

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ТЕЛЕКОМУНІКАЦІЙ**

Пояснювальна записка

до магістерської роботи

на тему: **“Методи оптимізації ресурсів обладнання шляхом
віртуалізації в системах хмарних сервісів”**

Виконав: студент 7 курсу, групи
АРЗМ-71
спеціальності

(шифр і назва спеціальності)

Дуборіз О.П.

(прізвище та ініціали)

Керівник _____

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

Нормоконтроль _____

Київ – 2021

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	8
ВСТУП.....	9
ОПТИМІЗАЦІЯ ВИТРАТ ПРИ ПОБУДОВІ ІНФОРМАЦІЙНО-	
1 ТЕЛЕКОМУНІКАЦІЙНИХ МЕРЕЖ З ВИКОРИСТАННЯМ	11
ВІРТУАЛІЗАЦІЇ.....	
1.1 Визначення віртуалізації.....	11
1.2 Типи віртуалізації у хмарних обчисленнях.....	14
1.2.1 Віртуалізація операційної системи.....	14
1.2.2 Віртуалізація обладнання.....	16
1.2.3 Віртуалізація робочих місць.....	19
1.3 Переваги віртуалізації.....	23
2 ВИКОРИСТАННЯ ХМАРНИХ СЕРВІСІВ ДЛЯ ЗБЕРІГАННЯ ТА	25
ДОСТУПУ ДО ДАНИХ.....	
2.1 Історія розвитку обчислень у хмарі.....	25
2.2 Сервісні моделі хмарних послуг.....	26
2.2.1 Все як послуга (XaaS).....	28
2.2.2 Програмне забезпечення як послуга (SaaS).....	30
2.2.3 Інфраструктура як послуга (IaaS).....	33
2.2.4 Платформа як послуга (PaaS).....	37
2.4 Компоненти архітектури хмарних обчислень.....	40
2.5 Моделі розгортання хмарних сервісів.....	42
2.5.1 Архітектура, структура та переваги публічної хмари.....	43
2.5.2 Особливості архітектури приватної хмари.....	45
2.5.3 Архітектура, переваги та особливості впровадження	47
гібридних хмар.....	
2.5.4 Впровадження громадських хмар.....	51
3 МЕТОДИ ПОВНОЇ ВІРТУАЛІЗАЦІЇ ТА ПАРАВІРТУАЛІЗАЦІЇ В	55
СИСТЕМАХ ХМАРНИХ СЕРВІСІВ.....	
3.1 Віртуалізація процесора.....	57
3.1.1 Метод віртуалізації за допомогою апаратного забезпечення...	58
3.1.2 Метод повної віртуалізації за допомогою бінарного	59
перекладу.....	
3.1.3 Метод паравіртуалізації за допомогою операційної системи...	60
3.2 Віртуалізація пам'яті.....	61
3.3 Віртуалізація пристроїв вводу-виводу.....	62
4 Застосування методу аналізу сумісності програмного забезпечення на	64
основі мережевого обладнання.....	
4.1 Аналіз проблем сумісності програмного забезпечення	65
4.2 Аналіз методів для вирішення проблем сумісності програмного	68
забезпечення	
ВИСНОВКИ.....	73
ПЕРЕЛІК ПОСИЛАНЬ.....	75
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ.....	78

ПЕРЕЛІК ТЕРМІНІВ ТА СКОРОЧЕНЬ

СУБД	система управління базами даних, набір взаємопов'язаних даних (база даних) і програм для доступу до цих даних
Тонкий клієнт	пристрій введення та відображення інформації (міжмережевий екран, брандмауер), загальна назва фізичних пристроїв чи програмних застосунків, для допуску, відмови, шифрування, пропуску трафіку згідно з набором правил безпеки
Фаєрвол	послуга надавання дискового простору для розміщення фізичної інформації на сервері, що постійно перебуває в мережі
Хостинг	Application Programming Interface, прикладний програмний інтерфейс
API	Domain Name System, система доменних імен, ієрархічна розподілена система перетворення імені комп'ютера або іншого мережевого пристрою в IP-адресу
DNS	Local area network, локальна комп'ютерна мережа
LAN	Translation lookaside buffer, буфер асоціативної трансляції, використовується для прискорення трансляції адреси віртуальної пам'яті на адресу фізичної пам'яті
TLB	Service level agreement, угода про рівень послуг
SLA	SQL Server Data Tools, сучасний засіб розробки, що дозволяє створювати реляційні бази даних
SSDT	Virtual Mashine, віртуальна машина, спеціальне програмне забезпечення, яке емулює роботу фізичної машини
VM	Virtual Machine Manager, монітор віртуальної машини, програмне забезпечення гіпервізора
VMM	Wide Area Networ, глобальна комп'ютерна мережа, використовується для поєднання мереж LAN та інших видів мереж
WAN	

ВСТУП

Найбільш звичним методом зберігання та доступу до необхідних програм, додатків та даних є зберігання на жорсткому диску комп'ютера. В наш час на зміну зберіганню даних на фізичному жорсткому диску приходять хмарні обчислення. Хмарні обчислення також називають «хмарою».

Хмарні обчислення (хмарні технології, хмара) - це метод обчислення, в якому програми та дані зберігаються в мережі Інтернет, свого роду оренда обчислювальних потужностей та інформаційного сховища на віддаленому сервері. Завдяки використанню такої технології можливо розмістити програми, додатки, платформи та навіть інформаційну інфраструктуру за межами офісу.

Хмарні технології активно використовуються приватними користувачами та користуються значним попитом в корпоративному секторі. Хмара вирішує багато ІТ потреб бізнесу та робить це з притаманною їй ефективністю: правильне використання ресурсів, високі стандарти безпеки, гнучка система тарифікації, глибока інтеграція з власними ІТ системами, зручні інтерфейси для керування, і допоміжні сервіси для використання ІТ інфраструктури. Останніми роками намітилася тенденція на використання хмарних сервісів навіть українськими державними підприємствами та органами влади.

Наразі тема оптимізації ресурсів обладнання шляхом віртуалізації в системах хмарних сервісів є надзвичайно актуальною. Пандемія та наслідки, до яких вона призвела, підвищили терміновість, з якою підприємства трансформують організацію своїх ІТ потужностей і хмарні обчислення стали ключовим фактором, що сприяє цьому. Хмарні технології набирають все більшу кількість прихильників і стають важливим ІТ інструментом для бізнесу. Використання хмарних технологій викликає значний інтерес серед фахівців, оскільки дозволяє підвищувати гнучкість ІТ інфраструктури та суттєво економити кошти на її утримання.

Метою даної магістерської роботи є дослідження методу для аналізу на основі віртуалізації існуючих проблем сумісності програмного забезпечення, які мають негативний вплив на роботу інформаційної системи. В результаті дослідження та оцінки наявності потенційного конфлікту між застосунками внаслідок впливу сумісності програмного забезпечення було отримано комплекс рішень щодо попередження потенційних проблем.

1 ОПТИМІЗАЦІЯ ВИТРАТ ПРИ ПОБУДОВІ ІНФОРМАЦІЙНО-ТЕЛЕКОМУНІКАЦІЙНИХ МЕРЕЖ З ВИКОРИСТАННЯМ ВІРТУАЛІЗАЦІЇ

1.1 Визначення віртуалізації

Віртуалізація – це сучасна технологія надання моделі ресурсу, замість самого фізичного ресурсу, це представлення певних обчислювальних ресурсів або їх логічного об'єднання, який надає переваги над оригінальною конструкцією, створення віртуального, тобто штучного, об'єкту чи середовища, процес запуску віртуальної комп'ютерної системи у середовищі, абстрагованому від фактичного обладнання (Рис. 1.1) [1]. Зазвичай мається на увазі одночасний запуск декількох операційних систем на комп'ютерній системі. Для програм та додатків, що працюють на віртуалізованому комп'ютері, імітується перебування на власному спеціалізованому комп'ютері, на якому програмне забезпечення, бібліотеки, операційна система являються унікальними для віртуалізованої гостьової системи та не підключені до операційної системи хоста, що знаходиться під ним. Віртуалізація абстрагує програмне забезпечення від апаратної частини.



Рис 1.1 Віртуалізація – перехід від фізичного серверу до логічного

В основі віртуалізації лежить можливість одного комп'ютера виконувати роботу кількох комп'ютерів завдяки розподілу його ресурсів на декілька середовищ. За допомогою віртуальних серверів і віртуальних настільних комп'ютерів можна розмістити кілька ОС і кілька додатків в єдиному розташування. Таким чином, фізичні та географічні обмеження перестають мати якусь значення. Крім енергозбереження та скорочення витрат завдяки більш ефективному використанню апаратних ресурсів, віртуальна інфраструктура забезпечує високий рівень доступності ресурсів, більш ефективну систему управління, підвищену безпеку і вдосконалену систему відновлення в критичних ситуаціях. Сучасні концепції віртуалізації надають можливість віртуалізувати будь-яке робоче середовище й надіслати його куди завгодно: настільні комп'ютери, ноутбуки, планшети, мобільні, сервери в дата центрах тощо, тобто розмножити та емулювати процес.

Серед причин популярності віртуалізації в обчислювальних технологіях є використання користувачами персональних комп'ютерів можливості запускати програмне забезпечення, призначене для інших операційних систем, без перемикання комп'ютера чи перезавантаження в іншу систему.

Адміністраторам серверів віртуалізація в обчислювальних технологіях надає можливість запускати різні операційні системи, та пропонує спосіб сегментації великої системи на певну кількість менших частин, що дозволяє максимально ефективно використовувати сервер різними користувачами чи програмам з різними призначеннями та ресурсними потребами. За допомогою віртуалізації можна для безпеки ізолювати програми, які працюють у віртуальній машині від процесів, що відбуваються в іншій віртуальній машині на тому ж хості.

Гіпервізор нерозривно пов'язаний віртуалізацією. Це програмне забезпечення, яке розгортається на сервері та безпосередньо взаємодіє з його

фізичними ресурсами. Гіпервізор відповідає за те, щоб віртуальні машини «бачили» ці ресурси як власні.

Гіпервізор виконує віртуалізацію, забезпечує створення віртуальних машин та їх функціонування, дозволяє віртуалізувати системні ресурси та забезпечує паралельну одночасну роботу кількох операційних систем на одному комп'ютері на яких виконується власна ОС. Гіпервізори умовно діляться на два типи: перший тип (рідні англ, native) здійснює управління віртуальними машинами на стороні апаратного забезпечення, діє як ОС. Перший тип забезпечує максимальну продуктивність та надійність, оскільки працює незалежно від ОС. Другий тип гіпервізорів (хостовий) працює як традиційні програми, його можна запускати і зупиняти, як звичайну програму. Другий тип забезпечує гіршу продуктивність в порівнянні з першим.

Віртуальна машина - це створений еквівалент апаратного забезпечення, на якому працює одна чи кілька інших операційних систем. Віртуальні машини мають доступ до обчислювальних ресурсів за допомогою апаратного забезпечення і одночасно обмежений доступ до процесора та пам'яті хост-машини, один чи кілька фізичних або віртуальних дискових пристроїв для зберігання, віртуальний або фізичний мережевий інтерфейс, USB-пристрої, відеокарти, якими надається спільний доступ до віртуальної машини. Віртуальні машини поділяються на два основні типи: віртуальні машини, що працюють безпосередньо на реальному обладнанні та віртуальні машини, які встановлені як прикладна програма над існуючою ОС (рис 1.2).

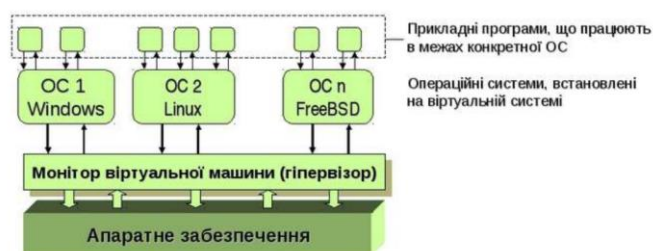


Рис.1.2 Типи віртуальних машин

1.2. Типи віртуалізації в хмарних обчисленнях

Існує кілька типів віртуалізації: віртуалізація операційної системи, віртуалізація сервера додатків, віртуалізація додатків, адміністративна віртуалізація, віртуалізація робочих місць, віртуалізація мережі, віртуалізація обладнання, віртуалізація сховища, віртуалізація інфраструктури.

Розглянемо деякі з них:

1.2.1. Віртуалізація операційної системи.

Віртуалізація операційної системи, являється частиною технології віртуалізації та різновидом віртуалізації сервера. Цей тип включає установку другого або кількох екземплярів операційної системи, наприклад Windows, на одному комп'ютері. Це дає підприємствам можливість зменшити кількість фізичного обладнання, необхідного для роботи їх програмного забезпечення, за рахунок скорочення кількості реальних машин та дозволяє компаніям заощадити на енергії, кабелях, обладнанні, просторі в стійках, при цьому дозволяючи запускати таку ж кількість додатків.

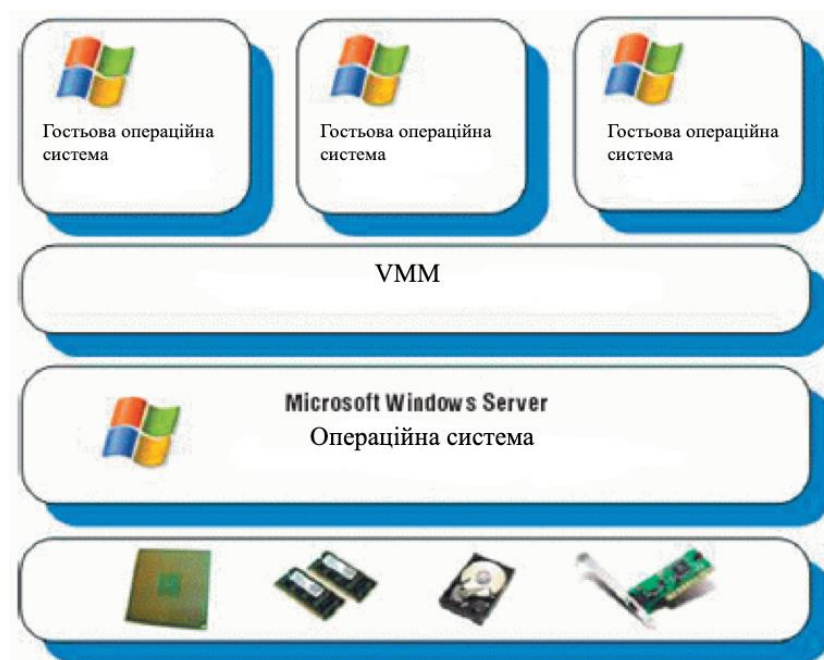


Рис 1.3 Віртуалізація операційної системи - розміщення кількох ОС на власній ОС

Операційна система комп'ютера керує усім програмним та апаратним забезпеченням комп'ютера. Операційною системою одночасно можуть запускатися кілька різних комп'ютерних програм. Це робиться за допомогою центрального процесора комп'ютера. Поєднана робота кількох компонентів комп'ютера координується операційною системою.

При віртуалізації операційної системи додатки не заважають один одному, хоча вони працюють в одному комп'ютері. Ядро операційної системи дозволяє існувати більше ніж одному ізольованому екземпляру простору користувача. Це контейнерами програмного забезпечення, вони є двигунами віртуалізації. Віртуалізація дозволяє розділити кілька додатків на контейнери. Віртуалізація операційної системи найпоширеніша у хмарних обчисленнях. Віртуалізація операційної системи застосовується для інтеграції апаратного забезпечення сервера шляхом переміщення служб на окремих серверах, що забезпечує безпеку апаратних ресурсів. Віртуалізація операційної системи використовується для віртуального середовища хостингу.

Існує два типи віртуалізації операційних систем: віртуалізація ОС Linux та віртуалізація ОС Windows. Для віртуалізації систем Linux використовується програмне забезпечення VMware Workstation. В обох типах для встановлення будь-якого програмного забезпечення за допомогою віртуалізації необхідно спочатку встановити програмне забезпечення VMware.

У віртуалізації операційної системи є два типи віртуального диска для підключення через мережу до віртуального диска:

- приватний диск, використовується одним клієнтом або одним підприємством. На цьому диску окремий клієнт чи підприємство може зберігати інформацію на основі призначених можливостей.

- спільний диск, використовується кількома клієнтами одночасно. Зміни, здійснені клієнтами, застосовуються індивідуально і не впливають на налаштування іншого клієнта під час перезавантаження системи.

Переваги віртуалізації операційної системи:

- ІТ середовище в традиційному розгортанні потребує завантаження кожної машини індивідуально. Віртуалізація операційної системи забезпечує можливість гнучкого та швидкого розгортання додатків і хмарних рішень за лічені хвилини;
- віртуалізація операційної системи значно зменшує використання фізичного простору, який використовує ІТ система, що призводить до зменшення обсягу технічного обслуговування та, відповідно, заощаджує матеріальні, людські та часові ресурси;
- віртуалізація надає можливість підприємствам підвищити ефективність використання серверного обладнання, підвищуючи таким чином рентабельність експлуатаційних робіт.

Віртуалізації операційної системи використовує програмне забезпечення, що дозволяє апаратному забезпеченню системи одночасно запускати кілька операційних систем. Віртуалізація операційної системи дозволяє забезпечити безпеку та розташування кінцевих апаратних ресурсів для великої кількості користувачів.

У віртуалізації операційної системи ядро працює з єдиною операційною системою і забезпечує її можливістю копіювання на кожній з ізольованих платформ. Віртуалізація ОС економічна, надійна та гнучка, тож перевагами віртуалізації операційної системи користуються як великі підприємства, так і невеликі компанії.

1.2.2. Серверна віртуалізація

Віртуалізація обладнання (серверна віртуалізація) абстрагує операційну систему та програми від рівня фізичного обладнання, яке використовує ці ресурси. Це архітектура програмного забезпечення, яке відповідає, щоб кілька операційних систем працювали на базі одного фізичного сервера. Завдяки серверній віртуалізації на одному фізичному сервері можна завантажувати декілька операційних систем у вигляді

віртуальних машин, у кожній з яких є доступ до обчислювальних ресурсів сервера.

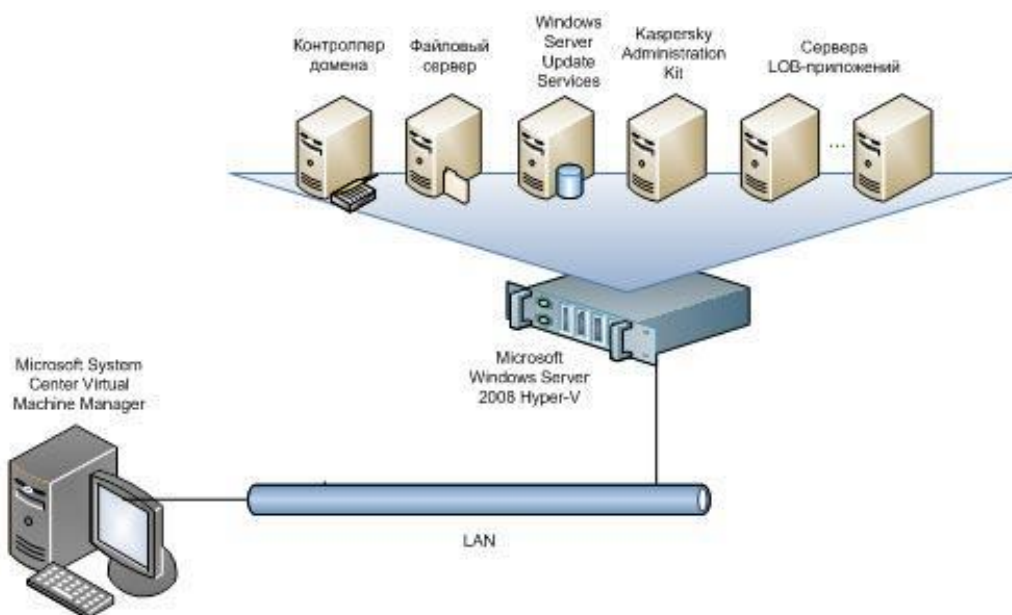


Рис 1.4 Схема організації віртуальних серверів

При цьому ПЗ кожного сервера самодостатнє та відокремлене від будь-яких фізичних пристроїв. Це ж ПЗ сприймає доступні ресурси як ресурси одного фізичного сервера, хоча по факту отримує невеликий пул ресурсів. Віртуальні сервери працюють як імітація фізичного обчислювального обладнання. Сервер знайомий з окремими фізичними серверами. Єдиний фізичний сервер ділиться на кілька ізольованих віртуальних серверів за допомогою програмного забезпечення (Рис. 1.4).

Віртуалізація середовища підвищує гнучкість і адаптивність ІТ-інфраструктури, знижує витрати на її утримання, робить робочі навантаження мобільними, а ресурси - доступними. Наслідками віртуалізації серверів буде зростання автоматизації бізнес-процесів, поліпшення керованості і економічності інфраструктури, мінімізація аварійних і призначених для технічного обслуговування простой.

У віртуальному сервері емулюється апаратне забезпечення: процесори, дискові накопичувачі, оперативна пам'ять. Операційні системи, встановлені на кожен віртуальний сервер, не "бачать" один одного і при цьому

функціонують так, як вони були б встановлені на звичайному комп'ютері. На одному «залізі» можна запускати кілька ОС та в різних пропорціях розподіляти між ними фізичні ресурси.



Рис. 1.5 Графічне зображення віртуалізації сервера

Віртуалізація серверів вирішує відразу кілька важливих завдань:

- дозволяє оптимізувати споживання обчислювальних ресурсів та ресурсів зберігання. До появи технології віртуалізації в дата-центрах накопичувалось обладнання, яке використовувалося неефективно. Поки одні машини працювали в половину (або менше) свого потенціалу, інші були перевантажені завданнями і часто зупинялися. Віртуалізація серверів дозволяє рівномірно розподілити робочі навантаження між машинами.
- стримує зростання кількості серверів, дозволяючи зменшити кількість серверів шляхом встановлення замість них одну машину і запускання на ній потрібне число ОС, наприклад, сімейства Windows.
- знижує експлуатаційні витрати на утримання фізичного обладнання. Так як кількість серверів менше, підприємство може заощадити на енергоспоживанні і кондиціонуванні приміщень (знизиться тепловиділення, тому можна буде використовувати менш потужні установки).
- спрощує міграцію даних. Під час передавання даних на інший сервер скорочується час виконання організаційних робіт: ІТ фахівцям досить

оновити драйвери на основний (хостовій) ОС, в той час гостьові ОС продовжать працювати в попередньому режимі, так як вони не прив'язані до фізичного обладнання. Для користувачів, які використовують ресурси віртуальних машин, такий «переїзд» залишиться непоміченим.

- підвищує продуктивність прикладного ПЗ. Якщо зазвичай одні додатки на 100% споживали фізичні ресурси однієї машини, при віртуалізації віртуальні машини, що працюють можуть автоматично переміщатися на менш навантажені сервери.

- скорочує простої обладнання. Менше часу потрібно для відновлення системи до початкового стану в разі збою. Віртуальні сервери підтримують технологію створення віртуальних знімків і мають можливість робити резервне копіювання даних за заздалегідь складеним розкладом.

- спрощує роботу з віртуальним середовищем. Потрібно менше технічних фахівців для обслуговування системи.

- дозволяє дистанційно керувати віртуальними серверами незалежно від їх кількості та територіального розташування. Наприклад, якщо фізична машина «зависла», не потрібно йти в серверну перезавантажувати її, це можна зробити з консолі безпосередньо з робочого місця адміністратора.

1.2.3 Віртуалізація робочих місць

Традиційний підхід до використання ІТ-сервісів в організації має на увазі їх розміщення на високонадійних і захищених серверах в ЦОД і доступ до них з ПК користувачів. Найчастіше саме призначений для користувача ПК виявляється найслабшою ланкою. Організація робочих місць користувачів на сьогоднішній день має безліч недоліків - потреба в частому оновленні парку комп'ютерів, високі фінансові витрати на закупку та утримання в разі закупки потужних екземплярів і т.д. Віртуалізація робочих місць дозволяє створити стабільну і надійну інфраструктуру зберігання і доставки робочих станцій на основі хост-серверів, що обслуговуються

централізовано. Це значно скорочує затрати і підвищує безпеку. Віртуалізація робочих місць - це загальне поняття, що означає будь-яку технологію, яка відокремлює середовище віртуальних робочих місць від обладнання, яке використовується для доступу до неї. Віртуалізацію робочих місць можна впровадити різними способами. Розглянемо один зі способів віртуалізації робочих місць - Virtual Desktop Infrastructure (VDI).

Віртуалізація робочих місць VDI дозволяє централізовано зберігати і обслуговувати додатки і дані будь-якої кількості ПК. Доступ до віртуальних робочих станцій здійснюється з клієнтських пристроїв з будь-якої точки земної кулі, де є Інтернет або через звичайну мережу. VDI - одна з найсучасніших технологій віртуалізації персональних робочих станцій, це розробка компанії VMware. Термін VDI був введений компанією VMware та згодом став аббревіатурою цілої технології. Замість встановлення кожному користувачеві ПЗ на комп'ютері для кожного користувача адміністратором створюється автоматично із шаблону його віртуальна машина, у вигляді образу робочого столу.

VDI - це програмно-апаратний комплекс, який працює таким чином: є сервер з серверної ОС, на якому створені та функціонують "образи" з клієнтськими операційними системами. Частина файлів у цих образах загальна, частина - відокремлена, але суть полягає в тому, що у кожного з користувачів - свій образ операційної системи, а доступ до нього здійснюється, як правило, через тонкі клієнти або ПК будь-якої конфігурації. На тонкому клієнті чи ПК завантажена спеціальна версія Windows (з приставкою Embedded), яка забезпечує підключення до сервера і роботу з "образом ОС" що зберігається на сервері, або дуже маленька операційна система, єдиним призначенням якої є завантаження "заліза" і базових драйверів, пошук і підключення до сервера і завантаження віртуального робочого столу на ваш монітор.

В інфраструктурі VDI гіпервізор розділяє сервери на віртуальні машини, що містять віртуальні робочі місця, до яких користувачі

отримують віддалений доступ з своїх пристроїв. Користувачі можуть отримувати доступ до віртуальних робочих місць з будь-якого пристрою і в будь-якій точці, при цьому всі процеси обробки даних виконуються на сервері. Користувачі підключаються до віртуальних робочих місць за допомогою брокера підключень, який є програмним шлюзом і виступає в якості посередника між користувачем і сервером. У VDI всі персональні комп'ютери консолідовані у віртуальному середовищі, частіше всього сервері. Доставляється по мережі на пристрій кінцевої точки де працює користувач. Кінцевою точкою може бути традиційний ПК, планшет або мобільний пристрій. Віртуальний комп'ютер дозволяє користувачеві взаємодіяти з операційною системою та її програмами так, ніби користувач працює локально за власним девайсом. Протоколи підключення дозволяють працювати навіть з «важкими» графічними додатками без видимих затримок (рис. 4).

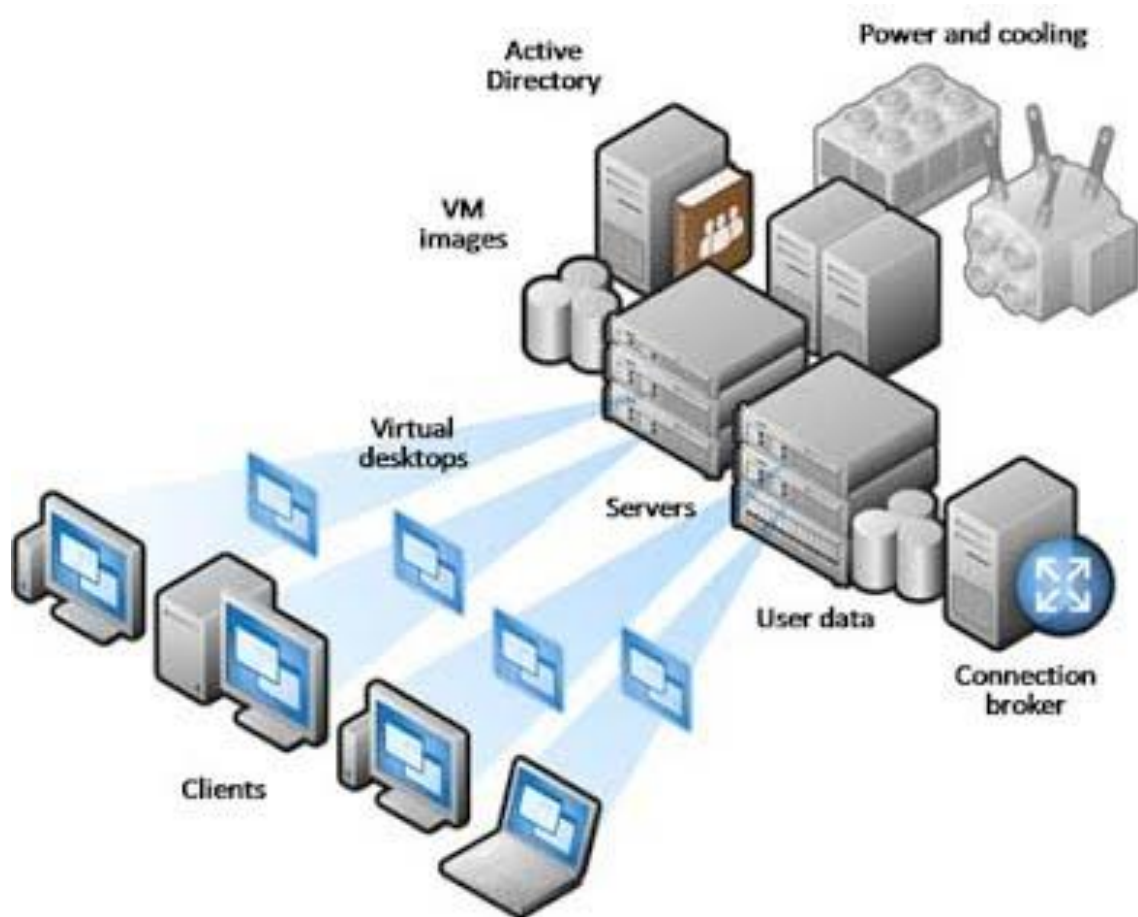


Рис. 1.6. Віртуалізація робочих місць

Переваги використання VDI:

- стандартизація: VDI забезпечує централізований, стандартизований робочий стіл працівникам. Незважаючи на те, що користувач отримує доступ до VDI з ноутбука, планшета, робочої станції або мобільного пристрою, користувальницький досвід точно такий же, без необхідності акліматизуватись на будь-якій фізичній платформі;

- мобільність: користувачеві завжди доступне його звичне робоче місце, повноцінне, абсолютно таке ж, як в офісі без наявності робочого ноутбука, можливість працювати з будь якого місця в будь який час;

- масштабованість: середовище VDI можна швидко розширити, якщо організація, наприклад, розширюється, дозволяючи новим працівникам отримати доступ до робочого віртуального робочого столу підприємства та відповідних додатків;

- полегшення підтримки віддалених та мобільних користувачів: надаючи віртуальний робочий стіл цим категоріям працівників, для них забезпечується можливість такої ж ефективної роботи, як в офісі.

- забезпечення користувача доступом до спеціальних корпоративних додатків, наприклад електронної пошти, загальних дисків, загальних файлів, загального календаря.

- продовження терміну експлуатації застарілих комп'ютерів. Оскільки в кінцевій точці відбувається мало фактичних обчислень, адміністратори можуть зробити застарілі ПК кінцевими точками VDI. При необхідності закупівлі нових ПК, підприємства можуть придбати менш потужні обчислювальні пристрої для кінцевих користувачів.

1.3. Переваги віртуалізації

Перевагами віртуалізації являються:

- збільшення коефіцієнту використання апаратного забезпечення. При виконанні повсякденних завдань більшість серверів завантажені на 5-20 відсотків. Використання кількох віртуальних серверів на одному фізичному

дозволить збільшити його завантаженість до 80 %, забезпечивши при цьому істотне скорочення фінансових вкладень на придбання апаратного забезпечення;

- зменшення витрат на заміну апаратного забезпечення. Віртуалізація дозволяє зменшити кількість серверів та пов'язаного з ними ІТ обладнання. Оскільки віртуальні сервери не прив'язані до конкретного обладнання, при оновленні парку фізичних серверів не потрібна повторна установка та налаштування програмного забезпечення. Віртуальна машина може бути просто скопійована на інший сервер;

- оптимізація витрат на придбання програмного забезпечення. Виробниками програмного забезпечення введено спеціальні умови на придбання ліцензованого програмного забезпечення для використання у системах віртуалізації. Наприклад, купуючи одну ліцензію на Microsoft Windows Server 2008 Enterprise, покупець отримує право одночасного її використання на одному фізичному сервері та чотирьох віртуальних в межах одного серверу, а Windows Server 2008 Datacenter ліцензується тільки на кількість процесорів і може використовуватися одночасно на необмеженій кількості віртуальних серверів;

- підвищення гнучкості використання віртуальних серверів та швидкості реагування системи. У випадку використання кількох серверів при навантаженні, що змінюється, віртуальні сервери є оптимальним рішенням, оскільки вони можуть бути безболісно перенесені на інші платформи, коли фізичний сервер працює з підвищеним навантаженням;

- забезпечення високої доступності та безперервності в роботі. Копіювання віртуальних машин, їх відновлення з резервних копій, оновлення критичних серверів займає значно менше часу. При виході з ладу обладнання, резервна копія віртуального сервера може бути відразу запущена на іншому фізичному сервері;

- підвищення керованості серверної інфраструктури. Нові методи керування віртуальною інфраструктурою дозволяють централізовано

управляти віртуальними серверами та забезпечувати налаштування, ініціацію, пріорітезацію задач, розподіл ресурсів, гнучкий розподіл трафіку, технічне обслуговування;

- зменшення витрат на електроенергію. Для великих датацентрів, де витрати на підтримку парку серверів включають значні витрати на електроенергію (живлення, системи охолодження), концентрація кількох віртуальних серверів на одному фізичному зменшує їх.

2 ВИКОРИСТАННЯ ХМАРНИХ СЕРВІСІВ ДЛЯ ЗБЕРІГАННЯ ТА ДОСТУПУ ДО ДАНИХ

2.1 Історія розвитку обчислень у хмарі

Концепція хмарних обчислень з'явилася відносно недавно, на сьогодні це найгучніший тренд на ринку ІТ, але під цією назвою визначається концепція розподілу часу з урахуванням сучасного розвитку технічних засобів. Хмарні обчислення – це не революція, а новий еволюційний виток спіралі розвитку ІТ індустрії. Завдяки принципу використання з розподілом часу можна домогтися більшої ефективності використання ресурсів. Ідея доступу до ресурсів за розподілом часу існувала від самого початку – з часів перших мейнфреймів, коли час роботи з комп'ютером був розподілений за графіком. Під його керівництвом американського вченого Джона Маккарті, засновника ідеї комп'ютерного розподілу часу, вперше в світі в 1954 році було розроблено систему протиповітряної оборони наземного середовища SAGE, яка дозволила декільком користувачам одночасно отримувати доступ до системи. Інший американський вчений Лестер Дональд Ернест висловлювався: "Без поділу часу, не було б сучасного Інтернету".

Американський вчений Джозеф Карл Робнетт Ліклайдер один із засновників мережі ARPANET, у своїй публікації «Міжгалактична Комп'ютерна мережа» в 1958 році висловлював ідею: «У майбутньому я зможу користуватися певними мережевими функціями, здійснюючи вибірку потрібних мені даних за допомогою системи, яка підбере необхідні мені програми. Для цього вона буде використовувати запропоновані їй описи, які з часом можна буде робити природною мовою. Між запозиченими програмами і моїми власними можна буде встановлювати зв'язок ... виконання завдань може відбуватися де завгодно». Отже ідея хмарних обчислень була запропонована, в той час, коли ще не було достатньо технічних засобів для її втілення.

Першим кроком до втілення хмарних обчислень можна вважати появу ASP (Application service provider - провайдера послуг доступу до додатків) у другій половині 1990-х років. ASP можна вважати одними із перших SaaS (програмне забезпечення як послуга) сервісів. Пальма першості належить сервісу електронної пошти від компанії Hotmail. Але відсутність на той час широких каналів інтернету та технологій віртуалізації стали на перепоні - за відсутності швидких та стабільних каналів інтернет користувачі не могли отримати якісні послуги, а без технологій віртуалізації неможливо було ефективно та гнучко розподіляти ресурси та масштабувати сервіси. Слід зазначити що стрімкий ріст інтернет користувачів у 2000-х роках, сформував попит на послуги SaaS.

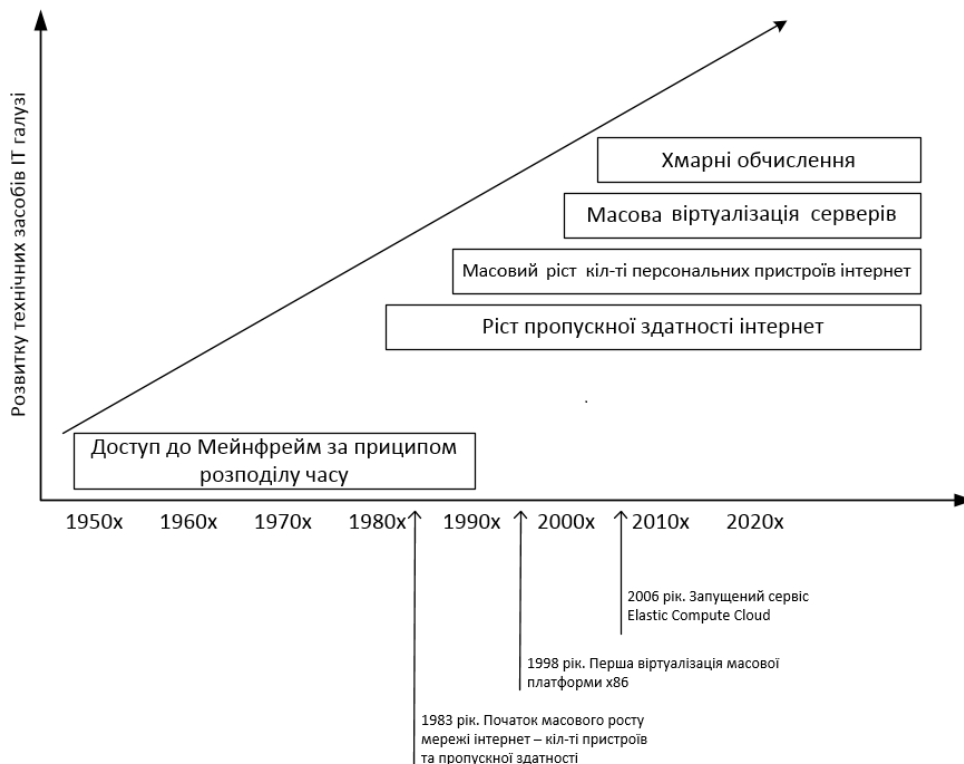


Рис. 2.1 Ріст попиту на хмарні обчислення

2.2 Сервісні моделі хмарних обчислень.

Хмарні обчислення - узагальний термін, який поєднує у собі кілька підходів та моделей з надання та управління ІТ сервісами. Більшість користувачів визначає хмарні обчислення лише за однією ознакою -

мережевим доступом, але хмарні обчислення це набагато більш об'ємна сутність. Згідно з визначенням Національного інституту стандартів і технології (NIST) США, хмарні обчислення (від англ. Cloud Computing) — це модель забезпечення доступу через мережу до спільного пулу обчислювальних ресурсів, що підлягають налаштуванню (наприклад, до комунікаційних мереж, серверів, засобів збереження даних, прикладних програм та сервісів), і які можуть бути оперативно надані та вивільнені з мінімальними управлінськими затратами та зверненнями до провайдера.

Користувач отримує доступ до даних з будь-якої точки за допомогою лише підключення до інтернету. Для отримання доступу до хмарних обчислень користувач реєструється та автентифікується. Швидкість передачі залежить, наприклад, від швидкості інтернету, потужності сервера тощо. Управління хмарними обчисленнями здійснює сам хост, який має достатню кількість серверів зберігання та обробки даних, через які користувачі дуже швидко отримують доступ.

Хмарні послуги постачаються через Інтернет та доступні у всьому світі за допомогою мережі інтернет [4]. Хмарні обчислення являють собою концепцію надання ІТ ресурсів у вигляді послуг: все як послуга (XaaS), мережа як послуга (NaaS), програмне забезпечення як послуга (SaaS), інфраструктура як послуга (IaaS), платформа як послуга (PaaS), моніторинг як послуга (MaaS), зв'язок як послуга (CaaS) та інші. Цей підхід є одним з наслідків більш масштабного зсуву в суспільній свідомості - переходу від продукт-орієнтованої ери виробництва до сервіс-орієнтованої ери надання послуг.

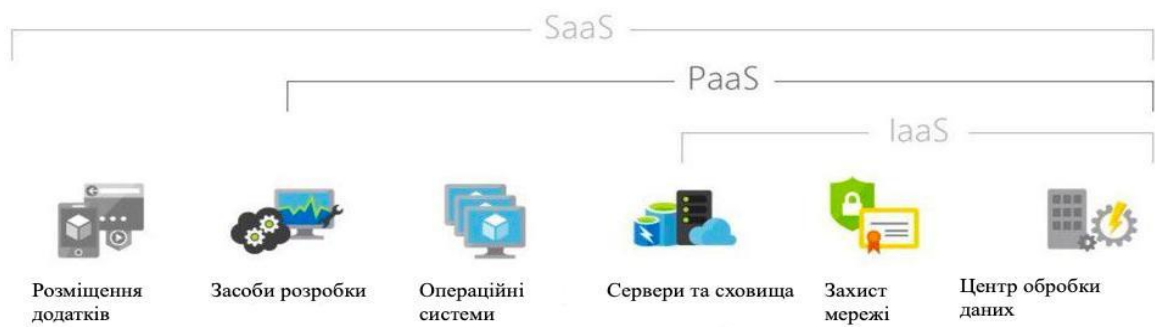


Рис.2.2 SaaS

2.2.1 Все як послуга (XaaS)

XaaS - це технологічна концепція, що охоплює декілька концепцій, пов'язаних з технологічними інноваціями в хмарі. Використання моделі XaaS в організаціях сприяє вирішенню проблем в галузі ІТ, одночасно прогресуючи в ситуаціях, коли є потреба зростати і розширюватися, дозволяючи швидко переходити з одного ринку на інший та з однієї бізнес-моделі на іншу. Це модель надання послуг, за які споживач не платить, поки вони не надаються. Під це визначення потрапляють всі послуги, які надаються через інтернет із застосуванням хмарних обчислень.

IaaS, PaaS і SaaS - це основні моделі надання хмарних послуг. Те, як вони співвідносяться одне з одним, часто зображують у вигляді піраміди з різним рівнем контролю інформації. Вершина - це кінцевий користувач, який працює з особистими даними, «загорнутими» в програму або сервіс зі зручним інтерфейсом. Програма або сервіс розгортаються на якійсь технологічній платформі, це другий рівень піраміди. Нарешті, її основа — це інфраструктура: віртуальні сервери, обчислювальні потужності, накопичувачі і канали зв'язку.

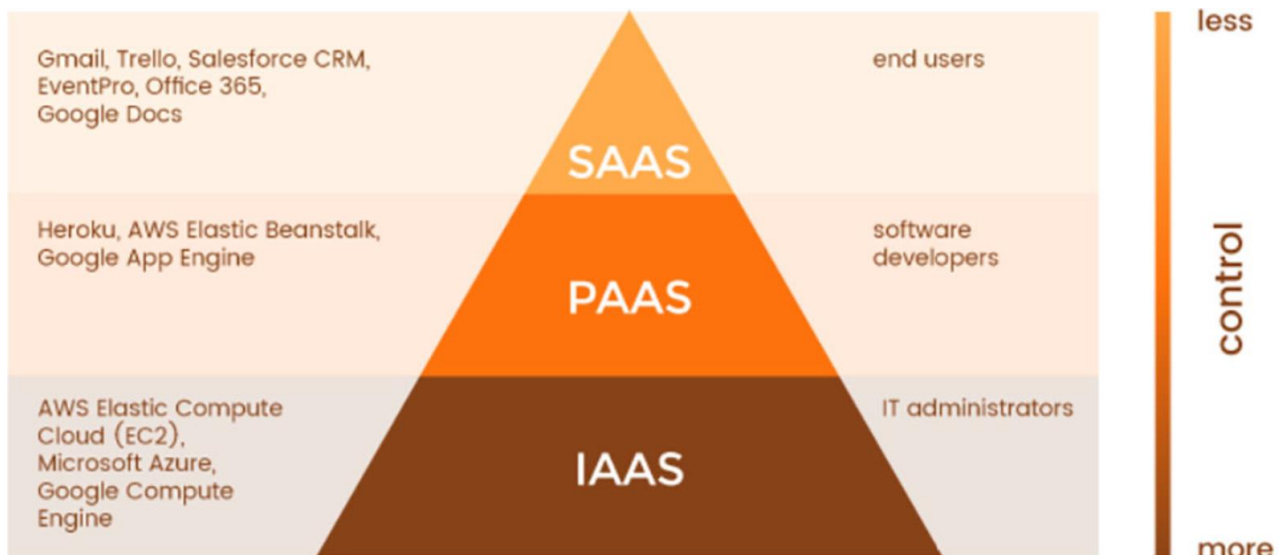


Рис 2.3 Піраміда основних моделей хмарних послуг

В IaaS споживач отримує споживач отримує інформаційно-технологічні ресурси — віртуальні сервери з певною обчислювальною потужністю та обсягами пам'яті. Всім «залізом» займається провайдер. Він встановлює на нього ПЗ для створення віртуальних машин, але не займається установкою і підтримкою ПЗ користувача. Провайдер контролює тільки фізичну та віртуальну інфраструктуру. Приклади IaaS: IBM Softlayer, Hetzner Cloud, Microsoft Azure, Amazon EC2, GigaCloud. Клієнти IaaS — це системні адміністратори компаній.

В PaaS хмарний провайдер надає споживачу доступ до операційних систем, засобів розробки і тестування, системам управління базами даних. Провайдер контролює не тільки сервери, системи зберігання даних і обчислювальні потужності, але також пропонує користувачеві на вибір певні платформи і засоби управління ними. Приклади PaaS: Google App Engine, IBM Bluemix, Microsoft Azure, VMWare Cloud Foundry. Користувачі PaaS-сервісів — це розробники ПЗ.

В SaaS програми та сервіси розробляє і обслуговує провайдер, розміщує їх у хмарі і пропонує кінцевому користувачеві через браузер або додаток на його ПК. Клієнт лише вносить абонплату (або користується сервісом безкоштовно), оновленням і технічною підтримкою програм

займається провайдер. SaaS-сервіси можуть надавати місце для зберігання файлів (Dropbox), офісний пакет документів для роботи (Google Doc, Microsoft Office 365), допомагати організувати фотографії (Flickr) або спілкуватися з іншими людьми (Facebook). Основний клієнт SaaS-сервісів — звичайний користувач.

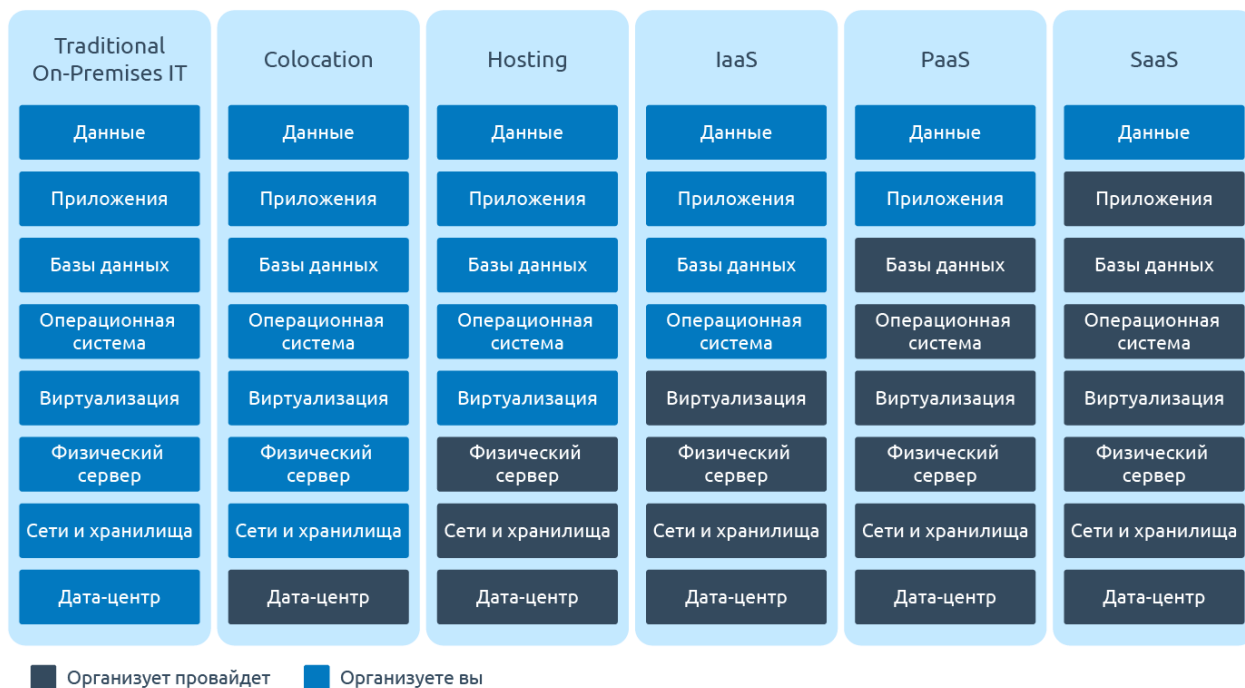


Рис.2.4 Основні моделі хмарних послуг

2.2.2 Програмне забезпечення як послуга (Software as a Service, SaaS)

Модель, де споживачеві надається можливість використання прикладного програмного забезпечення постачальника (розробника), який працює в хмарній інфраструктурі, доступного з різних клієнтських пристроїв або за допомогою тонкого клієнта, наприклад, браузера (веб-пошти) або інтерфейсу програми. Контроль і управління основною фізичною і віртуальною інфраструктурою хмари (мережі, сервери, операційні системи, зберігання) здійснюється постачальником SaaS. Дана схема передбачає, що користувач отримує в розпорядження програмне забезпечення, функції якого доступні через веб-інтерфейс, в той час як основна програмна частина програми знаходиться на сервері постачальника. Постачальник програмного

забезпечення як сервісу розробляє та керує власними програмами. Рішення SaaS в основному використовують архітектуру, в якій додаток обслуговує декілька користувачів та підтримує дані. Головна особливість SaaS в тому, що користувачеві не потрібно інсталиювати програму на свій комп'ютер. Також в розпорядженні споживача буде новітня версія програми, постачальник своєчасно оновлює потрібні файли.



Рис 2.5 Програмне забезпечення як послуга, графічне зображення

Програмне забезпечення як послуга має два різновиди :

- вертикальне SAAS - керує попитом конкретної організації (наприклад, програмне забезпечення для нерухомості, банківської справи, сільського господарства).
- горизонтальне SaaS. Це продукт, який зосереджено на програмному забезпеченні, наприклад, для маркетингу, кадрової роботи, інструментах для розробки продукції тощо.

Переваги програмного забезпечення як послуги:

- масштабованість. Дана послуга є масштабованою та надає клієнтам різноманітні функції відповідно до їх запити, наприклад супроводжуючі довідкові матеріали.

- надійність. SaaS-продукти в багатьох випадках надійніші рішень, встановлених на комп'ютер, тому що основна частина апаратних функцій на серверах постачальника дублюється. Якщо на стороні постачальника вийшов з ладу жорсткий диск, можна бути впевненим, що важлива інформація вже встигла записатися на резервні носії, що належать споживачу.

- актуальність. З кожним оновленням SaaS поступово вдосконалюється, що знижує завантаження персоналу та забезпечує кращий рівень послуг.

- універсальність. Користувач може працювати з будь-якого комп'ютера, головне, щоб був інтернет доступним. Багато з постачальників SaaS адаптують інтерфейси відповідного ПЗ не тільки до користування на ПК, але також і до специфіки мобільних пристроїв - смартфонів, планшетів. Також SaaS надає можливість одночасного користування одним і тим же інтерфейсом декількома користувачам

- економічність. У багатьох випадках SaaS-програми дешевше, ніж версії для ПЗ. Це обумовлено багатьма факторами: зокрема, відсутністю витрат на дистрибуцію дисків. До того ж розцінки від постачальників SaaS-програм прив'язані до конкретних функцій програми: користувач, таким чином, може не купувати ті можливості ПЗ, які він не задіюватиме. При цьому якісна технічна підтримка користувачів, як правило, входить в оплату підписки на хмарні додатків в рамках моделі SaaS. Традиційні формати ПЗ можуть вимагати поновлення «заліза» корпоративних комп'ютерів. Хмарні рішення, як правило, не дуже вимогливі до продуктивності апаратних компонентів ПК.

Недоліки програмного забезпечення як послуги:

- надання SaaS повністю залежить від доступу до інтернету, без мережі, користувач не матиме доступу до своїх даних чи ПЗ;

- недолік хмарного ПЗ в цілому і SaaS зокрема в тому, що між постачальником ПЗ і його користувачем часто передаються конфіденційні дані та не кожна система безпеки підприємства гарантує їх захист від перехоплення. Сучасні технології шифрування і перевірки прав доступу

дозволяють вибудувати дуже серйозні бар'єри, проте найважливіші види ПЗ недоцільно надавати в хмарному форматі. Можна відзначити недостатню універсальність концепції в цілому. Наприклад, антивірус в форматі SaaS не може бути настільки ж ефективним, як той, що працює за традиційною схемою, так як відповідний тип програм повинен мати повний доступ до ОЗП і жорстких дисків комп'ютера. Деякі базові функції антивірусного ПЗ, наприклад, сканування окремих файлів на предмет зараженості, багато виробників відповідного типу програм готові надавати користувачам за допомогою SaaS-платформ.

- обмеження функції додатків. Є складність з наданням повних версій ПЗ, тому постачальники хмарних послуг випускають ПЗ, у яких відсутні деякі функції.

Отже, SaaS, надаючи сервіс через Інтернет, надає споживачам можливість для підвищення ефективності роботи, економії фінансових коштів, досягнення необхідного комфорту користування різними додатками. Усі дані зберігаються в базі даних хмарного постачальника. Певні угоди забезпечують безпеку програм та даних. Використання SaaS значно економить загальні витрати підприємства.

2.2.3 Інфраструктура як послуга

Infrastructure as a Service (IaaS) - це модель хмарних обчислень, де споживачам надаються за передплатою фундаментальні інформаційно-технологічні ресурси - віртуальні сервери із заданою обчислювальною потужністю, операційною системою) і доступом до мережі. При користуванні IaaS споживач, використовує серверний час, кількість задіяних віртуальних процесорів і віртуальних обсягів пам'яті, а також простір зберігання), задану мережеву пропускну здатність, в деяких випадках - мережевий трафік. IaaS знаходиться на найнижчому рівні серед хмарних моделей обслуговування, на відміну від моделі PaaS (де провайдер надає

готове сполучне програмне забезпечення, СУБД, засоби розробки) і SaaS (на якому надається прикладне програмне забезпечення), в IaaS не передбачено контроль з боку постачальника послуг за які встановлюються програмним забезпеченням, він контролює лише фізичну і віртуальну інфраструктуру.

Хмарні провайдери надають IaaS у вигляді двох типів віртуальних серверів:

E-Cloud, призначений для корпоративних клієнтів і побудований на базі платформи VMware. Споживач отримує ІТ-ресурси, за допомогою яких він сам будує свою інфраструктуру; S-Cloud, призначений для малого і середнього бізнесу. Споживач отримує в оренду готовий сервер необхідних конфігурацій для розміщення своїх сервісів.



Рис. 2.6 Інфраструктура як послуга

IaaS дозволяє швидко розгортати і демонтувати середовища тестування і розробки, оперативно виводячи нові додатки на ринок, збільшувати масштаб середовищ тестування і розробки швидко і економічно. При зберіганні, архівації та відновленні даних підприємствам-споживачам немає необхідності робити капітальні вкладення і долати труднощі, пов'язані зі зберіганням даних і управлінням сховищем, для чого зазвичай потрібні висококваліфіковані фахівці з управління даними і забезпечення відповідності нормативним вимогам. IaaS дозволяє справлятися з

непередбачуваним попитом і стабільно зростаючими потребами в зберіганні даних, також може спланувати свою систем резервного копіювання та відновлення і управління ними. IaaS забезпечує всю інфраструктуру для підтримки веб-додатків, включаючи сховище, веб-сервери і сервери додатків, а також мережеві ресурси. Підприємства можуть швидко розгорнути веб-додатки на базі IaaS і легко масштабувати інфраструктуру, коли число звернень до додатків стає непередбачуваним. IaaS може забезпечити високопродуктивні обчислення на суперкомп'ютерах, в комп'ютерних мережах або кластерах складних завдання, які включають мільйони змінних і великі обсяги обчислень, інтелектуальний аналіз наборів даних без значного вкладення коштів.

Користувачеві надається можливість використання хмарної інфраструктури для самостійного управління ресурсами обробки, зберігання, мережами та іншими фундаментальними обчислювальними ресурсами, наприклад, споживач може встановлювати і запускати довільне програмне забезпечення, яке може включати в себе операційні системи, платформенне та прикладне програмне забезпечення. Користувач може контролювати операційні системи, віртуальні системи зберігання даних і встановлені програми, а також скористатися обмеженим контролем за набором доступних мережевих сервісів (наприклад, фаєрволом, DNS)

Переваги інфраструктури як послуги:

- усуває капітальні витрати і знижує поточні витрати. IaaS дозволяє уникнути попередніх витрат на розгортання локального центру обробки даних і управління ним.

- покращує безперервність бізнес-процесів та ефективність аварійного відновлення. Реалізація високого рівня доступності, безперервності бізнес-процесів і аварійного відновлення вимагає значних витрат, так як для цього потрібно багато одиниць обладнання та співробітників. Однак завдяки відповідній SLA, IaaS дозволяє знизити витрати і використовувати додатки і

дані в звичайному порядку при виникненні надзвичайної ситуації або відключенні харчування.

- дозволяє швидко впроваджувати інновації. При необхідності запустити новий продукт або ініціативу, обчислювальна інфраструктура буде підготовлена за хвилини або години, а не за дні, тижні чи місяці, як в разі використання внутрішньої інфраструктури.

- дозволяє швидко масштабувати ресурси, щоб обробляти піковий обсяг звернень, а потім знову зменшувати обсяг виділених ресурсів при зменшенні активності, для заощадження коштів.

- постачальник хмарних служб забезпечує безпеку додатків і даних за рахунок обмеженого доступу та створення резервних копій.

- усуває необхідність в обслуговуванні та модернізації програмного та апаратного забезпечення чи при усуненні неполадок, оскільки в IaaS суттєво менше проблем із сумісністю.

Недоліки інфраструктури як послуги:

- недолік хмарного ПЗ в цілому і IaaS зокрема в тому, що між постачальником ПЗ і його користувачем часто передаються конфіденційні дані та не кожна система безпеки підприємства гарантує, що дані не потраплять в руки третіх осіб. Ця ж умова актуальна для компаній, що працюють в контрольованих галузях. В такому випадку зберігання даних на серверах, які належать підприємству та знаходяться на території інших держав, заборонено чинним законодавством.

- необхідність наявності постійного широкосмугового підключення до мережі Інтернет.

Таким чином, IaaS - це послуга з надання користувачеві обчислювальних ресурсів за запитом, на основі яких він може розгорнути і використовувати будь-яке ПЗ (включаючи ОС), позбавляє підприємства від необхідності підтримки складних інфраструктур центрів обробки даних, клієнтських і мережевих інфраструктур, а також дозволяє зменшити пов'язані з цим капітальні витрати та поточні витрати. З огляду на все незаперечні

переваги хмарної технології IaaS, можна рекомендувати її в якості кращого рішення для забезпечення ІТ інфраструктури для бізнесу.

2.2.4 Платформа як послуга (Platform as a service, PaaS)

Платформа як послуга - це модель хмарних обчислень, яка дозволяє споживачу отримувати доступ до інформаційно-технологічних платформ: операційних систем, систем управління базами даних, проміжного ПЗ, засобів розробки і тестування розміщених у хмарних провайдерах. У цій моделі вся інформаційно-технологічна інфраструктура, включаючи обчислювальні мережі, сервери, системи зберігання, цілком керується провайдером, ним же визначається набір доступних для споживачів видів платформ та набір керованих параметрів платформ, а споживачеві надається можливість використовувати платформи, створювати їх віртуальні екземпляри, встановлювати, розробляти, тестувати, експлуатувати на них прикладне програмне забезпечення, при цьому динамічно змінюючи кількість споживаних обчислювальних ресурсів. PaaS використовує базове устаткування та програмні засоби нижнього рівня, що надаються хмарою. Кінцевий користувач просто використовує апаратне забезпечення ЦОД, операційну систему, проміжне ПО, бази даних постачальника для розміщення свого додатку або сервісу.



Рис 2.7 Платформа як сервіс

РaaS має деяку схожість з IaaS, проте клієнти РaaS-провайдера можуть користуватися середовищем, додатками, але не мають можливості масштабувати інфраструктуру, вимнути потужності або змінити конфігурацію. Різниця між послугами IaaS і РaaS полягає в тому, що в рамках платформи як сервісу споживач отримує обчислювальну платформу і стек рішень, але не впливає на конфігурацію віртуальної інфраструктури. РaaS надає платформу і середовище розгортання з попередньо встановленими налаштуваннями, яка дозволяє користувачам розробляти, тестувати і розгортати свої додатки. При цьому грамотна стратегія використання API робить роботу з РaaS максимально ефективною. Приклади РaaS-рішень: Google App Engine, VMWare Cloud Foundry, IBM Bluemix і ін.

Переваги платформи як сервісу:

- провайдер РaaS-послуг бере на себе всі оновлення, виправлення і поточне обслуговування програмного забезпечення;
- споживачу не потрібно попередніх інвестицій в обладнання та програмне забезпечення, все необхідне надає провайдер;
- гнучкість при розгортанні платформи: інфраструктура розробки, тестування і розгортання повністю керується провайдером РaaS. РaaS пропонується через розміщену інфраструктуру постачальника хмарних послуг.
- доступність на декількох платформах. РaaS надає можливість отримання доступу з будь-якого місця та пристроїв. Крім того, деякі постачальники послуг пропонують вибір розробок для декількох платформ, таких як комп'ютери та браузері, що роблять міжплатформенні програми швидшими та простішими в розробці.
- управління життєвим циклом додатків. Можливості, що надаються РaaS, підтримують життєвий цикл додатків: створення, тестування, розгортання, управління, зміну налаштувань.

- відсутність витрат на придбання та обслуговування нового програмного забезпечення завдяки наявності інструментів, які надаються хмарними провайдерами.

Зазвичай користувачі отримують доступ до пропозицій PaaS через веб-браузер. PaaS може доставлятися через державні, приватні або гібридні хмари. Завдяки загальнодоступній хмарній системі PaaS, споживач контролює розгортання програмного забезпечення, тоді як постачальник хмарних технологій постачає всі основні компоненти ІТ, необхідні для розміщення програм, включаючи сервери, системи зберігання даних, мережі, операційні системи та бази даних. Забезпечення розміщеного середовища для розробки, тестування та розгортання програм найпоширеніше застосування для PaaS.

Недоліки платформи як сервісу:

- так як PaaS - хмарна послуга, вона несе в собі ті ж ризики, що є в будь-яких хмарних пропозиціях, наприклад, проблеми безпеки даних, так як в основі PaaS лежить принцип використання загальних ресурсів - мереж та серверів. З іншого боку, найбільші хмарні провайдери ефективно справляються з такими проблемами. Тому ризики для безпеки даних не такі високі, як здаються;

- деякий ризик моделі PaaS в тому, що споживач залежить від політики провайдера і від проблем з інфраструктурою і серверами.

Не зважаючи на це, модель PaaS дає гнучкість для розробки додатків і зручність в управлінні ресурсами. Компанії використовують PaaS, щоб розробляти, запускати, управляти API і мікросервісами. Також інструменти, які надаються PaaS, дозволяють підприємствам аналізувати дані, які допомагають їм приймати оптимальні рішення і точніше прогнозувати події. Підприємства можуть використовувати PaaS, щоб отримати доступ до платформи управління бізнесом. Платформа управління надається як сервіс нарівні з іншими хмарними послугами. Модель PaaS може також служити механізмом доставки для платформ комунікації. Це дозволяє розробникам

додавати опції спілкування - голос, відео, месенджери. Провайдер PaaS може надавати такі послуги, як встановлення та підтримку баз даних підприємства. Також у PaaS підтримуються середовища додатків, мови програмування та інструменти, які використовуються для інтернету речей. В управлінні майстер-даними входять процеси, політики, стандарти і інструменти, які керують важливими бізнес-даними компанії. PaaS також включає механізм управління сервісом, такий як управління робочим процесом, виявлення та резервація. Завдяки цим особливостям це одна з найнадійніших і захищених хмарних служб.

2.4 Компоненти архітектури хмарних обчислень

Нова технологія хмарних обчислень є стрімко зростаючою та достатньо гнучкою для використання великими та дрібними підприємствами. Більшість підприємств мають потребу у хмарних послугах для зберігання, захисту інформації та отримання доступу до неї з будь якого місця у будь-який час.

Фронтенд та бекенд – це дві основні складові архітектури хмарних обчислень, вони взаємодіють між собою в мережі.

Фронтенд (front-end) - це клієнтська частина користувацького інтерфейсу, якою керує користувач та яка використовується відповідно до його вимог. Фронтенд складається з інтерфейсів та додатків, за допомогою яких користувач отримує доступ до хмарних обчислень (наприклад, додаток чи браузер).

Бекенд (back-end) - це частина, якою керують компанія-провайдер хмарних послуг, власне це механіка сервісу. Бекенд має віртуальні машини, систему безпеки, великі сховища даних та сервери. Бекенд призначений для управління політикою безпеки та управління трафіком.

Функція архітектури хмарних обчислень полягає в ефективному моделюванні заданої функціональності системи в реальному ІТ-світі. Архітектура хмарних обчислень полягає фактично в абстрагуванні трьох

моделей (IaaS, PaaS та SaaS) таким чином, щоб конкретна організація, яка використовує хмарні обчислення, досягла поставлених цілей і задач.

Архітектурі хмарних обчислень повинна задовольняти таким вимогам:

- створення еластичного пулу віртуальних ресурсів;
- підтримка механізму доставки сервісом «за вимогою»;
- автоматизація процесів управління ІТ;
- забезпечення еластичного масштабування та неперервності бізнес-процесів;
- підтримка безпеки систем і процесів;
- інтеграція продуктів і забезпечення мультивендорних рішень.

Архітектура хмарних обчислень складається з наступних основних компонентів:

- гіпервізор, яких складається з програмного та апаратного забезпечення, а також з вбудованого програмного забезпечення, що створює та запускає віртуальні машини. Гіпервізор надає користувачеві віртуальну операційну платформу, яка дозволяє керувати операційною системою користувача для керування хмарою. Поняття гіпервізора близьке до поняття традиційного терміну ядра в операційній системі.

- ПЗ для управління, складається з різноманітних стратегій, що підвищують продуктивність роботи хмари. Програмне забезпечення для управління забезпечує своєчасну доставку сховища, необхідний рівень безпеки, можливість постійного доступу тощо. Важливими функціями ПЗ для управління є забезпечення аудиту відповідності, управління аварійними ситуаціями та реалізація плану дій у надзвичайних ситуаціях.

- ПЗ для хмарного розгортання, ініціює роботу моделей SaaS, PaaS та IaaS. Розгортання складається з розгортання усіх необхідних установок та конфігурацій хмари, реалізується в бекенді.

- маршрут зв'язку - частина архітектури, яка забезпечує підключення усієї хмари. Хмарні сервери підключаються за допомогою цього віртуального маршруту зв'язку. Це дозволяє налаштовувати маршрути та протоколи. Швидкість передачі залежить від швидкості інтернет-з'єднання.

- хмарний сервер - це віртуальний сервер, який розроблений та розміщений на платформі хмарних обчислень через мережу інтернет, доступний з будь-якого місця. Хмарні сервери не мають апаратних проблем, наявних на фізичних серверах. Хмарні сервери стабільні, швидкі, захищені та забезпечують роботу програмного забезпечення і можуть працювати незалежно один від одного.

- служба зберігання даних, розроблена для створення додатків, служб, підприємств, що мають доступ до зберігання за межами сайтів та надаються миттєво. Служба хмарного зберігання даних забезпечує реалізацію автоматичного масштабування. Модель IaaS забезпечує масштабовану та гнучку здатність зберігання за допомогою інтерфейсу API сервісів, онлайн-інтерфейсів та додатків. Служба хмарного зберігання надає користувачеві віддалений доступ через мережу інтернет.

Архітектуру хмарних обчислень відображають компоненти, що в неї вбудовані. Архітектура хмарних обчислень передбачає різні методи доставки послуг споживачам. Вся архітектура хмарних обчислень забезпечує інтеграцію з хмарними системами різних провайдерів та можливість організувати більшу пропускну здатність, внаслідок чого користувач отримує доступ до хмари через мережу інтернет в будь-який час та з будь-якого місця.

2.5 Моделі розгортання хмарних сервісів

Виділяють такі чотири моделі розгортання хмарних сервісів:

Публічна (Public Cloud) - передбачає спільне використання платформ кількома організаціями. Найпоширеніший спосіб використання хмарних технологій. Все обладнання, ПЗ та інфраструктура належать провайдеру, він же займається його обслуговуванням через свій штат ІТ-фахівців. Клієнти лише орендують необхідні обчислювальні потужності. Управлінням такої

хмари, зазвичай, займається зовнішній провайдер, наприклад, Amazon EC2, GoogleApps, Salesforce.

Приватна (Private Cloud) – передбачає платформу, яка створюється і контролюється однією організацією. Це фізично ізольована віртуальна інфраструктура з ексклюзивним доступом, підвищеною продуктивністю та безпекою. Її можна підключити до вже існуючої інфраструктури або інтегрувати з публічною хмарою. Приватна хмара призначена для великих компаній зі значною кількістю ІТ-сервісів для яких важлива висока інформаційна безпека та здатність витримувати пікові навантаження на ІТ-інфраструктуру.

Гібридна (Hybrid Cloud) - передбачає об'єднання приватної та публічної хмари в одну інфраструктуру. Ресурси публічної хмари можна використовувати для запуску нових або короткострокових проєктів, поступово нарощуючи потужності. У приватній хмарі можна розміщувати бізнес критичні додатки та продуктивне середовище, якому необхідна висока продуктивність та безпека. Це зручний і сучасний спосіб організації роботи бізнесу.

2.5.1 Архітектура, структура та переваги публічної хмари.

Публічна хмара на сьогодні найпоширеніша концепція надання хмарних послуг. В даному випадку вся ІТ-інфраструктура розміщується у сервіс-провайдера (компанії, яка надає сервіс). Такий оператор самостійно закуповує чи орендує сервери, системи зберігання даних, мережеве обладнання та інші апаратні компоненти, оплачує ліцензії на програмне забезпечення, а також підтримує, модернізує і оновлює всі системи в міру необхідності. При цьому сервіс-провайдер надає хмарні послуги великій кількості незалежних замовників, спільно використовують одну і ту ж ІТ-інфраструктуру, яка знаходиться під його керуванням і контролем. Компанія-клієнт в свою чергу отримує доступ до необхідного обсягу ІТ-ресурсів за певну абонентську плату.

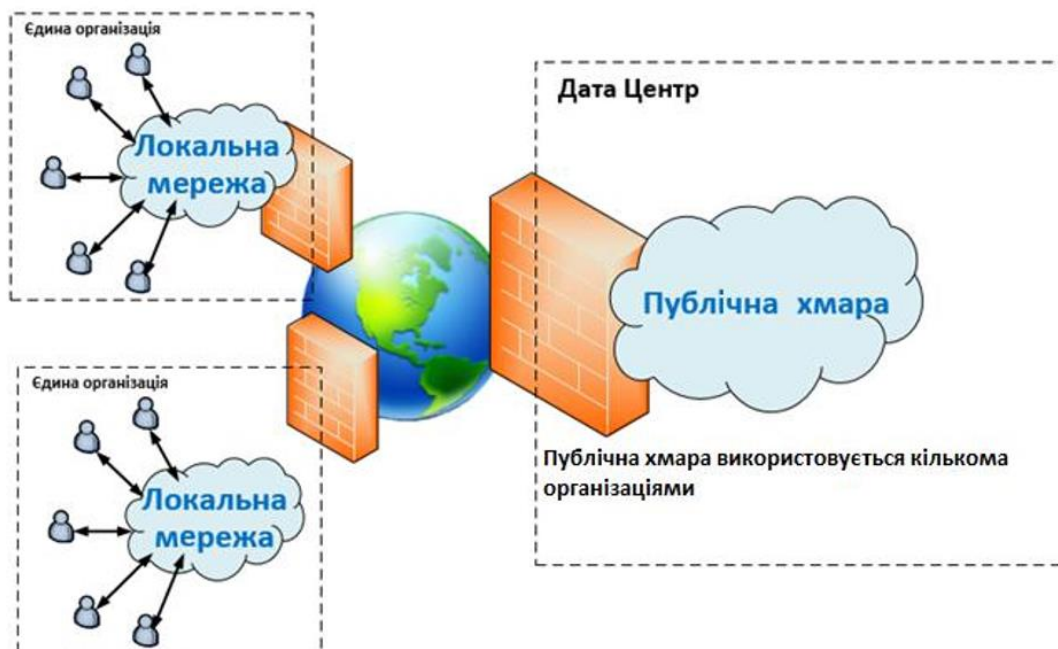


Рис 2.8. Публічна хмара

Велика популярність концепції публічного хмари обумовлена кількома потужними чинниками, а саме:

- Гнучкість (як технологічна, так і фінансова), швидка реакція на пікові навантаження найголовніші аспекти використання публічних хмар. Власна ІТ-інфраструктура вимагає від підприємства суттєвих капітальних вкладень, в той же час, хмарна модель дозволяє платити тільки за спожиті ІТ-сервіси, перетворюючи капітальні витрати в операційні. Багатьом підприємствам простіше вносити відносно невелику абонплату, ніж фінансувати власну інфраструктуру. Власна інфраструктура передбачає надмірність (надмірні фінансові вкладення), оскільки необхідно не тільки передбачити певний запас потужності для пікових навантажень, але і забезпечити резервування ключових підсистем. Публічна хмара повністю знімає цю проблему. Обчислювальні ресурси можуть бути отримані саме в тому обсязі, який необхідний на конкретний момент часу.

По суті, все, що потрібно споживачу для користування ресурсами публічного хмари - це стабільний широкопasmовий доступ в Інтернет. При

цьому від подібних послуг можна в будь-який момент відмовитися чи змінити провайдера;

- Безпека та резервування. Перевагою приватного хмари є і те, що провайдер повністю бере на себе ризики, пов'язані із забезпеченням працездатності ІТ-інфраструктури, її резервування, відмовостійкості, безпеки;

- Модернізація ІТ інфраструктури. Провайдер постійно модернізує свою ІТ-інфраструктуру, регулярно оновлюючи обчислювальний обладнання та версії програмного забезпечення. Відповідно, користуючись послугами громадського хмари, споживач отримує нове та краще обладнання та ПЗ.

Використання публічної хмари має певні недоліки, а саме:

- Контроль ІТ інфраструктури сторонньою компанією (провайдером);

- Залежність від швидкості та стабільності доступу до мережі інтернет;

- Потреба в нових підходах кібербезпеки для захисту даних.

2.5.2 Особливості архітектури приватної хмари

Мова йде про гнучку, глибоко віртуалізовану платформу, ресурси якої знаходяться в розпорядженні однієї організації. Фізично ІТ-інфраструктура може розміщуватися як в дата-центрі споживача, так і за його межами - конкретне місце не має значення. Нерідко трапляється ситуація, коли приватна хмара являє собою мережу з декількох корпоративних дата-центрів. Приватне хмара має цілий ряд переваг в порівнянні з публічної платформою. Наприклад, в загальному випадку вона забезпечує більш високу швидкість роботи (оскільки обчислювальні ресурси знаходяться в мережі організації).

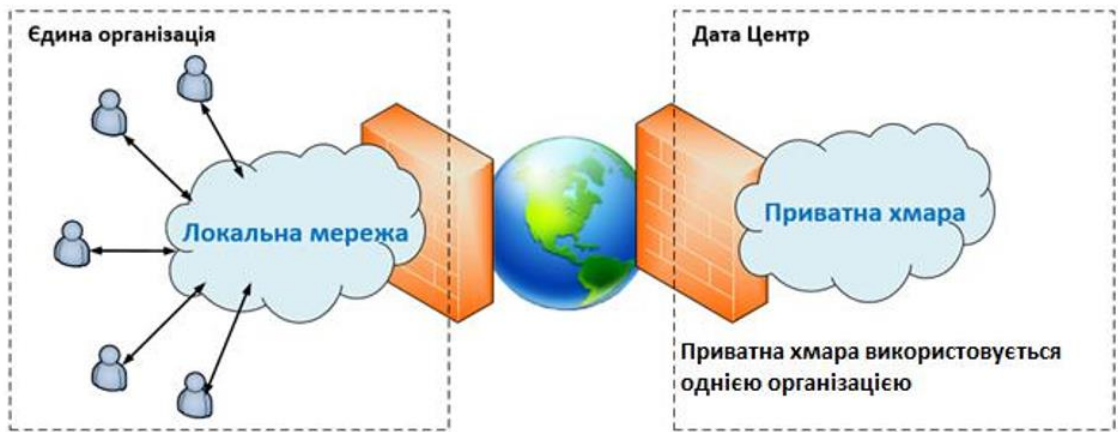


Рис.2.9 Приватна хмара

Це може бути важливо, наприклад, в разі використання потужних аналітичних систем або інструментів інженерного моделювання, при роботі з професійної графікою тощо. Приватна хмара забезпечує максимальний рівень кіберзахисту, адже дані не залишають умовний внутрішній периметр безпеки.

Приватна хмара, на відміну від публічної передбачає, що вся обчислювальна інфраструктура знаходиться під повним контролем самої організації.

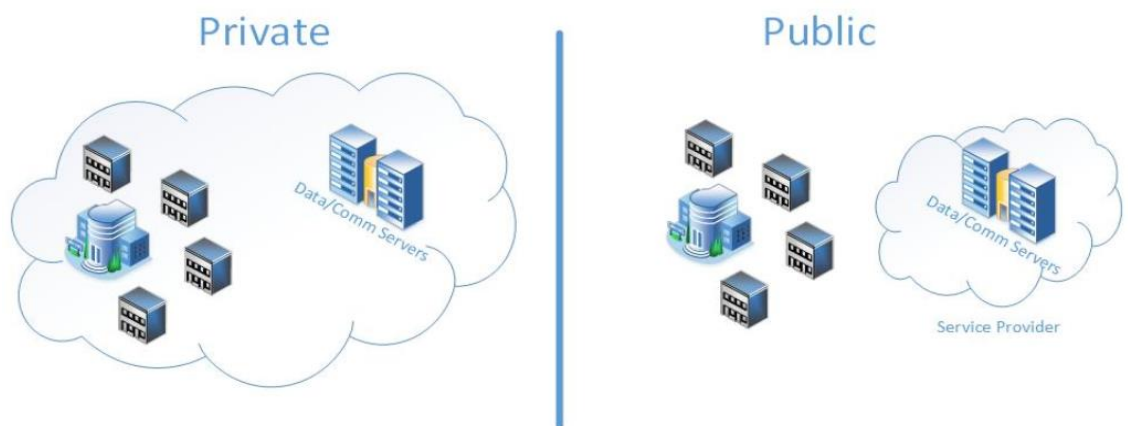


Рис 2.10 Порівняння схеми роботи публічної та приватної хмар.

Основними перевагами приватної хмари є наступні показники:

- Висока швидкість роботи;
- Високий рівень безпеки;
- Ексклюзивне користування ресурсами одним підприємством;
- Повний контроль обладнання та ПЗ.

Віртуалізована приватна хмара підвищує ефективність використання ІТ-ресурсів підприємства, але в той же час вимагає фінансових ресурсів на закупівлю обладнання, ПЗ та сюди постійні витрати на висококваліфікований персонал і регулярні оновлення. Також при створенні приватного хмари потрібно досить точно передбачити майбутнє навантаження на роки вперед - якщо розрахунок виявиться невірним, це призведе до зайвої «заморожування» коштів, або до того, що в самий невідповідний момент обчислювальної потужності або обсягу системи зберігання виявиться недостатньо. Питання масштабування на вимогу стоять в цьому випадку досить гостро і не завжди можуть бути оперативно вирішені. Тобто, є ризики більшості обмежень, характерних для власної ІТ-інфраструктури підприємства.

Основні недоліки приватного хмари:

- Великі фінансові вкладення на етапі закупівлі;
- Наявність висококваліфікованого персоналу;
- Затрати на модернізацію и оплату ліцензій ПЗ;
- Необхідність точного планування навантажень на тривалий період часу.

Останнім часом все більш популярним стає підхід, який отримав назву Trusted Private Cloud. У цьому випадку замовник розгортає власну приватну хмару на базі ІТ-інфраструктури сервіс-провайдера, який для цього виділяє абоненту пул ресурсів власного хмарного дата-центру в ексклюзивне користування. При цьому оператор забезпечує тільки загальну працездатність ІТ-системи або надає певний набір послуг.

2.5.3 Архітектура, переваги та особливості впровадження гібридних хмар

Гібридна хмара передбачає створення комбінації з двох чи більше приватних і публічних хмар в одну інфраструктуру з можливістю спільного використання їх потужностей та даних.

Ресурси публічної хмари використовуються для запуску нових проектів з можливістю поступового нарощування потужностей, (на етапі розробки використовується публічна хмара, для зберігання даних - публічна). Використання гібридної хмарі також доцільне при розміщенні продуктивних середовищ, критичних для бізнесу додатків, яким необхідна висока безпека та продуктивність, при підтримці підприємством додатків SaaS та переміщенні даних між центром даних та приватними ресурсами. Гібридна хмара надає можливість скористатися зовнішніми ресурсами при нестачі власних потужностей, однак найчастіше гібридні хмари використовуються в разі пікових навантажень як можливість виходу за межі потужностей хмарного середовища. Можливості гібридної хмари завдяки функції автоматичного масштабування дозволяють розподіляти навантаження між хмарами та надають більшу кількість варіантів для розміщення даних. Баланс між попитом на обробку даних та обчислювальними технологіями надає підприємствам можливість спланувати роботу власної ІТ інфраструктури, що допоможе впоратися з переповненнями (Рис. 2.11).

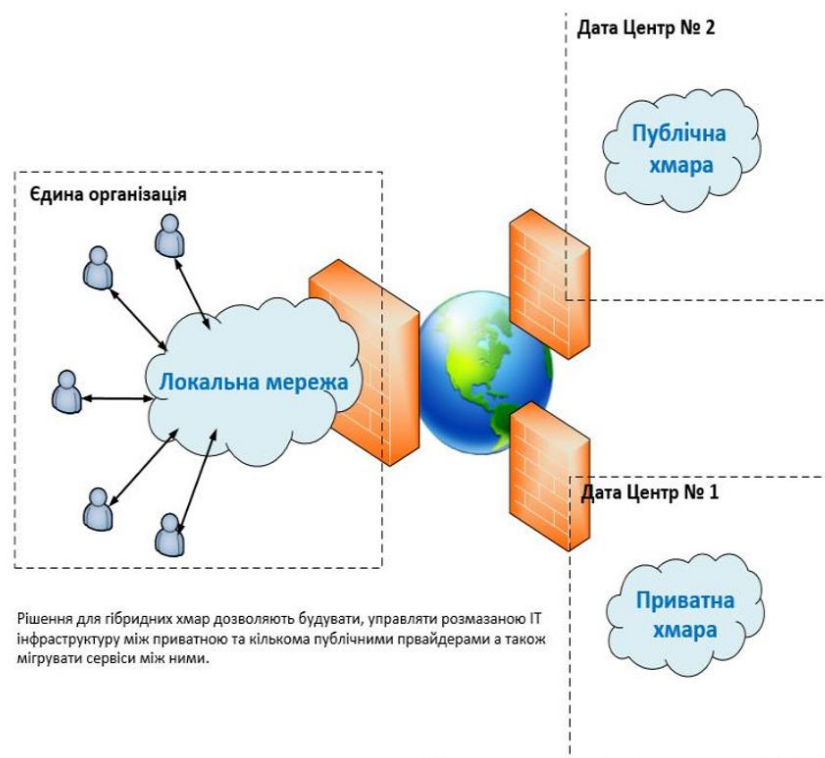


Рис. 2.11 Гібридна хмара

Архітектура гібридної хмари складається з трьох компонентів:

1. Публічна інфраструктура як послуга (Microsoft Azure чи Google Cloud Drive, веб-сервіс Amazon);
2. Приватна хмара, яка працює на платформі, або через приватного постачальника сервера, розміщеного у хмарі;
3. WAN – мережа для поєднання середовищ.

Підприємство не має прямого контролю над використанням архітектури в публічній хмарі. Приватна хмара повинна використовуватися для досягнення сумісності, використовуючи публічні чи приватні хмари. Для цього потрібне апаратне забезпечення всередині ЦОД, що включає сервери, сховище, балансири навантажень, LAN. Існують також інші компоненти архітектури гібридної хмари, наприклад, гібридний користувальницький інтерфейс, гібридний сервер, резервне копіювання, гібридне середовище розробки, функції гібридного додатка, гібридну мультимедіа та веб-додаток, гібридні дані.

Гібридна хмара створюється за допомогою гіпервізора та шарів хмарного програмного забезпечення. Ці реалізації програмного забезпечення та служб дозволяють даним мігрувати між приватною та публічною хмарою. За допомогою сервісів та програмного забезпечення можуть бути створені передові програми на загальнодоступних та приватних платформах.

Існує певний перелік інструментів управління гібридною хмарою, наприклад, Egenera P Cloud Director, Rightcale, Scalr і Cisco IBM Cloud Orchestrator, Red Hat Cloudforms, Abiquo Hybrid Cloud, VMware Cloud Suit тощо для виконання різнопланових задач, таких як налаштування, розгортання середовищ розробки, надання послуг та управління послугами, розгортання гнучких інструментів для масштабування, управління середовищем та інфраструктурними системами, аудит безпеки, відновлення після аварій, планування надзвичайних ситуацій, виставлення рахунків, обробка робочого процесу та багато іншого.

Для подолання проблеми сумісності хмарного ПЗ, необхідно реалізувати сервери, локальні мережі та пристрої зберігання даних.

Переваги гібридної хмари:

- Вирішує проблеми нестачі або резервування внутрішніх ресурсів за допомогою використання обчислювальних потужностей провайдера;

- Забезпечує миттєвий доступ до ресурсів з можливістю самообслуговування, що дозволяє збільшити прибуток компанії;

- Надає можливість підприємствам використовувати хмарний сервіс на повну потужність, з економією витрат публічної хмари і з максимальним рівнем безпеки приватної хмари, що забезпечує доступність критично важливих даних, обмежуючи їх від доступу третіх осіб.

- Надає гнучкість та масштабованість, можливість змінюватися відповідно до інтенсивності використання хмари.

- Забезпечує надійність та доступність. Копіювання даних відбувається у хмарі, в кількох різних безпечних та надійних місцях, дозволяє обмежувати права доступу.

- Забезпечує захищеність та керованість даних. При доступності даних будь-коли та з будь-якого пристрою конфіденційні дані не повинні переміщуватися та зберігаються на серверах.

- Забезпечує бездоганну інтеграцію платформ, дозволяючи адаптувати необхідну потужність публічних ресурсів для спільного користування, а також максимальну безпеку конфіденційних даних.

У розрізі переваг для підприємств можна відмітити те, що при використанні гібридної хмари прискорюється трансформація цифрового середовища, надається можливість оперативного реагування на потреби бізнесу, організовується консолідована, прозора і контрольована структура витрат на утримання ІТ інфраструктури.

2.5.4. Впровадження громадських хмар

Громадська хмара - це гібридна форма приватної хмари, яка є платформою для кількох орендарів, які дозволяють різним організаціям працювати на спільній платформі. Мета цієї концепції - дозволити кільком замовникам працювати над спільними проектами та програмами, що належать спільноті, де необхідно мати централізовану хмарну інфраструктуру. Іншими словами, громадська хмара - це розподілена інфраструктура, яка вирішує конкретні питання секторів бізнесу, інтегруючи послуги, що надаються різними типами хмарних рішень.

Підприємства, що беруть участь у цих проектах, такі як тендерні, бізнес-організації та дослідницькі компанії, зосереджуються на подібних проблемах у своїй хмарній взаємодії. Їх спільні інтереси можуть включати концепції та політику, що стосуються питань безпеки та дотримання вимог, а також цілі проекту (Рис 2.12).

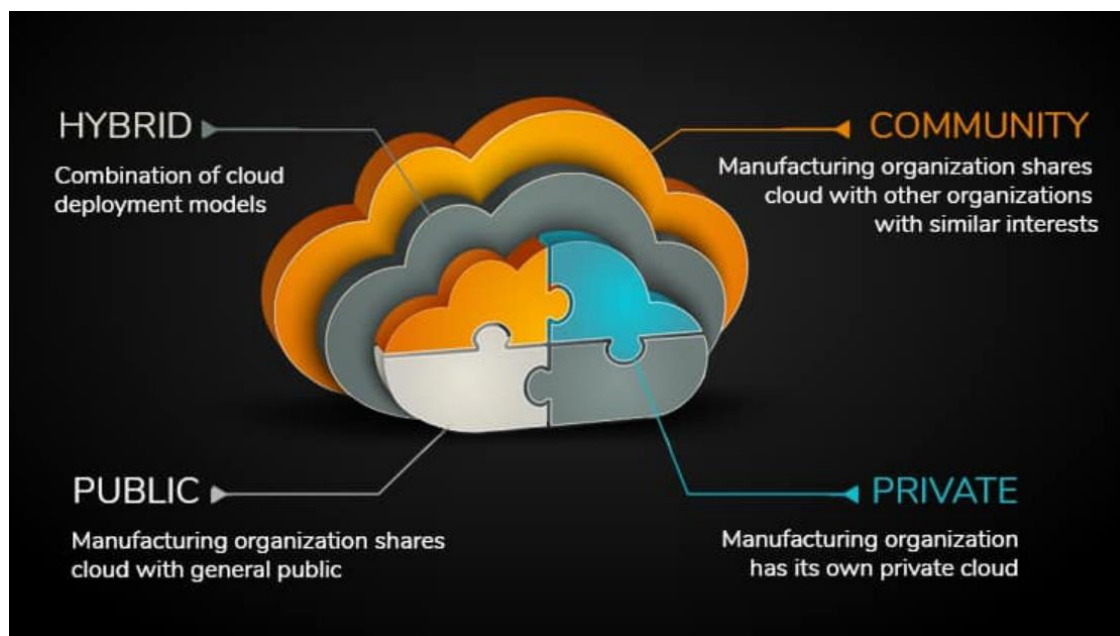


Рис 2.12. Моделі хмарного розгортання

Громадські хмарні обчислення полегшують своїм користувачам ідентифікувати та аналізувати свої бізнес-вимоги краще. Громадська хмара може розміщуватися в ЦОД, який належить одному з орендарів або сторонньому постачальнику хмарних послуг, і може бути як на місці, так і поза ним. Хмарна інфраструктура, яка призначена для використання

конкретною спільнотою споживачів із організацій, що мають спільні цілі (наприклад, місію, вимоги щодо безпеки, політику та відповідність різноманітним вимогам). Громадська хмара може перебувати у спільній власності, керуванні та експлуатації однієї чи більше організацій зі спільноти або третьої сторони (чи деякої їх комбінації). Така хмара може фізично знаходитись як в, так і поза юрисдикцією власника.

Можна навести приклади використання громадської хмари: кілька державних установ, які здійснюють операції між собою, можуть мати свої системи обробки на спільній інфраструктурі. Державні установи, які мають подібні вимоги щодо рівня безпеки, аудиту та конфіденційності, можуть використовувати громадські хмари. Також багатьом компаніям може знадобитися певна система або програма, розміщена на хмарних послугах. Хмарний провайдер може дозволити різним користувачам підключатися до одного середовища та логічно сегментувати свої сеанси. Такі налаштування позбавляють від необхідності мати окремі сервери для кожного клієнта, який має однакові наміри. (Рис.2.13)



Рис.2.13 Громадська хмара

Переваги громадської хмари:

- Гнучкість та масштабованість. Забезпечує сумісність між кожним із своїх користувачів, дозволяючи їм змінювати властивості відповідно до їхніх індивідуальних випадків використання. Вони також дозволяють компаніям взаємодіяти зі своїми віддаленими працівниками та підтримують використання різних пристроїв, будь то смартфон або планшет. Це робить цей тип хмарних рішень більш гнучкими до потреб користувачів. Складається із спільноти користувачів і як така є масштабованою в різних аспектах, таких як апаратні ресурси, послуги та робоча сила. Він враховує зростання попиту, і вам потрібно лише збільшити базу користувачів;

- Висока доступність та надійність. Захищає дані так само, як і будь-яка інша хмарна служба, тиражуючи дані та програми в декількох захищених місцях, щоб захистити їх від непередбачених обставин. Має надлишкову інфраструктуру, щоб переконатися, що дані доступні, коли та де вони вам потрібні. Висока доступність та надійність є найважливішими проблемами для будь-якого типу хмарних рішень.

- Безпека та відповідність вимогам. Користувачі можуть налаштувати різні рівні захисту своїх даних, а саме: можливість блокувати користувачам редагування та завантаження певних наборів даних, забезпечення конфіденційних даних суворими правилами щодо доступу до обміну конфіденційними даними, унікальним для певної організації, визначення пристроїв, які можуть зберігати конфіденційні дані.

- Зручність та контроль. Конфлікти, пов'язані зі зручністю та контролем, не виникають у громадській хмарі, оскільки всі орендарі діляться інфраструктурою та володіють нею та приймають рішення спільно. Це налаштування дозволяє організаціям мати свої дані ближче, уникаючи складностей приватної хмари.

- Менше роботи для IT-відділу. Наявність даних, програм та систем у хмарі означає, що підприємству не доведеться повністю ними керувати. Ця зручність позбавляє потреби орендарів використовувати додаткові людські ресурси для управління системою.

- Стійкість навколишнього середовища. У громадській хмарі організації використовують єдину платформу для всіх своїх потреб. Це об'єднання створює симбіотичний взаємозв'язок між розширенням та скороченням використання хмар серед клієнтів. Зі зменшенням кількості організацій, що використовують різні хмари, ресурси використовуються ефективніше, що призводить до меншого вуглецевого сліду.

Хмари спільноти - це відкриті системи, і вони усувають залежність організацій від постачальників хмарних послуг. Організації можуть досягти багатьох переваг, уникаючи недоліків як державних, так і приватних хмар.

Громадська хмара надає переваги організаціям у спільноті як окремо, так і колективно. Організаціям не потрібно турбуватися про проблеми безпеки, пов'язані з публічною хмарою, через закриту групу користувачів.

Ця нова модель хмарних обчислень має великий потенціал для підприємств, які шукають економічно вигідні хмарні послуги, для співпраці над спільними проектами.

Основним недоліком громадської хмари є питання щодо безпеки та довіри до хмарних обчислень в цілому. Проте проблеми безпеки щодо хмарних сервісів не є унікальними та стосуються будь-якого іншого типу хмар. Таким чином, можна з упевненістю сказати, що рішення громадської хмари надають унікальну можливість організаціям, які бажають працювати над спільними проектами.

Громадська хмара відповідає високим вимогам, одночасно забезпечуючи економічну ефективність публічної хмари. Тож вибір моделі хмарних обчислень залежатиме від індивідуальних потреб організацій, що планують співпрацювати. Громадська хмара надає економічно ефективний підхід, який має справу з меншою кількістю складностей в хмарному середовищі та одночасно забезпечує безпеку хмарної інфраструктури та даних.

3 МЕТОДИ ПОВНОЇ ВІРТУАЛІЗАЦІЇ ТА ПАРАВІРТУАЛІЗАЦІЇ В СИСТЕМАХ ХМАРНИХ СЕРВІСІВ ДЛЯ ВІРТУАЛІЗАЦІЇ ОБЛАДНАННЯ

Компанією-розробником програмних продуктів VMware, Inc. (США) у 1999 році було завершено розробку для віртуалізації x86-сумісних або x86-64-сумісних ОС на одному комп'ютері. Розробка полягала у поєднанні бінарного перекладу та безпосереднього виконання на процесорі, що дозволило на одному фізичному комп'ютері імітувати роботу кількох віртуальних. Завдяки впровадженню технології віртуалізації, світовими компаніями була отримана відчутна економічна вигода, що сприяло максимально швидкому впровадженню віртуалізованих (хмарних) обчислень. VMware пропонує два продукти віртуалізації для серверів: VMware ESX Server та VMware Server. На даний час постачальниками продуктів віртуалізації на світовому ринку, крім VMware, є IBM, HP, Parallels та Microsoft. Темпи зростання ринку віртуалізації в 2016 р. склали 78,7%. Зростання ринку віртуалізації збільшуватиметься протягом найближчих років. Технологія віртуалізації є платформою для побудови динамічної ІТ-інфраструктури, яка динамічно реагує на зростаючі потреби бізнесу. При віртуалізації на комп'ютері x86 створюється новий шар віртуалізації між апаратним забезпеченням та ОС (Рис 3.1), який дозволяє одночасну роботу кількох причірників ОС у віртуальних машинах на одному ПК, розділяючи та обмінюючись наявними фізичними ресурсами (накопичувач, процесор, пристрої вводу/виводу, пам'ять). Віртуалізація є технологією для проведення розробки та тестування ПЗ, підвищення рівня швидкості роботи ЦОД, консолідації серверів. Завдяки технології віртуалізації сервери можуть працювати в надійних працездатних конфігураціях, 24/7/365, без простоїв, необхідних для резервного копіювання або ТО обладнання.

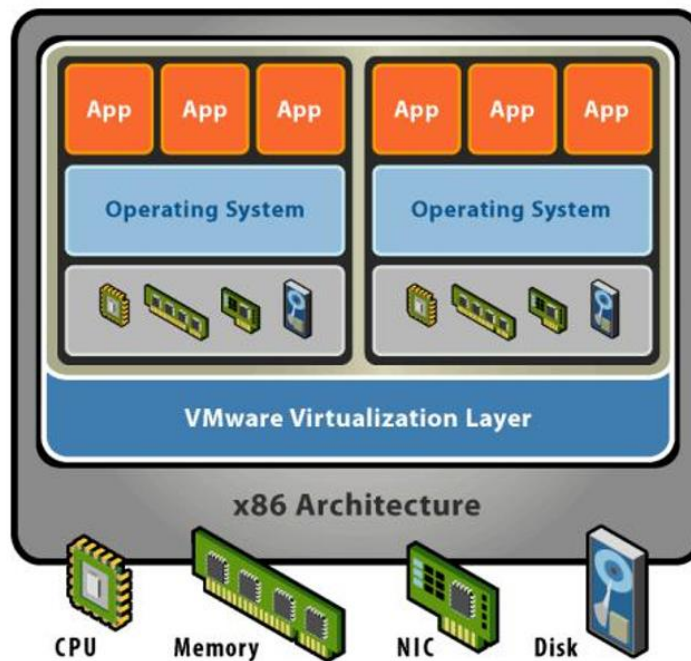


Рис 3.1. Шар віртуалізації x86

Для систем x86 при віртуалізації використовується розміщена архітектура чи архітектура гіпервізора. Розміщена архітектура встановлює та запускає шар віртуалізації як додаток поверх ОС, підтримуючи апаратну конфігурацію. Архітектура гіпервізора встановлює шар віртуалізації прямо в чистій системі на базі x86. Шар віртуалізації - це гіпервізор, тобто ПЗ, яке працює на апаратному забезпеченні та відділяє VM від фізичного серверу та в міру необхідності динамічно виділяє обчислювальні ресурси для кожної віртуальній машині, (Рис. 3.2).

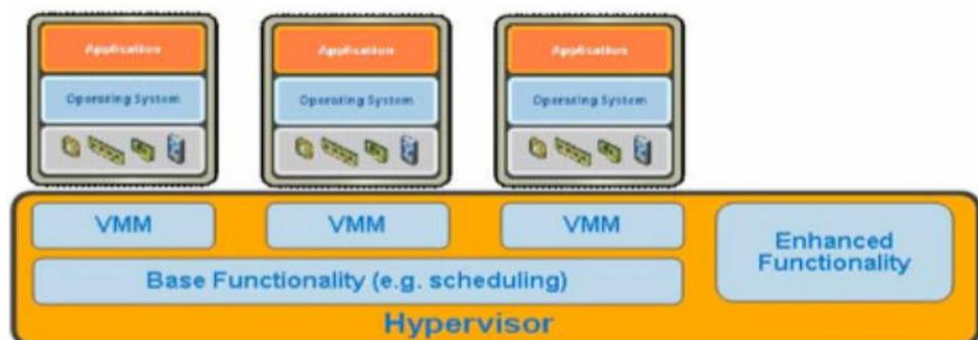


Рис 3.2. Віртуалізація процесора

Гіпервізор має доступ до апаратних ресурсів напряму, а не через ОС, тому він являється ефективнішим розміщеної архітектури, забезпечуючи максимальну масштабованість, надійність та продуктивність системи.

Розглянемо методи, які можна застосовувати для оптимізації ресурсів обладнання шляхом віртуалізації в системах хмарних сервісів.

3.1 Віртуалізація процесора

ОС x86 розроблені для роботи на пустому обладнанні, тому що вони повністю управляють апаратним забезпеченням комп'ютера. Архітектура x86 пропонує 4 рівні привілеїв, так званих кругів 0, 1, 2, 3 для ОС та продуктів для керування доступом до обладнання. Додатки рівня користувача працюють у крузі 3, ОС має доступ безпосередньо до пам'яті та обладнання та виконує свої інструкції в крузі 0. При віртуалізації архітектури x86 шар віртуалізації розміщується під ОС, що знаходиться у кільці 0 для створення VM, які забезпечують спільні ресурси та управління ними (Рис 3.3)

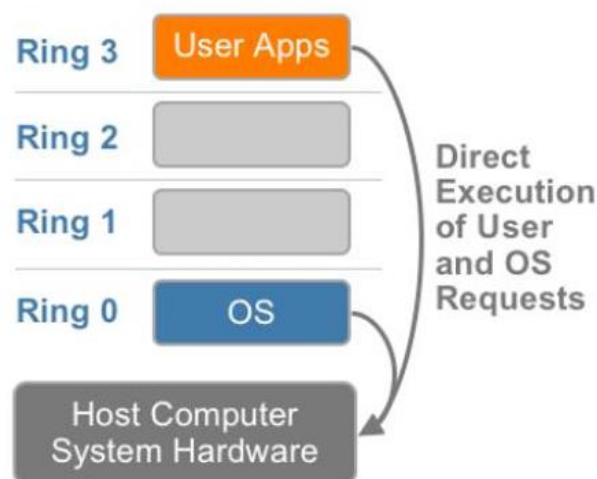


Рис. 3.3 Архітектура x86 рівня привілеїв без віртуалізації

Проблеми неефективної реалізації певних конфіденційних інструкцій можна вирішити методами бінарного перекладу, які запускають монітор віртуальної машини у кільці 0, переміщуючи ОС на круг 3 для користувачів і забезпечуючи ізоляцію та продуктивність ОС.

Існує 3 альтернативні методики для керування конфіденційними та привілейованими інструкціями для віртуалізації процесора на x86:

- Віртуалізація за допомогою апаратного забезпечення;
- Повна віртуалізація із застосуванням бінарного перекладу;
- Віртуалізація ОС з використанням паравіртуалізації чи віртуалізації.

3.1.1 Метод віртуалізації за допомогою апаратного забезпечення

Виробники апаратного обладнання застосовують віртуалізацію та розробляють її нові функції, спрощуючи методи віртуалізації.

Технології Intel-Virtualization та AMD-Virtualization спрямовують пільгові інструкції, надаючи нових функцій процесору, що в свою чергу дозволяє монітору віртуальної машини запускатися в кореневому режимі нижче круга 0. Привілейовані та конфіденційні виклики встановлюються для автоматичної пастки до гіпервізора, усуваючи необхідність бінарного перекладу чи паравіртуалізації (Рис.3.4).

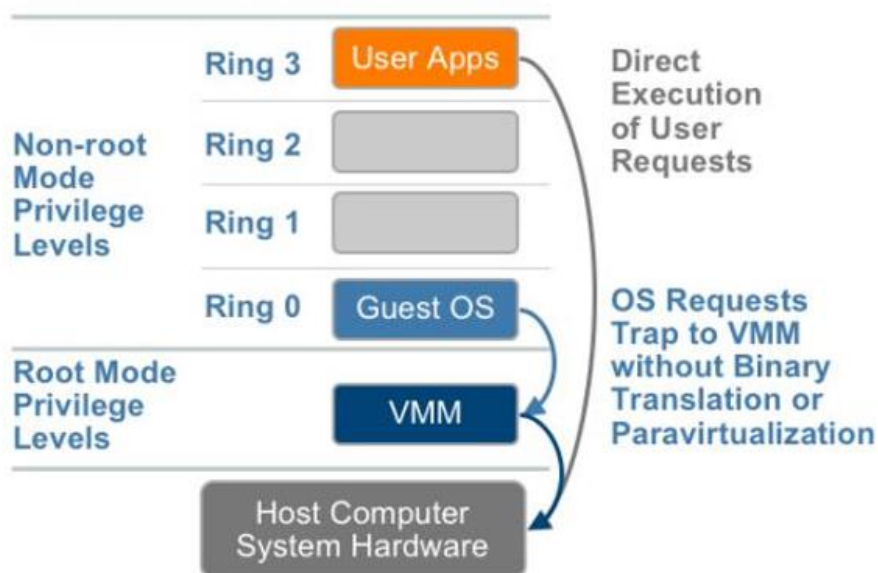


Рис. 3.4 Апаратна віртуалізація x86

3.1.2 Метод повної віртуалізації за допомогою бінарного перекладу

Операційну систему x86 можна віртуалізувати, використовуючи метод комбінації бінарного перекладу та прямого виконання. Цей метод, схематично зображений на Рис. 3.4, переводить код ядра на заміну невіртуалізованих інструкцій новими послідовностями інструкцій, які мають вплив на віртуальне обладнання. Таким чином, код рівня користувача напряду виконується на процесорі для забезпечення продуктивності віртуалізації. Всі монітори віртуальної машини надають всім віртуальним машинам послуги фізичного обладнання: віртуальні пристрої, BIOS, управління пам'яттю.

Така комбінація бінарного перекладу та прямого виконання забезпечує повну віртуалізацію, тому, що відбувається повне абстрагування гостьової операційної від базового обладнання за допомогою шару віртуалізації. Гостьова ОС не потребує змін. Метод повної віртуалізації це єдиний метод, при використанні якого не потребується допомоги апаратного забезпечення чи ОС для віртуалізації привілейованих та конфіденційних інструкцій. Гіпервізор транслює всі інструкції для операційної системи та зберігає результати для їх подальшої реалізації.

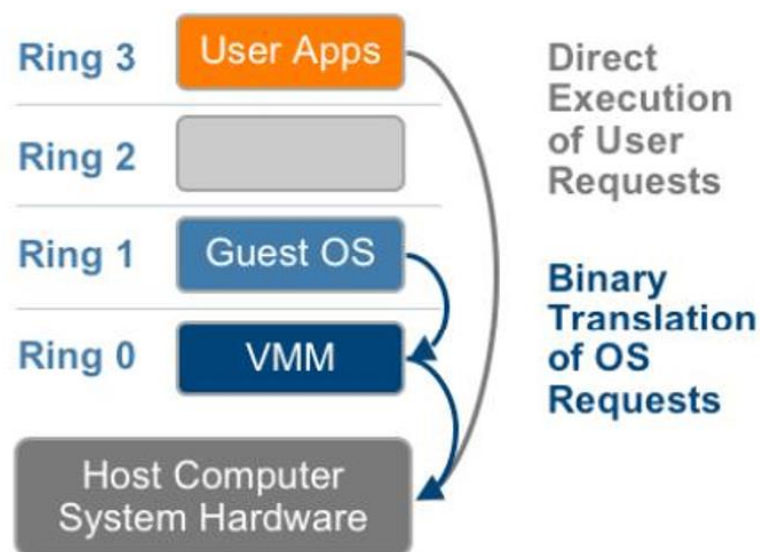


Рис. 3.4 Бінарний підхід до віртуалізації x86

Повна віртуалізація за допомогою бінарного перекладу пропонує найвищий рівень ізоляції та безпеки для VM, та спрощує міграцію та

портативність. Гостьова ОС працює за допомогою віртуалізованого обладнання або на апаратному обладнанні.

3.1.3 Метод паравіртуалізації за допомогою операційної системи

Паравіртуалізація передбачає зміну ядра ОС (Рис. 3.5) для заміни невіртуалізованих інструкцій на гіпервиклики, які напряду взаємодіють з гіпервізором. Гіпервізор надає інтерфейси гіпервиклику для інших операцій з ядром, наприклад, операції з управління пам'яттю, обробки переривань та зберігання часу.

Неповна віртуалізація - підхід, при якому віртуалізується не 100% апаратного забезпечення. Відбувається неповна віртуалізація процесора, тобто за винятком часткової підміни або приховування питань системних викликів, бінарний код віртуальної машини виконується процесором безпосередно.

Повна віртуалізація - підхід, при якому емулюється взагалі все апаратне забезпечення, включаючи процесор. Дозволяє створювати апаратно незалежні середовища, і запускати ОС і прикладне ПЗ для платформи x86 (наприклад, на системах на системах SPARC). Зворотною стороною повної незалежності є високі накладні витрати на віртуалізацію процесора і низька підсумкова продуктивність.

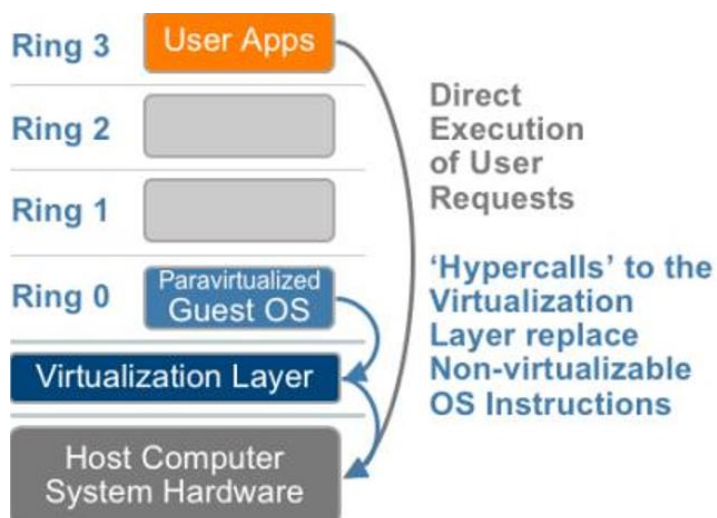


Рис. 3.5 Паравіртуалізаційний підхід до віртуалізації x86

Відмінність паравіртуалізації від повної віртуалізації, в тому, що немодифікована операційна система не знає, що вона є віртуалізується та конфіденційні виклики ОС реалізуються за допомогою двійкового перекладу. Цінність паравіртуалізації полягає в оптимізації витрат на проведення віртуалізації. Також продуктивність паравіртуалізації над повною віртуалізацією може сильно відрізнятись залежно від навантаження. Також паравіртуалізація не підтримує немодифіковані операційні системи, в зв'язку з чим її сумісність та портативність мають слабкі показники. Паравіртуалізація вносить проблеми в підтримку та ремонт середовищ, оскільки такий процес вимагає модифікацій ядра ОС.

3.2 Віртуалізація пам'яті

Критично важливим компонентом для віртуалізації, крім процесора являється віртуалізація пам'яті. Віртуалізація пам'яті включає в себе не лише обмін пам'яттю безпосередньо фізичної системи, а й динамічний розподіл на віртуальні машини. Віртуалізація пам'яті VM має спільні риси з підтримкою віртуальної пам'яті, яку реалізують сучасні ОС. Сучасні процесори x86 включають для оптимізації роботи віртуальної пам'яті такі компоненти як блок управління пам'яттю та буфер перегляду зовнішнього перекладу.

Для запуску віртуальних машин в одній інформаційній системі потрібен достатньо високий рівень віртуалізації пам'яті. Тобто, необхідно здійснювати підтримку блоку управління пам'яттю для підтримки гостьової операційної системи. Гостьова ОС забезпечує контроль відображення віртуальних адрес на фізичні адреси пам'яті, але гостьова ОС не має прямого доступу до фактичної пам'яті віртуальної машини. VMM відповідає за співставлення фізичної пам'яті гостя та фактичної пам'яті машини. Монітор віртуальної машини проводить використання тіньових таблиць сторінок для пришвидшення відображень. Лінією червоного кольору на рис. 3.7 позначено, використання монітором віртуальної машини апарату TLB.

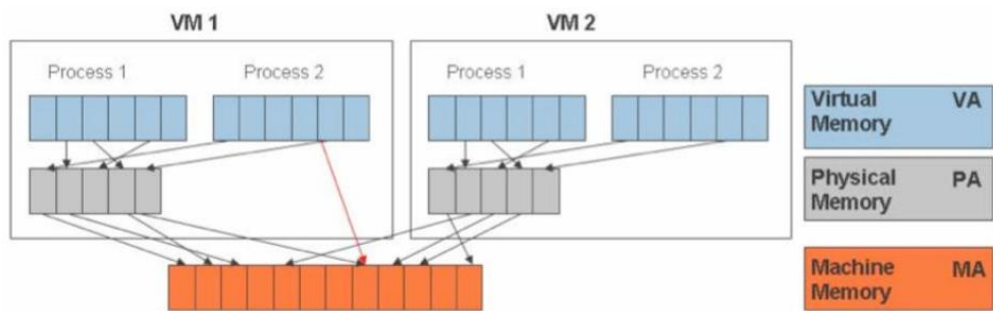


Рис. 3.6 Графічне зображення віртуалізації пам'яті

Для можливості відображення віртуальної пам'яті в пам'яті машини для уникнення при кожному доступі двохрівневого перекладу. При зміні операційною системою віртуальної пам'яті на відображення фізичної пам'яті, монітор віртуальної машини оновлює таблиці тіньових сторінок для увімкнення прямого пошуку.

3.3 Віртуалізація пристроїв введення/виведення

Завершальним компонентом, після віртуалізації процесора та пам'яті, являється віртуалізація пристроїв введення/виведення. Пристрої введення/виведення беруть на себе керування маршрутизацією запитами введення/виведення між фізичним обладнанням та віртуальними пристроями.

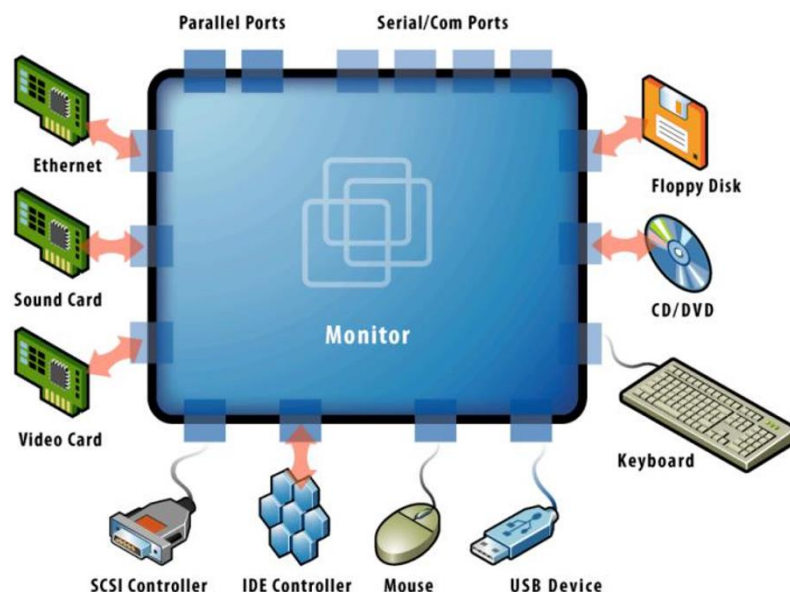


Рис. 3.7 Віртуалізація пристроїв вводу/виводу

Віртуалізація та управління пристроями введення/виведення, засновані на ПЗ, забезпечує великий набір функцій та спрощене керування на відміну від прямого переходу на апаратне забезпечення.

Після віртуалізації фізичне обладнання гіпервізор представляє кожній VM набір стандартизованих віртуальних пристроїв (Рис. 3.8), які ефективно емулюють обладнання та перемикають запити VM на апаратне забезпечення інформаційної системи. Завдяки цій стандартизації на послідовних драйверах пристроїв стандартизується віртуальна машина та перенесення на платформи, оскільки всі VM незалежно від реального фізичного обладнання в системі, мають налаштування для роботи на одному і тому ж віртуальному обладнанні.

4 ЗАСТОСУВАННЯ МЕТОДУ АНАЛІЗУ СУМІСНОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА ОСНОВІ МЕРЕЖЕВОГО ОБЛАДНАННЯ

У цій частині пропонується розглянути концепцію сумісності прикладного програмного забезпечення та отримати розуміння його впливу на високу надійність застосування інформаційної системи. У цьому розділі розглядаються питання типової сумісності програмного забезпечення. Також висвітлено метод аналізу сумісності програмного забезпечення на основі апаратної віртуалізації. Проводиться аналіз сумісність двох типових програм, призначених для захисту інформації та оцінка отриманих результатів.

Поняття сумісності програмного забезпечення це характеристика ПЗ та його компонентів, яка визначає їх здатність взаємодіяти між собою та працювати під управлінням однієї ОС та на одній апаратній платформі. Робота програм може страждати через конфлікт правил та алгоритмів, що призводить до зниження продуктивності та в окремих критичних випадках навіть до краху хост-операційної системи.

Сумісність програмного забезпечення має значний та безпосередній вплив на ступінь надійності інформаційних систем. Особливо це стосується складних інформаційних систем, які потребують підвищеного рівня безпеки. Завдяки динамічному розвитку та швидкозмінним тенденціям розвитку, такі інформаційні системи мають у своєму складі велику кількість програмних додатків та вимагають розгортання системи інформаційної безпеки, до складу якої входить антивірусне ПЗ та ПЗ для управління хостом. При розробці деяких програмних продуктів виробниками може не враховуватися фактор сумісності, через що виникає проблема ризику для надійності інформаційної системи. Оцінка та вивчення можливості конфлікту між компонентами ПЗ, з врахуванням специфічних особливостей, призначення, функціональності, гнучкості, структури системи уникнути потенційної проблеми та забезпечити високу продуктивність та надійний рівень безпеки.

4.1. Аналіз проблем сумісності програмного забезпечення

Для виникнення несумісності програмного забезпечення існує багато причин. Розглянемо найбільш важливі та розповсюджені:

- Конкуренція ресурсів. Виникає за умов використання одного і того ж системного ресурсу в кількох різних додатках. Наслідки зазвичай взаємні, відбуваються при наявності збою в системі, при використанні невірної графіки. Як приклад можна привести такий сценарій: конфлікт роботи антивірусного ПЗ, яке вимагає фільтрації читання та запису реєстру, та робота ПЗ для управління хостом, що теж контролювати читання та запис реєстру обов'язково приведуть до зачеплення такого ж положення системного ядра.

- Незавершеність внутрішньої логіки обробки. Наприклад, певне ПЗ забезпечує роботу пакету шаблонів проектів для моніторингу реєстру. У випадку, якщо будь-яка програма викликає API для запису чи зчитування реєстру, його алгоритм реалізується в модулі ядра цього ПЗ безпеки. Тому ПЗ для безпеки крім перевірки повноважень API, необхідно здійснити перевірку параметрів. За такого сценарію системне ядро може вийти з ладу.

- Несумісність середовищ. Середовища розгортання та тестування ПЗ значно відрізняються. Основні відмінності середовищ полягають в програмному та апаратному забезпеченні, мережевих середовищах, ОС тощо. Наявність таких відмінностей призводять до виявлення проблеми сумісності, яку було не знайдено під час проходження тестування ПЗ.

- Зміна продуктивності системи. При запуску ПЗ може надати частковий вплив на зміну функціональності системи в цілому. Як приклад можна привести заборону системою виклику певних API, заборону запуску ПЗ чи заборону створення необхідних портів.

В зв'язку зі значними темпами розвитку технологій хмарних обчислень, методи віртуалізації обладнання починають застосовувати при дослідженнях в галузі інформаційної безпеки. Особливо актуальною є

використання віртуалізації при виявленні зловмисного коду, перевірці довірених програм для обчислень тощо.

Такий метод передбачає створення віртуальних машин за допомогою апаратної віртуалізації та програми моніторингу та аналізу сприятливої поведінки та ПЗ для забезпечення захисту пам'яті. Принцип даної платформи був розроблений Академією наук КНДР у 2016 році (Рис. 4.1).

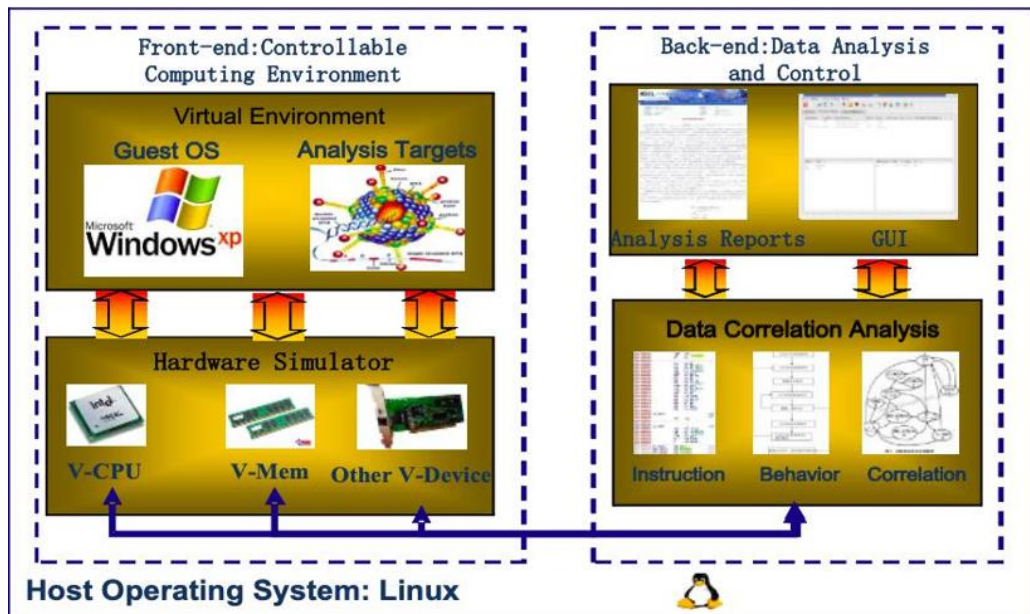


Рис. 4.1. Побудова платформи для динамічного аналізу сумісності ПЗ

Згідно принципу цієї платформи сценарій аналізу сумісності розпочинається з аналізу попереднього стану виконання процесів, стану модулів та потоків, стану цілісності структури даних ОС, стану модуля та стану ядра, а також стану роботи файлів, мережі, оцінки роботи реєстру, системного обслуговування та інших дій програми, пов'язаних з процесом. Після завершення аналізу процесів ресурсів, що використовуються в процесі, а також базових методів та результатів їх впливу на загальну цілісність інформаційної системи, отримані дані можуть бути передані для оцінювання конфлікту між додатками та рівня можливості конкуренції за ресурси (Рис 4.2).

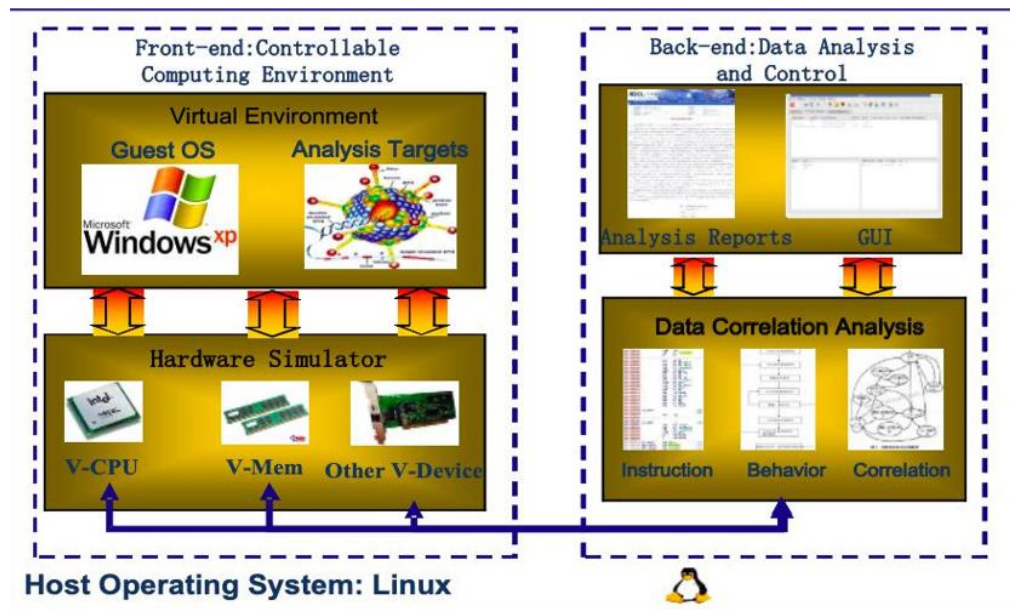


Рис. 4.2 Методи отримання даних та аналізу ПЗ

Етапи отримання даних та аналізу ПЗ:

- Етап 1. Відбувається процес збору даних про стан системи. На основі отриманих даних проводиться аналіз ключових даних ядра, формуються запуснені процеси, аналізується стан та шлях виконання процесів;
- Етап 2. Проводиться процес збору технологічних даних. За допомогою додатку для моніторингу процесів в інформаційних системах проводиться аналіз використаних ресурсів пам'яті, аналіз стану відкритих файлів та реєстру та модуль, через який надаються дані для аналізу сумісності.
- Етап 3. Запускається процес збору даних про ресурси. Відстежується шлях реалізації цільової програми, динаміка використання фізичної пам'яті, контролюється розподіл пам'яті віртуальної. На основі зібраних даних проводиться оцінка впливів ПЗ на систему.
- Етап 4. Проводиться збір даних щодо дій, які відбуваються під час процесу. На даному етапі шляхом моніторингу інструментів, що лежать в основі API, наприклад, запису часу, перехоплення керованого виклику, визначення параметрів позиції та середнього значення повернення

відбувається семантичний аналіз отриманих даних, методів та ресурсів методи, що були задіяні.

- Етап 5. Відбувається процес збору даних щодо цілісності системи. Аналізується структура ядра ключів ОС, код ключового ядра, запускається процес зворотнього аналізу виконаного ключа модуля та формату файлу модуля. Відображається стан і формат коду виконання та загальний вплив цільової програми на ОС. При цьому аналізу піддаються модифікації API ключового ядра, адреси SSDT та Shadow SSDT. Здійснити оцінювання можна шляхом порівняння з ключових даних в ядрі. Також дозволяється визначити спосіб, джерело та зміст модифікації.

- Етап 6. Проводиться збір даних про діяльність процесу. Відбувається процес перехоплення процесу функціонування, реєстром та файлами цільової програми, а також зворотнього аналізу структури додатку для моніторингу процесів в інформаційних системах. Аналіз системи обробки та аналіз роботи цільової програми дозволяє отримати дані для оцінювання конфлікту ПЗ через наявну конкуренцію за ресурси інформаційної системи.

4.2. Дослідження рівня забезпечення безпеки типовим програмним забезпеченням.

Серед основних процесів життєвого циклу ПЗ можна виділити етапи інсталяції, запуску і видалення ПЗ.

На етапі встановлення програма проводить копіювання файлів в систему, налаштування інформаційного середовища, та отримання доступу до ресурсу та системного реєстру. У випадку зміни чи задіяння будь-якого пов'язаного файлу, виникне проблема із встановленням ПЗ. Тобто, на етапі інсталяції аналіз спрямований на файли та реєстри. Використовуються інструменти для завантаження та аналізу файлів, такі як WinDbg, SysTracer, ProcMon, що проводять аналіз щодо наявності змінених файлів, папок, записів реєстру, встановлених програм та служб, програм для запуску.

При запуску відбувається старт функцій моніторингу, підключення і перехоплення. Конфлікти можуть бути виявлені у цілісності системи. На даному етапі аналіз конфлікту базується на результатах ручної перевірки цілісності основних даних та цілісності модуля ядра системи з використанням інструментів для завантаження та аналізу файлів таких як Wookon, XueTr, WinDbg, які допомагають впровадити аналіз впливу на цілісність системи програми, що була запущена.

Системний модуль, реєстри та файли видаляються на етапі видалення ПЗ. Потенційний конфлікт може виникнути в роботі реєстрів та файлів. Ключовим підходом є аналіз ресурсу, який не було видалено повною мірою.

Результат аналізу етапів інсталяції, запуску та видалення подані в наступних таблицях:

Таблиця 4.1.

Результат аналізу етапу інсталяції

Вміст	Антивірусне ПЗ	ПЗ для моніторингу хостів
Файлові операції	2493 створення 7378 модифікацій 22 видалення 19 перейменувань	129 створень
Реєстр	129 доповнень 346 модифікацій 3 вилучення ключового значення	54 доповнення 279 модифікації 4 вилучення елементів
Процес	9 створень 7 видалень	3 створення

Аналіз результатів на етапі запуску

Вміст	Антивірусне ПЗ	ПЗ для моніторингу хостів
Модуль	Завантаження 9 модулів	Завантаження 58 модулів
Процес	Завантаження 7 процесів	Завантаження 3 процесів
Цілісність	Завантаження 10 драйверів 35 SSDT функцій 1 диспетчерська функція 12 повідомлень Зміна 137 байтів ядра Реєстрація 7 функцій системного виклику Використання 2 портів Створення 3 фільтрів введення віртуального обладнання	Завантаження 9 драйверів 4 SSDT функцій Реєстрація 6 функцій системного виклику Використання 4 портів Створення 2 одиниць віртуального обладнання
Цілісність	Додавання 3 елементів запуску Додавання 5 служб запуску	Додавання 5 елементів запуску 1 послуга запуску

Згідно отриманих результатів очевидно, що в основному антивірусне ПЗ та ПЗ для моніторингу хостів задіювали власні ресурси. Під час встановлення обидвох додатків одночасно, перекриття функцій призводило до заповнення ресурсів. Обидва ПЗ, наприклад, підключили ідентичні функції SSDT: NtCreateKey, NtDeleteKey, NtDelete-ValueKey, NtOpenKey, NtQueryValueKey, NtSetValueKey [10].

Взявши за основу результати проведеного аналізу можемо сформулювати конкретні рекомендації для уникнення проблеми сумісності.

На етапі запуску ПЗ необхідне максимально можливе збільшення системного ресурсу для зниження рівня проблеми високої конкуренції за ресурси. При роботі з певними системами, коли антивірусне ПЗ та ПЗ для моніторингу хостів потрібно запускати одночасно, об'єм системної пам'яті

повинен бути не меншим від рекомендованого значення - 4 Гб. Для забезпечення стабільної ефективності та сумісності потрібно за можливості виключати використання зайвого ПЗ. Налаштування "білого списоку" обох програмних продуктів повинен бути проведено таким чином, щоб забезпечити їх безконфліктну взаємодію.

Таблиця 4.3.

Результат аналізу після етапу видалення

Вміст	Антивірусне ПЗ	ПЗ для моніторингу хостів
Модуль ядра	4 існували, 5 існували	5 існували після перезавантаження
Процес	Усі існували перед запуском	Усі існували перед запуском
Файл та реєстр	Залишок перед запуском	Залишок перед запуском
Пункт запуску	4 видалено повністю після перезавантаження	5 видалено повністю після перезавантаження
Системне обслуговування	Усі існували перед запуском	Усі існували перед запуском
Ядро	Скасовано після перезавантаження	Скасовано після перезавантаження
Таймер	Скасовано перед запуском	Скасовано перед запуском
Видалення системного виклику	6 видалено повністю після перезавантаження	5 видалено повністю після перезавантаження
Порт	Усі випущені після перезавантаження	Усі випущені після перезавантаження
Драйвер фільтрування	3 видалено повністю після перезавантаження	8 видалено повністю після перезавантаження

На етапі розробки ПЗ потрібно з обережністю проводити використання неопублікованих технічних інструментів та засобів. Особливості їх використання повині бути детально задокументованими для можливості використання цієї інформації при виникненні проблем із сумісністю. Під час розробки, таких складових як мережевий порт, файл чи реєстр, системний

хук, рекомендовано не використовувати ресурси, що уже задіяні антивірусним ПЗ та ПЗ для моніторингу хостів.

Необхідно враховувати погіршення продуктивності, яке було спричинене ПЗ для забезпечення безпеки. Обов'язковою є попередня перевірка пропускнуої здатності системи. Беручи до уваги той факт, що програмному забезпеченню для забезпечення безпеки для оновлення даних та зв'язку необхідно здійснювати використання частини потоку даних, потрібно враховувати необхідність відповідних налаштувань механізму передачі даних.

До рекомендацій можна додати використання допоміжних засобів аналізу сумісності. Коли немає проблеми з сумісністю, Налаштування ОС повинні забезпечити організацію режиму збереження даних при системних збогах як "збереження всіх у пам'яті". При активації цього режиму всі дані, збережені в пам'яті, матимуть можливість бути збереженими також на жорсткому диску і бути доступними для читання та аналізу при знаходженні причин можливих проблем з сумісністю. [9].

Проведення аналізу наявних проблем сумісності потребує часу, досвіду та розуміння процесів, тому на етапі виникнення проблем із сумісністю, сценарій розвитку подій необхідно зберегти та захистити для проведення подальшого детального аналізу.

ВИСНОВКИ

З розвитком сучасних засобів обробки, передавання та приймання цифрової інформації зростає попит не тільки на потужні розрахункові рішення, але і на програмне забезпечення, метою якого є оптимальне використання наявних ресурсів системи. Одним з шляхів оптимального використання ресурсів телекомунікаційного обладнання є розміщення сервісів та послуг провайдера на віддалених серверах, а виконання відповідних обчислювальних операцій – здійснювати за допомогою хмарних розрахунків. Таким чином, користувач смарт девайсу отримує одразу готовий результат і не витрачає ресурси гаджету на виконання тієї чи іншої математичної операції.

Широкої популярності дана модель взаємодії між абонентом та провайдером знаходить своє застосування в мережах п'ятого покоління. Значно зручніше використовувати сервер, потужності якого вистачить для обслуговування сотень пристроїв в режимі онлайн одночасно, ніж використовувати ресурси смартфона та персоналізовано вирішувати задачу зобробки мультимедійних даних.

Яскравим прикладом віртуалізації обладнання (перенесення розрахунків в хмарне середовище) на сьогоднішній день є відеохостінги. Після завантаження відео файлу користувач має можливість редагувати його, накладати ефекти або навіть конвертувати в режимі реального часу використовуючи розрахункові потужності сервісу. Подібні можливості поширюються майже на всі сучасні цифрові послуги, а популярність подібних сервісів лише зростає.

Розглянуті в роботі методи переходу до хмарних обчислень дозволяють зменшити навантаження як на телекомунікаційні системи, так і на абонентськ обладнання, що дозволить зекономити ресурси та час на виконання математичної операції, і, відповідно, надати послуги більшій кількості абонентів. З розглянутих методів віртуалізації можливо зробити

висновок по те, що для досягнення ефективного результату необхідно не лише збільшувати потужності серверного обладнання, а і вдосконалювати програмне забезпечення, з метою оптимального використання наявних ресурсів.