентаризація* містить дві вкладки - *Огляд і Вузли мережі*. Цей розділ забезпечує можливість перегляду таких деталей інвентаризації вузлів мережі, як тип обладнання, так звані PID - Product ID (моделі пристроїв), встановлене програмне забезпечення, серійні номери, час

роботи. Інакше кажучи, це інструмент для відображення текстових даних, які можна витягти з вузлів мережі за допомогою Zabbix. У кожному розділі збору інвентарних даних буде приділятися достатньо часу, для отримання повноцінних відомостей про об'єкт моніторингу.

Меню *Звіти* є інструментом перегляду та аналізу зафіксованих системою Zabbix подій, з різними функціями вибірки, фільтрів і методів представлення даних. Крім того, в меню Звіти представлені розгорнуті звіти про дії, скоєних юзерами, і системою (наприклад, відправка повідомлень по електронній пошті).

Меню *Налаштування* - найважливіша ланка в системі Zabbix. Тут планується концепція моніторингу всього обладнання. У цьому меню зазвичай використовується тільки три основні вкладки - *Групи вузлів мережі, Шаблони, і Вузли мережі.*

Меню *Налаштування - Групи вузлів* мережі об'єднують в собі пристрої, в залежності від задачу, виходячи з трьох наступних принципів:

- за територіальною ознакою (всі пристрої в групі знаходяться в одній локації або офісі);

- за принципом однотипності пристроїв (серверне обладнання в одній групі, веб-сайти в інший, мережеві сервіси в третій, мережеве обладнання в четвертій, і т.п.);

- за принципом розмежування прав доступу (різним юзерам системи Zabbix призначаються різні права на перегляд або на перегляд і редагування тих чи інших груп).

При експлуатації Zabbix в великих корпоративних мережах було помічено, що пристрої краще об'єднувати в різні групи, керуючись всіма трьома принципами (за місцем розташування, за однотипності пристроїв, щодо розмежування прав доступу). Все залежить від конкретного задачу моніторингу. До речі, в головному меню *Моніторинг - Панель*, обладнання об'єднується в групи для перегляду, саме виходячи з приналежності об'єктів до тієї чи іншої Групі вузлів мережі.

Деякі шаблони можна застосовувати до будь-яких пристроїв. Шаблони ж, використовують SNMP, створюються для кожної групи однотипних мережевих пристроїв або ж для кожної окремої моделі мережевого пристрою (якщо на підприємстві використовується обладнання різних виробників), адже MIB-файли і OID для тих чи інших параметрів у таких пристроїв значно відрізняються.

Меню *Налаштування* - *Вузли мережі* дає можливість додавати, змінювати і видаляти об'єкти моніторингу.

Меню *Налаштування* - *Обслуговування* задає періоди обслуговування певних Вузлів мережі або Груп вузлів мережі. Даний пункт простий у використанні, але непопулярний на практиці.

Меню *Налаштування* - *Дії* дає можливість гнучко налаштовувати дії. Властивість даного інструменту полягає в тому, що дію можна створити в зв'язці з будь-якою подією, яке зафіксує Zabbix.

Меню *Налаштування* - *Кореляція подій* дає можливість встановлювати залежності між подіями, що відбуваються. Це допомагає зменшити кількість повідомлень про події, якщо одні події залежать від інших.

Меню *Налаштування* - *Виявлення* є інструментом автоматичного додавання нових вузлів мережі.

Налаштування - *Послуги* відповідає за налаштування SLA.

Меню *Адміністрування* - *Загальні* дає можливість конфігурувати параметри системи Zabbix (тему веб-інтерфейсу, додавання призначених для юзера зображень, регулярні вирази, макроси, перетворення значень та інше).

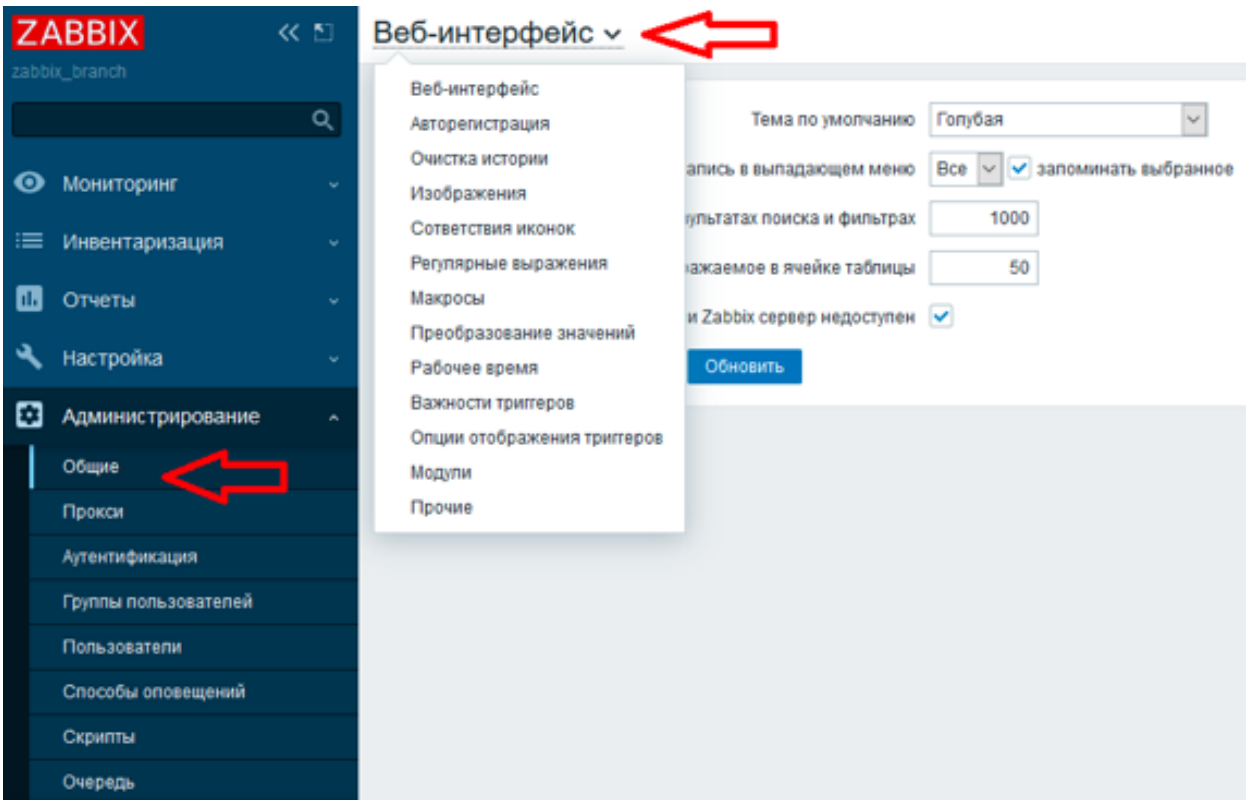

Рис. 3.32. Меню Адміністрування - Загальні

Меню *Адміністрування* - *Проксі* дає можливість організовувати моніторинг , коли кілька серверів Zabbix здійснюють моніторинг різних компонентів або процесів в мережі, а потім передають на головний сервер Zabbix. Це зменшує нагруження на головний сервер, здійснювати моніторинг «закритих» або недоступних для головного сервера компонентів мережі.

Меню *Адміністрування* - *Аутентифікація* дає можливість вибрати тип аутентифікації при вході юзера на сервер Zabbix:

- внутрішня аутентифікація має на увазі введення логіна і пароля, що зберігаються на сервері;

- LDAP-аутентифікація означає зовнішню аутентифікацію через сервіси Microsoft Active Directory або OpenLDAP;

- HTTP-аутентифікація означає аутентифікацію через веб-сервер Apache.

Далі необхідно розв’язати задачу з різними часовими поясами. Збудуємо елементарний bash-скрипт, який буде виводити час з поправкою на часовий пояс.

Zabbix має спеціальний каталог, в якому за замовчуванням зберігає скрипти. У файлі конфігурації Zabbix цей каталог вказано у змінній ExternalScripts:

ExternalScripts = / usr / lib / zabbix / externalscripts

За потреби його можна змінити. Створимо перший скрипт (Рис.3.33):

cd / usr / lib / zabbix / externalscripts

sudo nano universal_time.sh

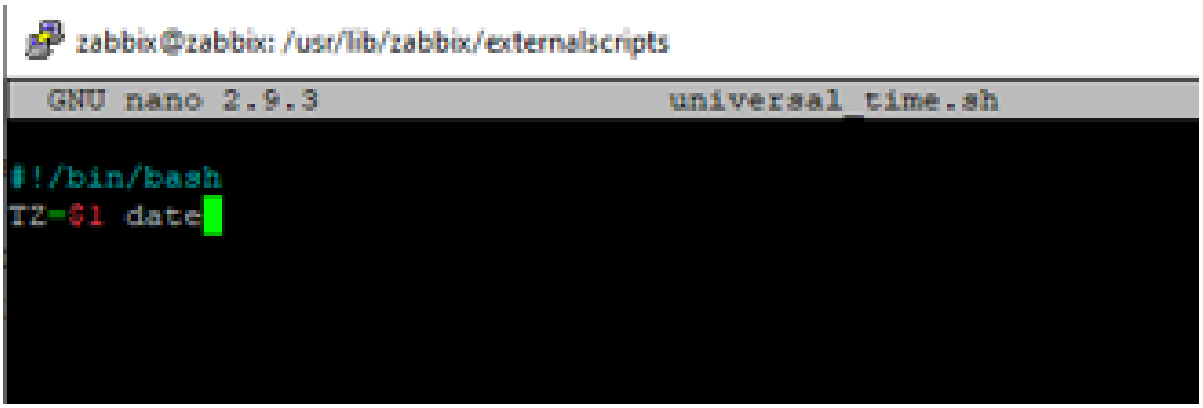

Рис. 3.33. Вміст скрипта виведення часу з поправкою на часовий пояс

Скрипт простий:

#! / Bin / bash - позначення використовуваної оболонки, bash;

TZ - Таймзона, це параметр утиліти date, що задає часовий пояс;

$ 1 - зовнішній агрумент скрипта, або змінна, яка буде змінною величиною;

date - утиліта для виведення часу в консоль;

Далі, зробимо скрипт виконуваним:

sudo chmod + x universal_time.sh

Після цього проаналізуємо його роботу (Рис. 3.34):

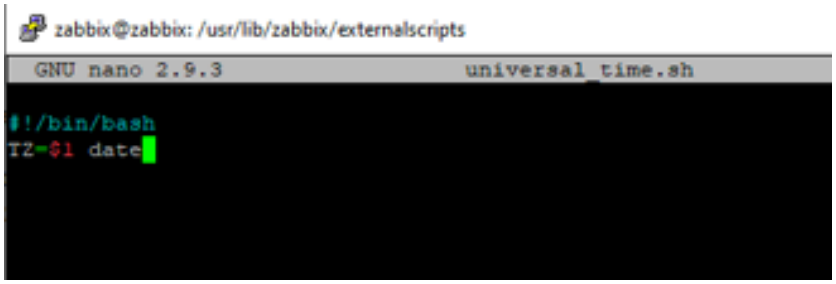

Рис. 3.34. Робота скрипта і утиліти date

Тепер створимо на існуючому вузлі мережі Zabbix_Server кілька елементів даних, для отримання значень часу в різних часових поясах. Я створюю групу елементів даних (інструмент угруповання елементів даних) з назвою «1. Time », щоб вона була вгорі після угруповання під назвою (Рис. 3.35):

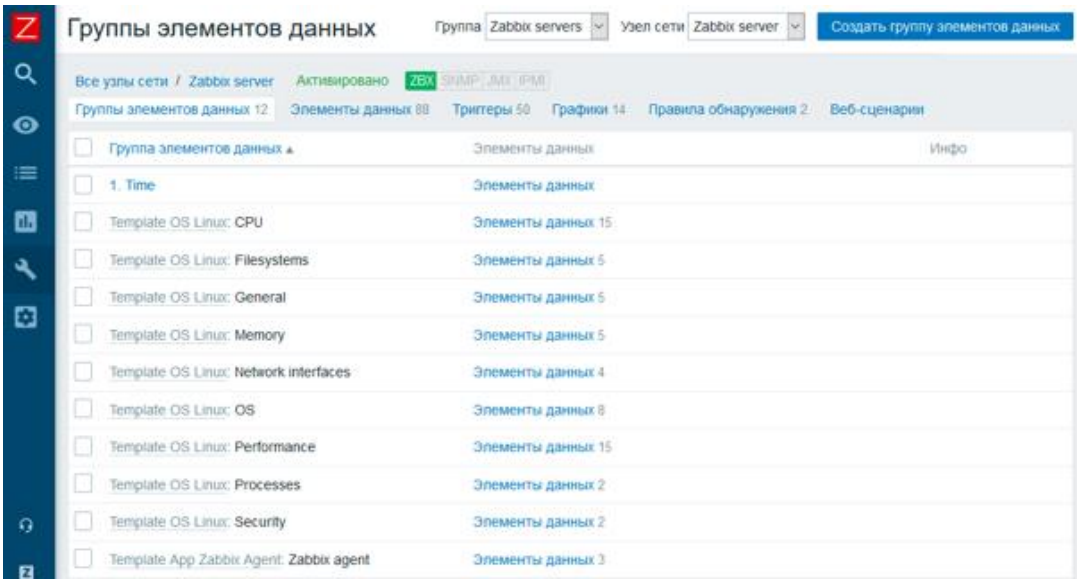

Рис. 3.35. Створення групи елементів даних

Після цього перейду до створення самих елементів даних, які і будуть запитувати у нашого сервера час з різними часовими поясами (Рис. 3.36):

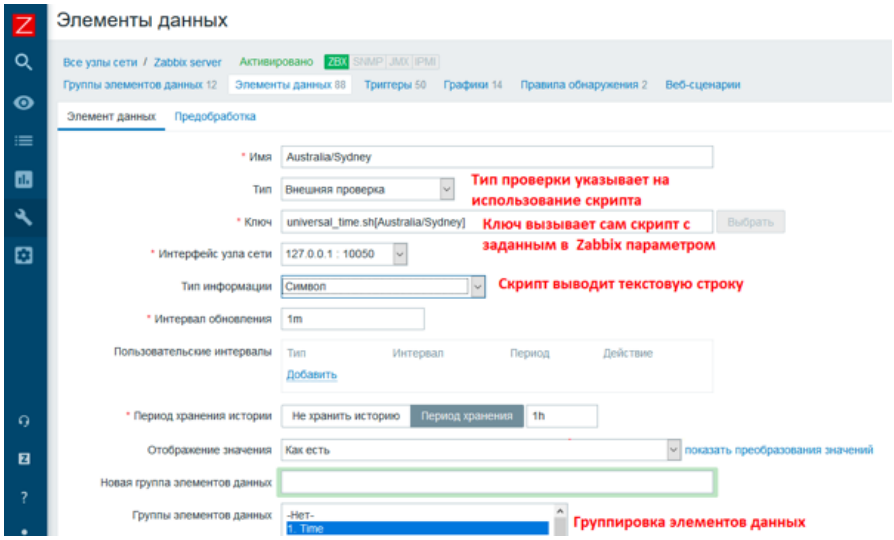

Рис. 3.36. Створення елемента даних

Для наочності створено інші елементи даних.

Обравши всі створені елементи даних, і вручну примусово запустили кнопкою «Виконати зараз». Потім перевіривши результат в останніх даних перейдемо до наступного кроку (Рис. 3.37):

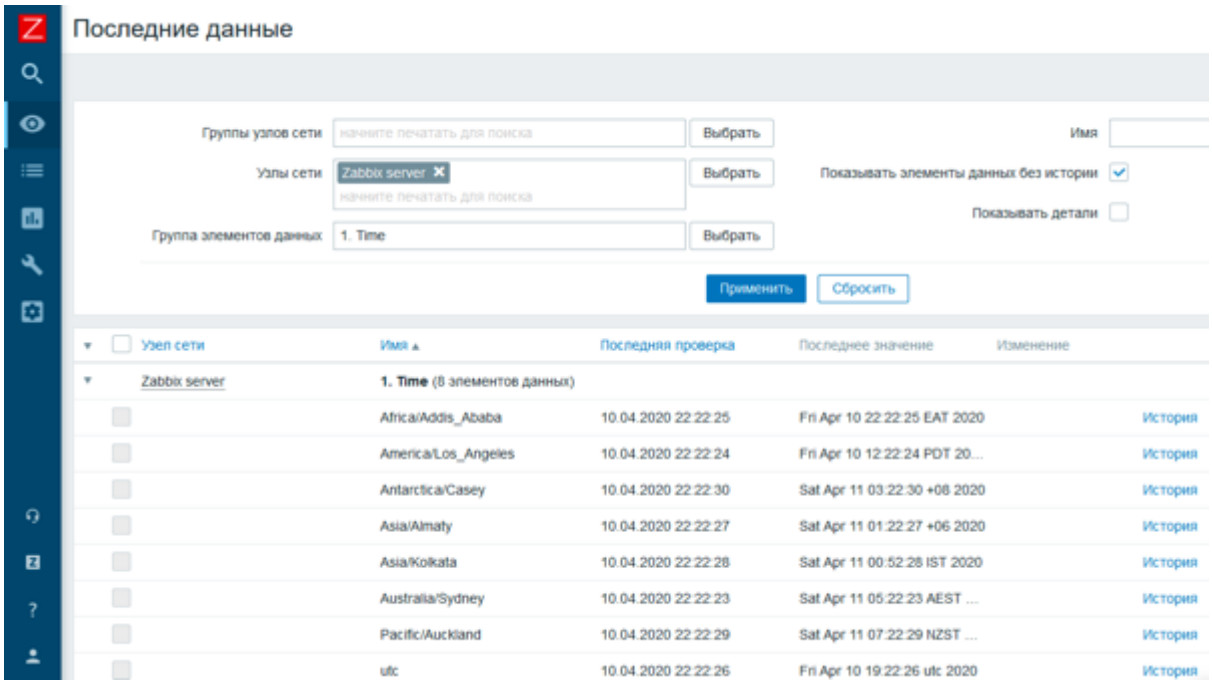

Рис. 3.37. Результат опитування в останніх даних

Тепер доповню основну панель моніторингу віджетами Простий текст і Годинник. Результат нижче (Рис. 3.38):

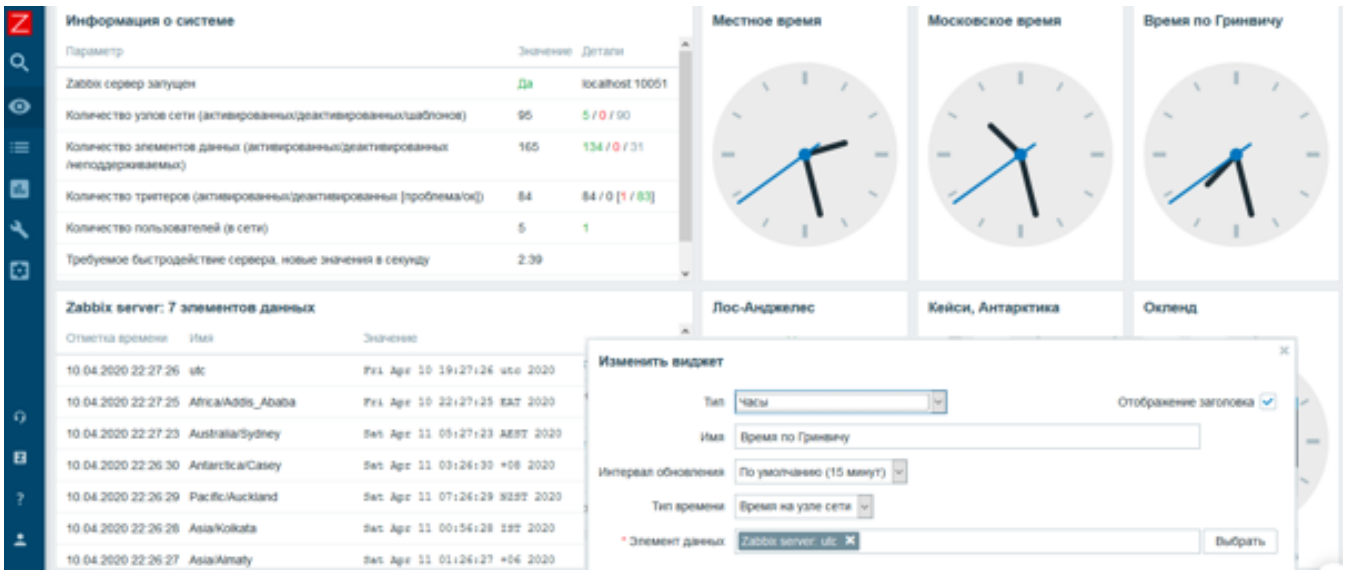

Рис. 3.38. Панель моніторингу

Отже, на цьому огляд системи закінчений. У наступному розділі приступимо безпосередньо до створення віртуальної машини.

3.4 Вигрузка віртуальної машини системи

Дія потрібні для вигрузки диску віртуальної машини доступні через стандартний провідник Windows. Якщо невідомо де знаходиться папка з віртуальної машиную, це можна побачити з Virtualbox (Рис.3.39)

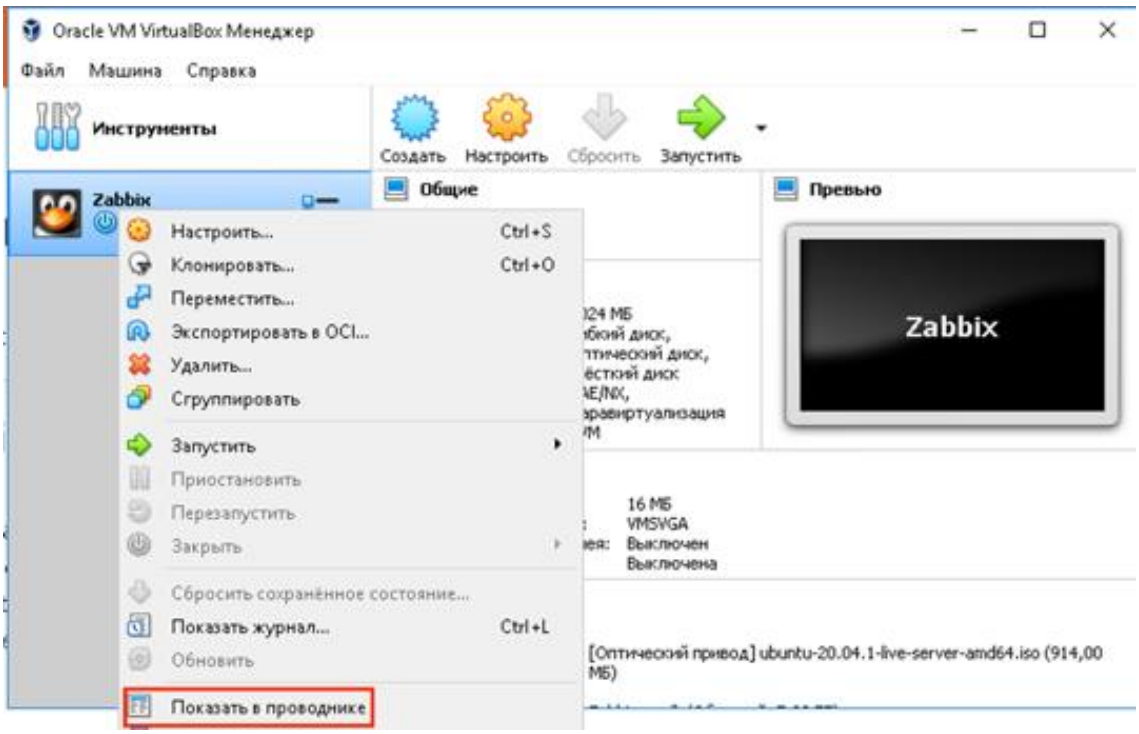

Рис. 3.39. Шлях до віртуальних дисків

Перейшовши до папки з віртуальними дисками, потрібний файл для розгортування в Вашій системі має вигляд .vmdk. (Рис.3.40)

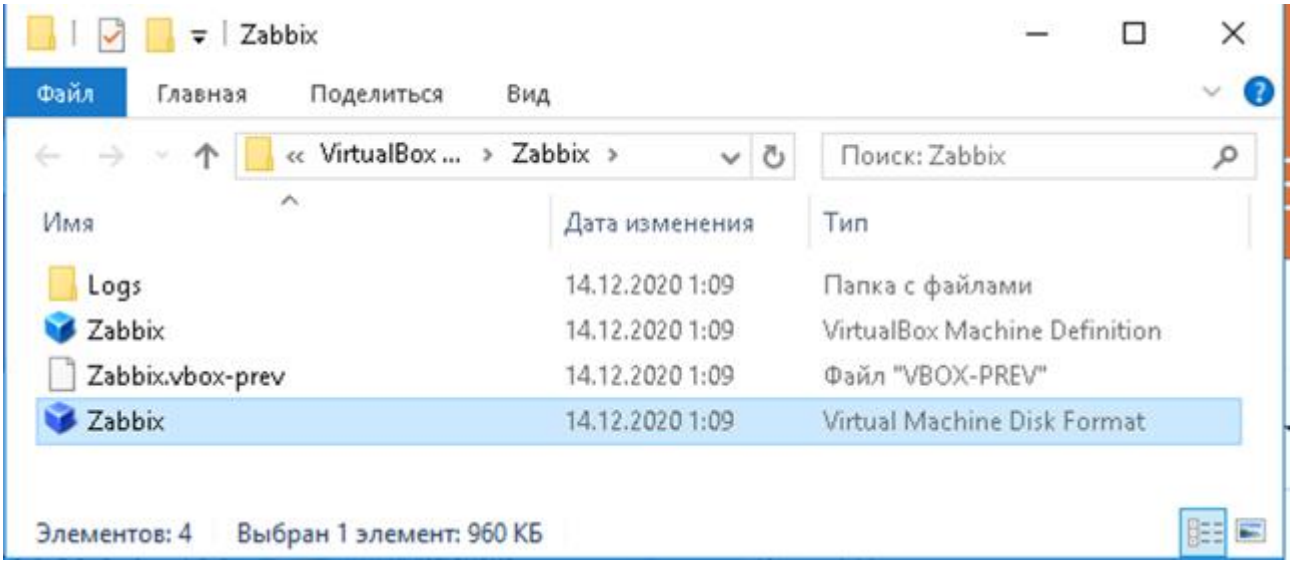

Рис. 3.40. Файл диску віртуальної машини

ВИСНОВКИ

Бізнес у будь-якій сфері сильно зав'язаний на доступність та працездатність його IT-інфраструктури 24/7/365. Щоб забезпечити цю працездатність, необхідно заздалегідь виявляти вузькі місця у конфігурації систем та мереж, а також швидко дізнаватися про наявність поломки та її причину. Для цих потреб у компаніях, де подібне стеження неможливе за рахунок лише фахівців, прийнято використовувати системи моніторингу. Системи моніторингу бувають платні і вільно розповсюджуються, а також розрізняються за своїм наповненням «з коробки», масштабованості, необхідним ресурсам та рівнем знань, необхідним для прийнятного налаштування. При виборі, розробці, впровадженні систем моніторингу спочатку потрібно визначитися з об'єктами, які будуть підлягати спостереженню, а також критичні події та показники, які визначать кількість сповіщень при поломці, частоту сканування та інші параметри та наслідки. Причому оцінювання показників насамперед потрібно здійснювати не з погляду технічного інженера, а з точки зору кінцевого користувача.

Хід виконання роботи повністю відповідав поставленим у запровадженні задачам. Для початку були висунуті загальні вимоги до моніторингу проектів, що підтримуються, володіючи списком яких можна було приступити до пошуку відповідного готового рішення. Потім докладно розглянуто різні існуючі системи моніторингу серверів від широкого різноманіття фірм-розробників, проаналізовано їх позитивні та негативні сторони, відповідність їх сформульованим вимогам. У результаті, зроблено висновок про неможливість чи неоптимальність використання вищерозглянутих існуючих систем.

У міру виникнення проблем та нових завдань, у проекті розроблялися різноманітні модулі, вівся пошук додаткових бібліотек та способів інтеграції з існуючими сервісами як для максимальної зручності написання скриптів, так і для відстеження специфічних показників.

Паралельно здійснювалося поетапне тестування створених скриптів та їхнє впровадження в експлуатацію. У результаті, розроблений моніторинг став невідривною частиною проектів, що підтримуються, що забезпечує високу доступність, відмовостійкість та інформацію про стан систем. У ході використання створеного моніторингу відділами адміністрування та підтримки серверів, розробки програмного забезпечення та підрозділом аналітиків було відзначено низку позитивних аспектів впливу на проект, а також несподівані плюси продукту:

- підвищення відмовостійкості сервісів, у тому числі шляхом інтеграції з ними;
- система дозволяє чітко визначати та планувати шляхи модернізації проекту та його інфраструктури, шукати «пляшкові шийки»;
- максимально швидко і точно інформує про можливу чи виниклу проблему;
- використання бази даних моніторингу для різних зовнішніх потреб (наприклад, звіти аналітичного відділу);
- делегування прав «з коробки»;
- простота інтеграції з різними зовнішніми бібліотеками та сервісами;
- автоматизація рутинних завдань системного адміністратора;

Аналіз телеметрії привів до потреби систем моніторингу в IT мережах. Також це дослідження представляє ключові вимоги систем моніторингу нового покоління. Ці системи повинні бути надійними, гнучкими та вимагати низьких експлуатаційних витрат для використання у виробництві. Крім того, я запропонував архітектуру системи, яка задовольняє ідентифікованому вимоги. Використання цієї архітектури допомагає розробити систему з реалізацією усіх вимог, включаючи швидкість та вартість володіння.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бараш Л. Моніторинг трафіку в сетях з комутацією пакетів [Трафік Моніторинг у мережах з комутацією пакетів]. Компьютерное обозрение , 2008, вип. 37 (654), С. 20-25.

2. Бройдо В. Л. Вычислительные системы, сети и телекоммуникации: Учебник для вузов / В.Л. Бройдо. - Спб.: Питер, 2003. - 688 с.

3. Гусева А.И. Работа в локальных сетях: Учебник / А. И. Гусева. - М.: Диалог - МИФИ, 2001. - 344 с.

4. Камалян А.К., Кулев С.А., Назаренко К.Н. и др. Компьютерные сети и средства защиты информации: Учебное пособие /Камалян А.К., Кулев С.А., Назаренко К.Н. и др. - Воронеж: ВГАУ, 2003.-119с.

5. Курносов А.П. Практикум по информатике/Под ред. Курносова А.П. Воронеж: ВГАУ, 2001.- 173 с.

6. Малышев Р.А. Локальные вычислительные сети: Учебное пособие/ РГАТА. - Рыбинск, 2005. - 83 с.

7. Новиков Ю. В. Локальные сети: архитектура, алгоритмы, проектирование. / Ю. В. Новиков. - М.: ЭКОМ, 2000. - 312 с.

8. Новиков Ю. В. Основы локальных сетей / Ю. В. Новиков. - М.: ЭКОМ, 2005. - 360 с.

9. Олифер В.Г, Олифер Н.А. Сетевые операционные системы/ В.Г. Олифер, Н.А. Олифер. - СПб.: Питер, 2002. - 544 с.: ил.

10. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы /В.Г. Олифер, Н.А. Олифер. - СПб.: Питер, 2002. - 672 с.

11. Флинт Д. Локальные сети ПК: принципы построения, реализация / Д. Флинт. - М.: Финансы и статистика, 2001. - 359 с.

12. Фридман А.Л. Основы объектно-ориентированной разработки программных систем. / А. Л. Фридман. - М.: Финансы и статистика, 2000. - 192 с.

13. Шафрин Ю.А. Основы компьютерной технологии / Ю.А. Шафрин. - М.: АБФ, 2001. - 560 с.

14. Яковлев В.А. Компьютерные сети / В.А. Яковлев. - М.: ИНФРА-М. 2001. - 244 с.

15. www.informika.ru - Интернет-учебник по информатике.

16. www.ito.edu.ru - Виртуальный университет информационных технологий.

17. www.ietf.org - Портал технических стандартов.

18. www.nagios.org - Сайт проекта Nagios.

19. www.debian.org - Сайт операционной системы Debian.

20. www.linux.org - Сайт ядра Linux.

21. www.habrahabr.ru - Русский блог-портал специалистов по информационным технологиям.

22. http://en.wikipedia.org/wiki/Wiki - Англоязычная свободная энциклопедия.

23. Храмцов П.Б.Информационные сети / Московская финансово-промышленная академия. - М., - 2004. - 290 с

24. Тулябаев Ф.А. Компьютерные сети и телекоммуникации 171 с.

25. Столлингс В. Современные компьютерные сети. 2-е изд СПб.: Питер, 2003. - 783 с.: ил.

26. Паркер Т., Сиян К. TCP/IP. Для профессионалов 3-е изд. СПб.: Питер, 2004. - 859 с.: ил.

27. Вишневский В.М. Теоретические основы проектирования компьютерных сетей Техносфера, 2003 г.

28. Casad J. Sams teach yourself TCP-IP in 24 hours

29. Hunt C. TCP-IP network administration

30. James Turnbull. Pro Nagios 2.0

31. Бейли Д., Райт Э. Волоконная оптика: теория и практика Волоконная оптика: теория и практика/Пер. с англ. - М: КУДИЦ-ПРЕСС, 2008 г. , 320 с.

32. Руководство пользователя. Коммутаторы локальных сетей D-Link. Учебное пособие 2004 г. 89 с.

33. Таненбаум Э. Компьютерные сети 4-е изд.

34. Feit S. TCP/IP: Architecture, Protocols, and Implementation with IPv6 and IP Security -- 2nd ed. Dr. Sidnie Feit Copyright 1997, 1993 All rights reserved Фейт С. TCP/IP: Архитектура, протоколы, реализация (включая IP версии 6 и IP Security) -- 2-е изд. Copyright © 1997, 1993 by The McGraw-Hill Companies, Inc. ISBN 0-07-021389-5 McGraw-Hill Издательство "Лори", 2000 , 450 с.

35. Уилсон Эд. Мониторинг и анализ сетей. Методы выявления неисправностей

36. ГОСТ 12.1.005-88 ССБТ. Общие санитарно-гигиенические требования к воздуху рабочей зоны.

37. СНиП 23.05-95. Естественное и искусственное освещение.

38. СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы.

39. СанПиН 2.2.2.542-96. Гигиенические требования к видеодисплейным терминалам, персональным ЭВМ и организации работы.

40. НПБ 105-03. «Пожарная безопасность».

41. СНиП 21-01-97. Пожарная безопасность зданий и сооружений.

42. ГОСТ 12.1.003-83 ССБТ. Шум. Общие требования безопасности.

43. ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности.

44. ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие требования.

45. http://ru.wikipedia.org/wiki/ - Русскоязычная свободная интернет-энциклопедия.

46. http://www.goural.ru/svregion/407 - Официальный туристический сайт Свердловской области

47. http://www.ugmk.com - Сайт компании УГМК-холдинг.

48. СН 2.2.4/2.1.8.562-96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки.

49. www.nix.ru - Сайт поставщика компьютерных комплектующих.

50. http://www.mprso.ru/a_2001.htm - Сайт министерства природных ресурсов Свердловской области.