ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ ТА МЕРЕЖ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «**МЕТОДИКА ВДОСКОНАЛЕННЯ РОБОТИ ЦОД НА БАЗІ ЗАСТОСУВАННЯ ПРОГРАМНО-КОНФІГУРОВАНИХ МЕРЕЖ**»

на здобуття освітнього ступеня магістра
зі спеціальності 172 Електронні комунікації та радіотехніка
(код, найменування спеціальності)
освітньо-професійної програми Телекомунікаційні системи та мережі
(назва)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Ростислав ТЕТЕРЯ

_____ _____
(*підпис*)

Виконав: Здобувач вищої освіти гр.ТСДМ-63

Ростислав ТЕТЕРЯ

Керівник: Костянтин ЛАВРІНЕЦЬ
ктн, доцент

Рецензент:

Київ 2025

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут телекомунікацій

Кафедра Телекомунікаційних систем та мереж
Ступінь вищої освіти Магістр
Спеціальність 172 Електронні комунікації та радіотехніка
Освітньо-професійна програма Телекомунікаційні системи та мережі

ЗАТВЕРДЖУЮ
Завідувач кафедри Телекомунікаційних систем та мереж
_____Віктор ЗАЇКА
«____» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Тетері Ростиславу Валерійовичу
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: **:** «Методика вдосконалення роботи ЦОД на базі застосування програмно-конфігурованих мереж»
керівник кваліфікаційної роботи Костянтин ЛАВРІНЕЦЬ, ктн, доцент
(Ім'я, ПРІЗВИЩЕ науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «30» жовтня 2025 р. № 467.

2. Строк подання кваліфікаційної роботи «___» _____ 2025 р.
3. Вихідні дані до кваліфікаційної роботи:

1. Організаційно-технічні відомості про SDNрішення для мереж ЦОД
2. Структура мережі Clos
3. Принцип інкапсуляції VXLAN
4. Науково-технічна література.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Організаційно-технічні відомості про програмно-конфігурованірішення мереж, що застосовуються в ЦОД
2. Дослідження та порівняння сучасних програмно-конфігурованих рішень для мереж ЦОД провідних вендорів

5. Перелік ілюстративного матеріалу**:** Презентація на слайдах

6. Дата видачі завдання «___» _____ 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Підбір та аналіз науково-технічної літератури	10.10.2025	Викон.
2	Організаційно-технічні відомості про програмно-конфігурованірішення мереж, що застосовуються в ЦОД	20.10.2025	Викон.
3	Дослідження та порівняння сучасних програмно-конфігурованих рішень для мереж ЦОД провідних вендорів	31.10.2025	Викон.
4	Висновки, вступ, реферат	30.11.2025	Викон.
5	Розробка презентації	15.12.2025	Викон.

Здобувач вищої освіти Ростислав ТЕТЕРЯ

(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник
кваліфікаційної роботи Костянтин ЛАВРІНЕЦЬ

(підпис) *(Ім'я, ПРІЗВИЩЕ)*

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут телекомунікацій

ПОДАННЯ
ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ
ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ
на здобуття освітнього ступеня магістра

Направляється здобувач Тетеря Р.В. до захисту кваліфікаційної роботи
(прізвище та ініціали)

за спеціальністю 172 Електронні комунікації та радіотехніка
(код, найменування спеціальності)
освітньо-професійної програми Телекомунікаційні системи та мережі
(назва)
на тему: «Методика вдосконалення роботи ЦОД на базі застосування програмно-конфігурованих мереж».

Кваліфікаційна робота і рецензія додаються.

Директор ННІТ _____ Владислав КРАВЧЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Висновок керівника кваліфікаційної роботи

Здобувач Тетеря Ростислав Валерійович показав гарну теоретичну та інженерну підготовку, уміння володіти новими комп'ютерними технологіями, користуватися навчальною, довідковою і науково-технічною літературою. Працюючи над завданнями, які були поставлені керівником, проявив ініціативність, сумлінність та хист до наукової роботи.
Все це дозволяє оцінити виконану кваліфікаційну роботу здобувача Тетері Ростислава Валерійовича на оцінку «**добре**» та присвоїти йому кваліфікацію Магістр з електронних комунікацій та радіотехніки.

Керівник роботи Костянтин ЛАВРІНЕЦЬ
(Ім'я,ПРІЗВИЩЕ)

«____» _____ 2025 року

Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувач Тетеря Р.В. допускається до захисту даної роботи в Екзаменаційній комісії.

Завідувач кафедрою Телекомунікаційних систем та мереж Віктор ЗІЇКА
(назва) *(підпис)* *(Ім'я, ПРІЗВИЩЕ)*

ВІДГУК РЕЦЕНЗЕНТА
на кваліфікаційну роботу
на здобуття освітнього ступеня магістра

здобувача вищої освіти Тетері Ростислава Валерійовича
(прізвище, ім'я, по батькові)
на тему Методика вдосконалення роботи ЦОД на базі застосування програмно-конфігурованих мереж «»

Актуальність: Тема роботи є актуальною, тому що у сучасному світі бурхливо збільшується кількість хмарних сервісів, що потребує нового підходу у побудові хмар.

Позитивні сторони : Була проаналізована методика побудови SDN рішень для мереж ЦОД, що дає підгрунтя для модернізації та розширення існуючих телекомунікаційних мереж з застосуванням технології інкапсуляції VXLANy поєднані з контролером SDN. Досвід який представлено у роботі підтверджує необхідність впровадження нових технологій на підприємствах України для отримання можливості зниження експлуатаційних витрат, та підвищенні ефективності роботи мереж ЦОД.

Недоліки: 1. Не розглянуті стандартизовані рішення побудови мереж ЦОД з використанням SDN контролеру.
2.Поза розглядом залишилися питання традиційних мереж під управлінням SDN контролера.

Відзначені зауваження не впливають на загальну позитивну оцінку магістерської кваліфікаційної роботи

Висновок: *кваліфікаційна робота заслуговує оцінку "***добре***", а здобувач* Тетеря Ростислав Валерійович *заслуговує присвоєння кваліфікації:* Магістр з електронних комунікацій та радіотехніки.

Рецензент:

науковий ступінь, вчене звання _____ _____

(*підпис)* *Ім'я, ПРІЗВИЩЕ*

РЕФЕРАТ

Текстова частина магістерської кваліфікаційної роботи на здобуття освітнього ступеня магістр:____ стор., __ рис.,____ табл.,_____ джерел.

Мета роботи – дослідження вдосконалення роботи телекомунікаційних мереж ЦОД на основі сучасних рішень.

Предмет дослідження –телекомунікаційна мережа ЦОД.

Об`єкт дослідження – побудова моделі SDN мережі ЦОД з інкапсуляцією VXLAN.

У дипломній роботі розглянуто організаційно-технічні відомості про SDN мережі ЦОД. Розглянуто питання структури системи мережі. Також було розглянуто принцип передачі трафіку на основі інкапсуляції VXLAN у поєднанні з EVPN.

КЛЮЧОВІ СЛОВА:

SDN, VXLAN, EVPN, DC, VM, VTEP.

ABSTRACT

Text part of the master's qualification work for the degree of master:____ pages, __ figures,____ tables,_____ sources.

The purpose of the work is to study the improvement of the operation of telecommunication networks of data centers based on modern solutions.
The subject of the study is the telecommunications network of the data center.
The object of the study is the construction of the SDN model of the data center network with VXLAN encapsulation.
The thesis examines the organizational and technical information about SDN of data center networks. The issues of network system structure are considered. The principle of traffic transmission based on VXLAN encapsulation in combination with EVPN was also considered.

KEYWORDS:
SDN, VXLAN, EVPN, DC, VM, VTEP.

ЗМІСТ

ВСТУП 9

1 ОРГАНІЗАЦІЙНО-ТЕХНІЧНІ ВІДОМОСТІ ПРО ПРОГРАМНО-КОНФІГУРОВАНІ РІШЕННЯ МЕРЕЖ, ЩО ЗАСТОСОВУЮТЬСЯ В ЦОД ..... **Помилка! Закладку не визначено.**1

1.1 Площина передачі трафіку ..... **Помилка! Закладку не визначено.**1

1.2 Зв’язок між overlay та underlay ... **Помилка! Закладку не визначено.**5

1.3 Типи underlay мереж ..... **Помилка! Закладку не визначено.**6

1.4 Типи overlay мереж ..... **Помилка! Закладку не визначено.**1

1.5 Розуміння потреби у VXLAN..... **Помилка! Закладку не визначено.**3

1.6 Основи VXLAN ..... **Помилка! Закладку не визначено.**5

1.7 Варіанти VXLAN..... **Помилка! Закладку не визначено.**8

1.8 Апаратна частина VTEP 30

1.9 EVPN..... **Помилка! Закладку не визначено.**2

2 ВИЗНАЧЕННЯ SDN ..... **Помилка! Закладку не визначено.**9

2.1 Парадигма та додатки SDN ..... **Помилка! Закладку не визначено.**1

2.2 Комутатори SDN ..... **Помилка! Закладку не визначено.**4

2.3 Контролери SDN..... **Помилка! Закладку не визначено.**6

2.4 Централізація управління в SDN . **Помилка! Закладку не визначено.**7

2.5 Менеджмент трафіку..... **Помилка! Закладку не визначено.**1

2.6 Інтерфейси програмування SDN.. **Помилка! Закладку не визначено.**3

2.7 Мережі Центрів Обробки Даних.. **Помилка! Закладку не визначено.**7

3 ДОСЛІДЖЕННЯ ТА ПОРІВНЯННЯ СУЧАСНИХ ПРОГРАМНО-КОНФІГУРОВАНИХ РІШЕНЬ ДЛЯ МЕРЕЖ ЦОД ПРОВІДНИХ ВЕНДОРІВ..... **Помилка! Закладку не визначено.**9

60

2.2 Aruba Fabric Composer ..... **Помилка! Закладку не визначено.**0

3.3 Apstra..... **Помилка! Закладку не визначено.**1

4.4 Матриця порівняння рішень..... **Помилка! Закладку не визначено.**2

ВИСНОВКИ ..... **Помилка! Закладку не визначено.**4

ПЕРЕЛІК ПОСИЛАНЬ ..... **Помилка! Закладку не визначено.**5

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ……..…66

ВСТУП

На сьогодні спостерігається невпинний та експоненціальний ріст інформатизації у світі. Значною мірою збільшується кількість хмарних сервісів. Беручи до уваги, що обсяг глобального ринку хмарних обчислень у 2023 році оцінювався в 405,65 мільярдів доларів США. Очікується, що ринок зросте з 480,04 мільярдів доларів США у 2024 році до 1712,44 мільярдів доларів США до 2029 року, демонструючи CAGR 19,9% протягом прогнозованого періоду. Глобальна пандемія COVID-19 була безпрецедентною та приголомшливою, оскільки попит на хмарні обчислення в усіх регіонах був вищим, ніж очікувалося, порівняно з рівнем до пандемії. Згідно з аналізом джерела, світовий ринок продемонстрував зростання на 13,8% у 2020 році порівняно з 2019 роком. [1]

Хмарні обчислення — це техніка обробки, зберігання та керування даними в мережі віддалених комп’ютерів, розміщених в Інтернеті, а не на персональному комп’ютері чи локальному сервері. Зокрема, це стосується спільної області зберігання, яка дозволяє всім мережевим пристроям отримувати доступ до даних одночасно. Хмарні обчислення – це підхід до надання IT-послуг, за якого сторонні постачальники послуг використовують Інтернет для надання обчислювальних ресурсів і програмних засобів. У цій моделі обслуговування клієнт платить лише за час, який він проводить за комп’ютером, а також за пам’ять і пропускну здатність, яку він використовує.

Місце де відбуваються хмарні обчислення називають ЦОД.Для організації хмари в ЦОД необхідно підготувати мережеву інфраструктуру.Мережева інфраструктура сучасного ЦОД повинназадовольняти потреби бізнесу, відповідаючи сучасним вимогам:

1. Інформаційна безпека сервісу;

2. Відмовостійкість та висока доступність;
3. Масштабованість системи;
4. Автоматизація системи.

Як відомо сучасні проблеми потребують сучасних рішень, а тому для задоволення потребі бізнесу світові виробники мережевого обладнання такі як: Cisco, Juniper Networks, Aruba Networksta та ін. пропонують власні SDN рішення для ЦОД. Цей проект пропонує огляд архітектури, технологій та протоколів, що застосовуються в SDN рішеннях ЦОД, а також пропонується порівняння класичної «legacy» VLAN архітектури з сучасною VXLANSDN фабрикою.

1 ОРГАНІЗАЦІЙНО-ТЕХНІЧНІ ВІДОМОСТІ ПРО ПРОГРАМНО-КОНФІГУРОВАНІ РІШЕННЯ МЕРЕЖ, ЩО ЗАСТОСОВУЮТЬСЯ В ЦОД

1.1 Площина передачі трафіку

В 2010-тих роках поширюються хмарні технології та віртуалізація серверів. Їх зріст відбувається з вибуховою швидкістю.[2] Кількість сервісів в межах одного ЦОД може досягати декількох тисяч. Це створює нові виклики для мережевої інфраструктури. Тому на ринок виходить нова «накладена» (англійською *overlay*)технологія, що здатна змінити правила гри. Це VXLAN.

Що таке VXLAN? VXLAN — це протокол інкапсуляції, який забезпечує підключення до центру обробки даних за допомогою тунелювання для розтягування з’єднань рівня 2 через базову IPмережу. У центрах обробки даних VXLAN є найпоширенішим протоколом для створення накладених мереж, які розташовані поверх фізичної мережі, що дозволяє використовувати віртуальні мережі. Протокол VXLAN підтримує віртуалізацію мережі центрів обробки даних, одночасно задовольняючи потреби центрів обробки даних із кількома клієнтами, забезпечуючи необхідну сегментацію у великому масштабі.Яку проблему вирішує VXLAN? За останнє десятиліття центри обробки даних стрімко розширили віртуалізацію своїх серверів, що призвело до значного підвищення оперативності та гнучкості. Віртуалізація мережі та відокремлення віртуальної мережі від фізичної мережі полегшує керування, автоматизацію та організацію. [3]

Віртуалізація — це, по суті, абстракція фізичних ресурсів для підтримки мультиарендації, що дозволяє створювати повністю ізольовані та розподілені робочі середовища. Можна віртуалізувати різні обчислювальні ресурси, такі як операційні системи, пристрої зберігання даних або мережеві компоненти пропонують кращу масштабованість і гнучкість. У сучасних архітектурах центрів обробки даних віртуалізація серверів дозволила

організаціям використовувати консолідовану обчислювальну інфраструктуру. В результаті цього консолідація програмних додатків, операційних систем і апаратних платформ, менше необхідні фізичні обчислювальні ресурси, що забезпечують величезну економічну вигоду та багато іншого переваги.

За допомогою віртуалізації сервера гіпервізор сервера забезпечує абстракцію програмного забезпечення для обчислювальних ресурсів шляхом реплікації фізичних атрибутів сервера в програмному забезпеченні, дозволяючи віртуальним машинам бути створені за допомогою цих атрибутів. Використовуючи однакові апаратні ресурси, можна створити кілька різних екземплярів операційних систем, розміщених у віртуальних машинах (гостьових ОС). Віртуалізація мережі, де рівень віртуалізації функціонально еквівалентний мережевому гіпервізору, забезпечує абстракцію програмного забезпечення для мережевих ресурсів шляхом реплікації мережевих атрибутів (наприклад, маршрутизація, комутація, брандмауер/мікросегментація, балансування навантаження тощо), що дозволяє створювати віртуальні мережі за допомогою цих атрибутів. Використовуючи ті самі фізичні ресурси мережі, можна створити кілька віртуальних мереж для підтримки кількох орендарів.

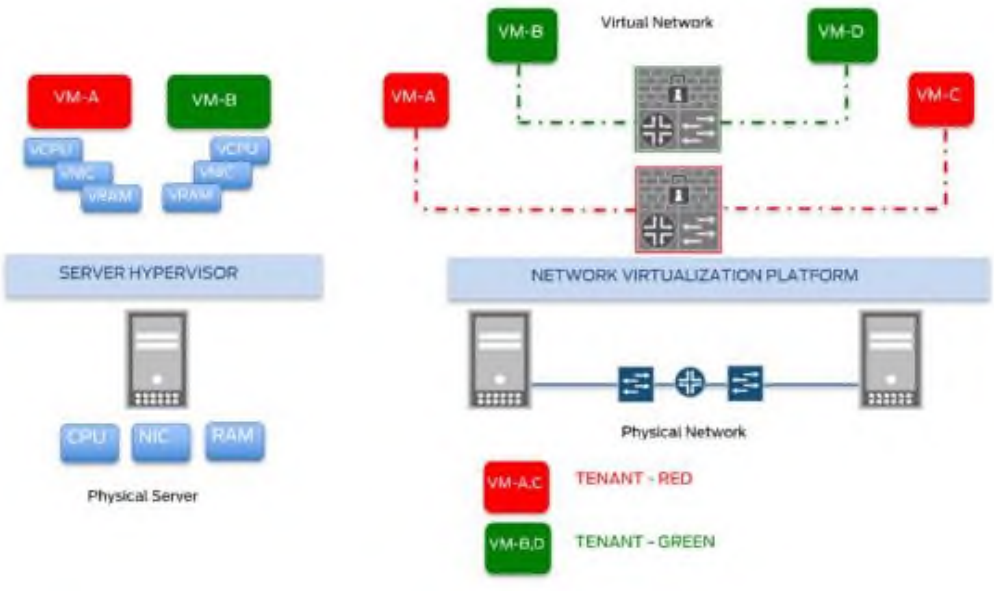


Рис. 1.1 Серверна та мережеві інфраструктури

Однак центри обробки даних з віртуалізацією серверів без віртуалізації мережі повинні зіткнутися з багатьма проблемами мережі. Віртуалізація мережі пропонує багато переваг для полегшення інтеграції віртуальних і фізичних мереж. Деякі з них включають: автоматизацію для полегшення надання, еластичність для підтримки динамічних змін і повного використання розгорнутих ресурсів, безпека через повну ізоляцію сегментів мережі та централізоване управління як віртуальною, так і мережевою інфраструктурою.

Мережеві накладення та віртуалізовані мережеві служби є важливими компонентами віртуалізації мережі. Віртуальні локальні мережі (VLAN) традиційно дозволяли використовувати кілька орендарів шляхом абстрагування локальної мережі сегменту для розділення мережі. Кожна VLAN відповідає унікальній IP-підмережі (окремій широкомовний домен). У застарілих моделях центрів обробки даних VLAN логічно еквівалентна орендарю або послузі, тому, наприклад, розміщення 10 орендарів було б еквівалентним забезпеченню 10 різних VLAN. Транкові з’єднання VLAN через інфраструктуру фізичної мережі з’єднують сервери щоб забезпечити зв’язок програм, які знаходяться на віртуальних машинах або контейнерах. Із зростаючими вимогами до програм – будь то для хмарних обчислень чи ініціатив щодо великих даних – масштабованість, еластичність і легкість ініціалізації в гібридних (фізичних і віртуальних) середовищах стали критичними вимогами до центру обробки даних. З використанням VLAN, мережевих ресурсів стати вузьким місцем у реалізації цих вимог.

Тут описано деякі проблеми, які виникають із застарілими невіртуалізованимиархітектурами центрів обробки даних які не використовують overlayрішення:

- Відсутність гнучкості та жорсткого забезпечення: надання нових орендарів або внесення змін до існуючих потребує змін як у віртуальній, так і у фізичній інфраструктурі (починаючи від гіпервізора сервера через усі порти базового мережевих пристроїв, щоб

додати/змінити відповідну VLAN). Це вимагає співпраці між різними командами, що додатково впливає на гнучкість бізнесу.

- Обмежений масштаб: поле VLAN-ID має 12 біт, що дає максимум 4094 VLAN для створити сегментів шару 2. Цього масштабу, якого можна досягти за допомогою VLAN, недостатньо для задоволення зростаючих вимог у середовищі з тисячами клієнтів.
- Обмежена стабільність: використання каналів VLAN вимагає, щоб усі пристрої були доступні. На масштабованість MAC впливає на обмеження кількісті доступних MAC-адрес які можна вивчати для кожного пристрою. Непередбачувані обсяги трафіку BUM при масштабуванні MAC перевищений ліміт, що ще більше впливає на стійкість.
- Неефективне використання ресурсів: механізми уникнення циклу, такі як STP, блокує надлишкові зв’язки, надаючи лише один активний шлях, таким чином не використовуються всі доступні ресурси пропускної здатності. Це також впливає на трафік дистрибутивів, оскільки кілька надлишкових шляхів не можуть бути використані для балансування навантаження.
- Підтримка розширення рівня 2 і DCI (з’єднання центру обробки даних): необхідно забезпечити середовище розподіленого робочого навантаження, що забезпечує безперервність бізнесу, висока доступність, резервування та аварійне відновлення, необхідні кілька центрів обробки даних для географічної різноманітності. Центри обробки даних часто з’єднані між собою через WAN для забезпечення зв’язку між елементами програми, розподіленими між кількома центрами обробки даних. Розширення рівня 2 відноситься до здатності розтягуватися Рівень 2 через WAN, щоб забезпечити з’єднання рівня 2 між ними програми. Часто потрібна підтримка підключення рівня 2 через DCI вимагає мобільності робочого навантаження, як міграція віртуальної машини (потрібне розширення рівня 2 оскільки і вихідний,

і цільовий центри обробки даних повинні підтримувати ту саму IP-підмережу і розширити VLAN для живої міграції віртуальної машини). Інші поширені випадки використання включають геокластеризація для досягнення аварійного відновлення (центри обробки даних, що містять членів кластера потрібна зв’язності тієї самої IP-підмережі та розширення VLAN для кластерних серверів), висока кластерна доступності (розширення рівня 2 для полегшення обміну контрольного трафіку та бази даних реплікації) і підтримка безперервності бізнесу під час зростання серверної ферми. В сценарії розгортання, де острови рівня 2 розділені ядром рівня 3,можуть забезпечити зв’язність рівня 2 між центрами обробки даних у глобальній мережі, тоді як застарілі архітектури, які використовують лише VLAN, не підтримують такої можливості.

-

1.2 Зв’язок між overlay та underlay

Overlay мережа віртуалізує базову фізичну інфраструктуру шляхом створення логічних мереж поверх фізичної базової мережі. Це відокремлює пропоновану послугу від базового транспорту та скасовує необхідність підтримувати стан транзитних пристроїв, що покращує масштабованість. Підтримка стану тепер обмежена лише відповідальними кінцевими точками накладання інкапсуляція/декапсуляція віртуального тунелю (наприклад, гіпервізор сервера або TOR). Транзитні пристрої можуть мати менші таблиці пересилання, оскільки вони пересилають трафік за допомогою зовнішньої інкапсуляції або транспортного заголовка, який містить лише базовіIP адреси. Це відокремлення пропонує додаткову перевагу для швидшого надання віртуальної мережі без необхідності конфігурації транзитного обладнання.

Базова інфраструктура є прозорою для служб додатків, які використовують віртуальні тунелі, надані overlay мережею, для зв’язку

Underlaymережа надає можливість для транспортування інкапсульованих даних клієнта між обчислювальними ресурсами. Іншими словами, underlay відповідає за IPдоступність кінцевої точки тунелю, тоді як завдання з керування віртуальним тунелем обробляється overlay пристроями. Якщо underlayструктурована для масштабування та стійкості, вона може забезпечити оптимальні шляхи трафіку, які overlaymережа може використати для кращого керування та розподілу трафіку.

Загальна структура пакета, зображена на рисунку 1.2., є лише логічним представленням заголовків пакетів, відображених у overlayта underlaymережі.

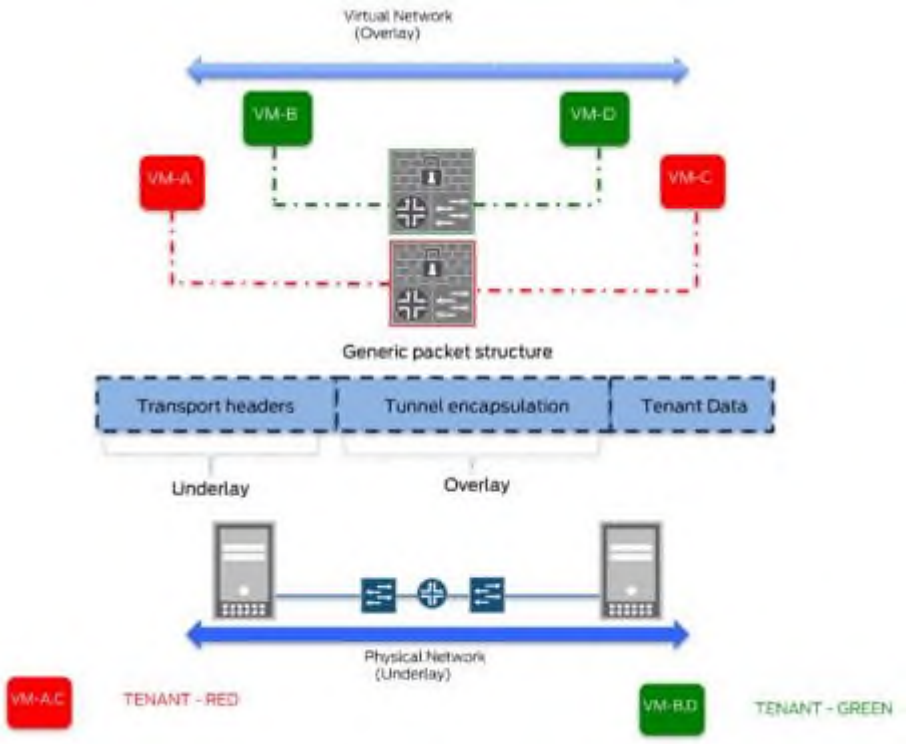


Рис. 1.2 Структура пакету

1.3 Типи underlay мереж

Структура центру обробки даних на основі архітектури Clos визначається як матриця тісно сплетених взаємозв’язків між мережевими пристроями, що утворює фабрику. Це забезпечує згладжену архітектуру з рівновіддаленими кінцевими точками та неблокуючим ядром для підтримки

низької затримки та високої пропускна здатність для трафіку East-Westi North-South. Такі фабрики є популярними серед центрів обробки даних.

Їх можна загалом класифікувати як:

- Мережа Ethernet: мережі Ethernet дозволяють керувати усією фабрикою як єдиним пристроєм. Прикладами мережевих рішень відJuniperEthernet є QFabric, VirtualChassisFabric (VCF) і JunosFusion. Переваги, які вони пропонують перед традиційними багаторівневими топологіямимережі «агрегація-ядро» включають: відсутність потреби в STP (використовується L3 зв'язність в транспортній мережі), підтримка декількох шляхів та балансування навантаження, єдине точка керування, а також розподілене пересилання без протоколів рівня керування. Недоліки включають обмеження масштабування, оскільки система не може масштабуватися та не є відкритим стандартом.
- Фабрика MPLS: MPLS в якостіunderlay для з'єднання всіх пристроїв у Closархітектурі. Концепції фізичної архітектури, які застосовуються до IP фабрики, також застосовуються до фабрики MPLS. ФабрикаMPLS може використовувати переваги FRR та інженерії трафіку, серед багатого набору функцій MPLS, які добре відомі та широко розгорнуті. Однак underlayy MPLS потребує, щоб всі мережеві пристрої підтримували MPLS. Протоколи сигналізації MPLS, як LDP, RSVP або BGP-LU можна використовувати для забезпечення побудови underlayмережі MPLS. LDP/RSVP пропонує свої добре відомі переваги. Використання MPLS у розгортанні центру обробки даних потребує використання протоколів LDP/RSVP LSP між TOR (припускаючи, що це кінцеві точкиoverlay мережі), що може призвести до обмеження масштабу площини керування через підтримку стану кожного LSP на транспортних пристроях. Інші супутні недоліки включають збільшення накладні витрати на керування кількома протоколами та відсутність детермінованих міток. MSDCs (MassiveScalableDataCenters) віддають перевагу BGP (eBGP) як єдиному протоколу маршрутизації через

масштабованість та стабільність, які він пропонує. Крім того, eBGPтакож пропонує переваги зменшення доменів відмов завдяки обмеженню широкомовнихL2 доменів, кращий багатошляховий доступ і механізми виявлення петель. BGP-LU (labeledunicast) також може служити як транспорт за допомогою відображення адреси інфраструктури (досяжність сервера) до міток MPLS (всередині AS або між кількома AS).Як альтернатива, сегментна маршрутизація або SPRING (SourcePacketRoutingInNetworking) можна використовувати для розповсюдження міток. Обидва IGP (ISIS – sub TLV та OSPF – непрозорі LSA) і BGP (розширення BGP-LU SPRING) підтримують SPRING. SPRING використовує вихіднуадресу пакету, щоб передати трафік правильним шляхом. Це дозволяє вузлу обрати оптимальний шлях проходження пакету, на основі вимог програми найкоротшого шляху. Стек міток еквівалентний ERO і використовується входом вузол, щоб явно вказати шлях, який потрібно пройти. SPRING пропонує переваги будучи детермінованим обчисленням міток із підтримкою ECMP, що забезпечує легкість усунення несправностей та інтеграцію із зовнішнім контролером, масштабованість в керуванні і менші операційні витрати.

- IP-фабрика: IP служить основою для з’єднання всіх пристроїв у структурі Clos, результатом є відкритий стандарт, плоска переадресація, горизонтально масштабована та неблокуюча архітектура spine-leaf (українською *листок-гілка*). Spine-leaf — це ролі, призначені мережевим пристроям фабрики, які з'єднані в ієрархічній структурі (3-ступінчастий Clos) в якій тунельоване віртуальнеoverlay з’єднання встановлене по верх звичайної IP мережі. Leaf пристрої забезпечують підключення для доступу тоді як пристрої Spine діють як точки агрегації. Поширені багатоступінчасті моделі IP- фабрики включають 3-рівневийClos (1 і 3 (leaf), 2 (spine)) і 5-рівневий Clos (1 і 5 (leaf), 2 і 4 (spine) і 3 рівень (superspineякі забезпечують зв’язок між

spinekомутаторами). Дивіться рисунок 1.3.Відкритий стандарт дозволяє підтримку технології між різними виробниками телекомунікаційного обладнання. Фабрику IP Clos можна масштабувати горизонтально, оскільки можна додати більше leafабо spineпристроїв для збільшення пропускної здатності без будь-якої серйозної реконструкції мережі. Кожен leafкомутатор підключено до кожного spineкомутатор за допомогою маршрутизованих p2p. Це дозволяє обмежити домен відмови, оскільки збій будь-якого вузла не призводить до виходу з ладу всієї структури. Багатошляховість (ECMP) тепер доступна між будь-якими двома кінцевими пристроями завдяки численним підключеннядо spineпристроїв, що забезпечує резервування, відмовостійкість і ефективне балансування трафіку на всіх доступних шляхах. L3 фабрики, за своєю суттю вільні від петель. Для забезпечення потреб в управління існують так звані контролери або інші інструменти автоматизації. Вони допомагають автоматизувати розгорнення та експлуатацію фабрики.

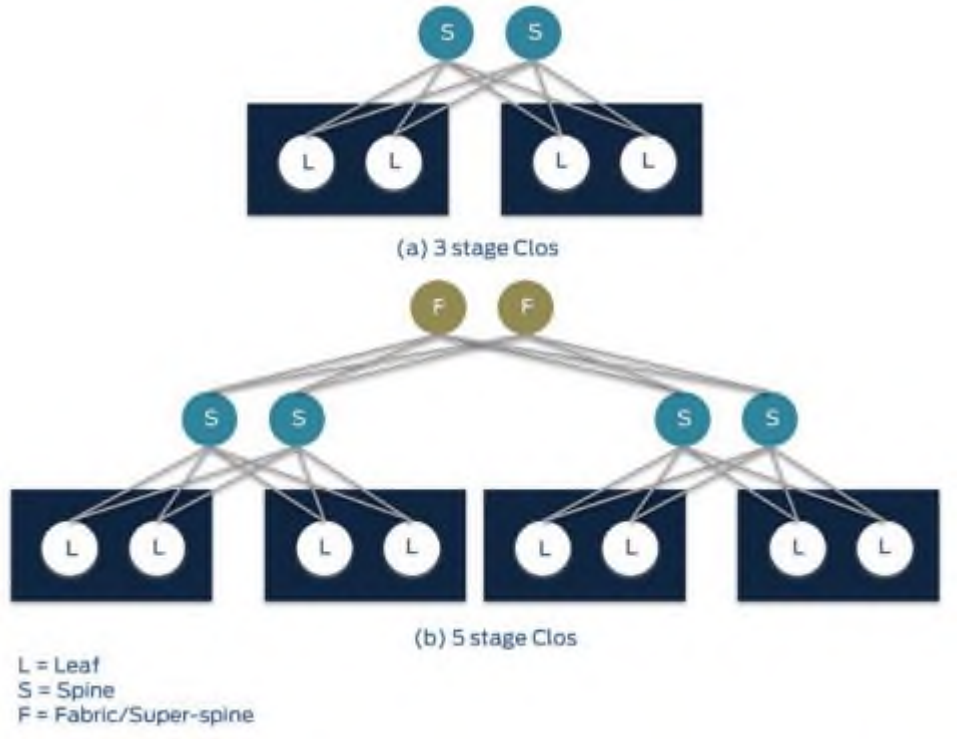


Рис. 1.3 5-рівневаClosархітектураIP-фабрики

- Популярні варіанти протоколів маршрутизації для забезпечення рівня керування IP-фабрики:
    - IGP(OSP/ISIS): протоколи на основі стану каналу, такі як OSPF/ISIS, розміри LSDB і так званий «*flooding»*повідомлень про стан каналів можуть стати вузьким місцем.Однак, незважаючи на те, що зони маршрутизації можуть допомогти зменшити масштаб де відбувається обмін інформації про стан каналів, суміжний рівень 2/ вимога до ієрархічного дизайну магістралі може ускладнити дизайн. Незважаючи на те, що ці протоколи легко налаштувати, є операційні витрати на керування кількомапротоколами для підключення до глобальної мережі, оскільки BGP є кращим варіантом в ядрі. Крім того, можливості управління трафіком обмежені (звичайні випадки використання як переміщення трафіку від spine вузла під час вікна обслуговування стає складним завданням для IP-фабрики на основі IGP).
    - iBGP: одна з вимог iBGP це наявність зв’язності кожен-з-кожним і має обмежену масштабованість через підвищений стан і накладні витрати на конфігурацію/операцію. Відбивачі маршрутів (англійською route-reflector) BGP може допомогти у вирішенні задання повної зв’язності , але на базуючись на алгоритмі вибору шляху BGP, рефлектори маршруту будуть анонсувати лише єдиний найкращий шлях дляйого клієнтів. У цій ситуації функція додавання додаткового шляху може забезпечити ECMP з iBGP. Це накладає додаткові вимоги до фабрики. Використання route-reflector-ів може забезпечують кращу продуктивність та непогану масштабованість, але це додає дизайну зайву складність.
    - eBGP: eBGP має кращий механізм уникнення петель використовуючи атрибут AS_PATH і можливості передачі по

кількох шляхах. Кожен вузол leaf/spine може бути поміщений в окремій AS, що призводить до розподіленості на домени маршрутизації. Використання 4-байтового AS може запобігти вичерпанню номерів AS, що використовується пристроями фабрики. Протокол був створений з можливістю високого масштабування, щоб підтримувати велику кількість пірингів та префіксів разом із простотою скороченої взаємодії протоколу, якщо BGP використовується в межах центру обробки данихабо через WAN. Кращі можливості traffic-engineeringспрощують управління трафікомза допомогою маніпулювання атрибутами BGP (наприклад, через використання способу as-path-prepend або використаннясommunity). Є певні труднощі у конфігуруванні політик маршрутизації та інших налаштувань, але це вирішується за допомогою контролеру або інших інструментів автоматизації, мова про які буде в наступному підрозділі.

1.4 Типи overlay мереж

Накладені мережі створюють канали зв’язку між додатками/службами орендарів шляхом створення логічних/віртуальних тунелів. Вони можуть підтримувати L2 (L2 overlay – розширення мережі L2, щоб віртуальні машини могли спільно використовувати ту саму IP-підмережу) і L3 (L3overlay – спільний доступ до мереж L3) транспортувати між службами, приховуючи їх від основної мережевої інфраструктури. Тунельна інкапсуляція складається з IDтунеля, який є функцією overlay мережі і діє як демультиплексор, який використовується для розрізнення різних потоків трафіку, що передаються тунелем, кожен потік представляє окреме з’єднання.

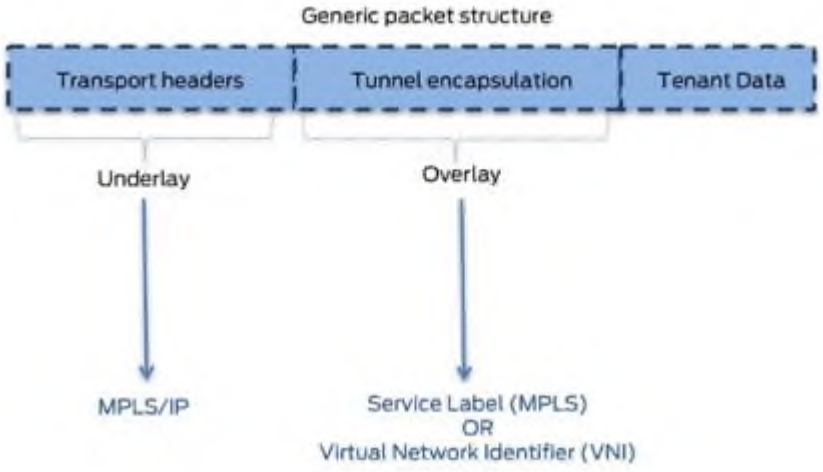


Рис. 1.4 Поля загальної структури пакета для overlay/underlay

Кожен канал зв'язку складається з площини керування для обміну інформацією про топологію клієнта (MAC-адреси/IP-маршрути) та інкапсуляції площини даних для інкапсуляції і пересилання трафіку між кінцевими точками накладеного тунелю через віртуальний тунель.Приклади інкапсуляції площини даних:MPLSoMPLS, MPLSoGRE, MPLSoUDP та VXLAN. Детальніше про останній спосіб: може використовуватися для транспортування пакетів даних через базові IP-мережі, які забезпечує підключення між кінцевими точками VXLAN Tunnel (VTEP). VXLAN інкапсулює MAC в UDP для розширення сегментів рівня 2, таким чином забезпечуючи рівень 2 накладання через мережу рівня 3. Для кращої масштабованості та через складність механізму багатоадресної передачі, що використовуються для вивчення MAC адрес зазвичай розгортаються OVSDB і EVPN для площини управління. ОскількиVXLAN немає внутрішньої підтримки безпеки, IPsec може використовується для забезпечення конфіденційності даних, коли основний IP-транспорт є ненадійним.

Приклади площини керування включають OVSDB, EVPN:

- OVSDB: OVSDB — це протокол площини керування, який визначає трафік та те, як трафік обміняли, але не що з цим робити. В основному розроблений для централізованих функцій керування на основі контролера для підтримки програмованості, він може налаштовувати

апаратні пристрої та імпорт/експорт динамічно отриманих адрес.З іншого боку, EVPN базується на контролі протоколу маршрутизації MP-BGP, і може запропонувати площину розподіленого керування або працювати з централізованим контролером. BGP має вбудовану підтримку масштабованості та атрибути, що становлять фреймворк на основі політик. Це дає можливість реалізувати безліч різноманітних функцій.

- EVPN: вивчення MAC адрес між PE досягається за допомогою площини керування. Він пропонує гнучкий вибір для інкапсуляції площини даних і площини керування MP-BGP (різні типи маршрутів EVPN) підтримує обмін MAC/IP-адресами. Підтримка all-activemultihoming дозволяє краще використовувати кілька каналів CE-PE для розподілу трафіку та підвищує доступність. Автоматизовані вибори DF, виявлення багатодомних PE та автоматичне виявлення сусідів у тій самій VPN дозволяють легко та спрощено ініціалізувати. Швидка конвергенція, мобільність MAC адрес, підтримка анонсів /32 хост-маршрутів, що дозволяють оптимізувати трафік і мобільність віртуальних машин (VMTO), ARP suppression і підтримка розподіленого шлюзу anycast та багато інших переваг EVPN.
-

1.5 Розуміння потреби у VXLAN

VXLAN (VirtualeXtensibleLocalAreaNetwork) — це технологія накладання рівня 2 на інфраструктуру рівня 3. Він надає засоби для розтягування мережі рівня 2, забезпечуючи тунельну інкапсуляцію за допомогою MAC-адрес в UDP (MAC в UDP) через IP underlay. Він використовується для передачі трафіку MAC від окремих віртуальних машин в інкапсульованому форматі через логічний «тунель». Нижче наведено деякі вимоги, пов’язані з розгортання центрів обробки даних, що розглядаються VXLAN:

- Вища масштабованість для кількох орендарів: ключова характеристика центру обробки даних рівня 2 мереж — це використання ними віртуальних локальних мереж (VLAN) для забезпечення ізоляції трансляції. А 12-бітний ідентифікатор VLAN використовується для ідентифікації сегмента рівня 2 і використовується в Ethernet кадрах даних для поділу великої мережі рівня 2 на кілька широкомовних доменів. У випадку, коли віртуальні машини в центрі обробки даних згруповано відповідно до їх VLAN, тисячі VLAN можуть знадобитися для розподілу трафіку відповідно до конкретна групи, до якої може належати VM. Із зростаючим впровадженням віртуалізації вимоги до багатокористувацького середовища прискорюють потребу в можливості створити більшу кількість віртуальних мереж. Як наслідок, ця верхня межа в 4094 є недостатньою в таких ситуаціях. VXLAN усуває це обмеження, використовуючи 24-бітний ідентифікатор мережі VXLAN (VNI) для ідентифікації сегмента рівня 2, що забезпечує збільшений простір для віртуальних мереж, до 16 млн ідентифікаторів. Це зменшує вузьке місце пов’язане з VLAN.
- Оптимальне використання ресурсів: STP забезпечує уникнення петель шляхом блокування надлишкових шляхів. Для STPбагатошляхова модель недоступна, щотакож створює труднощі для мережевих операторів, які застосовують додаткові порти, але не можуть використовувати їх. VXLAN може використовувати переваги underlay IP-фабрики, що усуває необхідність відключати портидля уникнення петель і допомагає досягти багатошляхового доступу завдяки ECMP. Крім того, ентропія порту відправника в UDPзабезпечує ефективне балансування навантаження трафіку VXLAN шляхомйого рівномірного розподілу.
- Мобільність робочого навантаження та гнучке розгортання: хмарні обчислення включають роботу за вимогою еластичного надання ресурсів для багатокористувацького середовища фізичної

інфраструктури. POD зазвичай складається з однієї або кількох стійок серверів із пов’язаною мережею та підключенням до сховища. Розширення крос-подів є поширеним випадком використання, коли орендарі можуть починати з POD і через розширення вимагати серверів/VM на інших модулях, особливо у випадку, коли орендарі на інших модулях не повністю використовують усі свої ресурси. VXLAN забезпечує розширені домени рівня 2, що з’єднують окремі сервери/VM для підтримки безперебійного переміщення та гнучкого розміщення віртуалізованих ресурсів у мережі центру обробки даних.

- Швидка ініціалізація: для підтримки гнучких потреб бізнесу, ініціалізація програм має бути швидкою і легкою. VXLAN як оверлей дозволяє досягти цього.

1.6 Основи VXLAN

Нижче наведено короткий перелік деяких основних компонентів VXLAN:

- Сегмент VXLAN: кожна накладена мережа VXLAN називається сегментом VXLAN. Хости (наприклад, віртуальні машини), які є частиною одного сегмента VXLAN (одного широкомовного домену), можуть спілкуватися один з одним. Для зв’язку між VNI потрібен шлюз рівня 3.
- Ідентифікатор мережі VXLAN (VNI): кожен сегмент VXLAN ідентифікується 24-бітним кодом, цей ідентифікатор називається VNI. Це підтримує високомасштабну сегментацію мережі, дозволяючи до 16 млн сегментів VXLAN для спільного існування в одному адміністративному домені. А VNI забезпечує ізоляцію трафіку для кожного сегмента VXLAN. Хост ідентифікується за допомогою унікального поєднання його MAC-адреси та VNI. Це дозволяє створювати однакові IP мережі для існування різних

сегментах. У результаті різні VNI перетворюються на різні домени і трафік не перетинає між сегментами мережі.

- Кінцева точка тунелю VXLAN (VTEP): кадр рівня 2, створений кінцевим хостом, інкапсульований у заголовку VXLAN, який містить VNI, який використовується для ідентифікації VXLAN сегмент, до якого належить цей кінцевий хост. Тунелі також використовуються для додавання зовнішнього заголовка інкапсуляції, що складається з заголовків UDP, IP та Ethernet для транспортування трафіку кінцевого хоста через базову IP мережу. Ці кінцеві точки тунелю або NetworkVirtualizationEdges (NVE), які відповідають за відображення кінцевих хостів у відповідні сегменти VXLAN (відображення локальної VLAN у VXLAN VNI) і VXLAN пов'язана тунельна інкапсуляція та декапсуляція називаються VTEP. VTEP можуть існувати на будь-якому мережевому пристрої (середовище багатьох постачальників) або обчислювальний елемент (наприклад, сервер гіпервізор). Кожен VTEP ідентифікується унікальною IP-адресою (адреса lo0). Фізична основна мережа відповідає за забезпечення IP-досяжності між VTEP для забезпечення наскрізної доставки трафіку. VTEP виконують дві важливі функції а саме: 1) Віддалене виявлення VTEP для створення відображення MAC-адрес кінцевого хоста і відповідні VTEP для сегментів VXLAN, за які він відповідає; і, 2) Інкапсуляція VXLAN трафіку кінцевого хоста, який прозоро транспортується через базової мережі та подальшої декапсуляції отриманих пакетів для доставки кінцевим хостам. VTEP можуть перемикати пакети в тому самому VNID або тій самій трансляції домен (шлюзи L2) або може маршрутизувати пакети через різні VNID або різні підмережі (шлюзи L3).
- Інкапсуляція VXLAN: інкапсуляція VXLAN додає додаткові 50-54B накладні витрати в результаті додавання заголовків UDP, IP та

Ethernet до кадру, створеного кінцевим хостом. Тому важливо, щоб IP undelay лежала в основі мережі підтримувала Jumbo (Великі) кадри для врахування цих накладних витрат, що означає, що знадобляться значення MTU відображати однаково по всьому шляху, пройденому інкапсульованим пакетом. Доданий заголовок UDP заповнює значення вихідного порту, обчислюючи хеш оригіналу вміст внутрішнього кадру (заголовки L2/L3/L4). Це дозволяє забезпечити ентропію потоку та ефективне балансування навантаження трафіку VXLAN через шляхи ECMP, запропоновані IPмережею. Контрольна сума UDP встановлена на 0x0000. Якщо для контрольної суми не встановлено значення 0x0000 джерелом VTEP, тоді приймаючий VTEP повинен перевірити контрольну суму. Якщо контрольна сума не вірна, то такий сегмент відкидається. Зовнішній IP-заголовок складається з IP-адрес VTEP відправника та отримувача, і ці значення використовуються для пакету, який буде маршрутизованаunderlayмережею. Пристрої, що лежать в основі underlayмережі не перевіряють VXLANзаголовок, а лише здійснюю маршрутизацію до VTEP на якому термінується тунель.

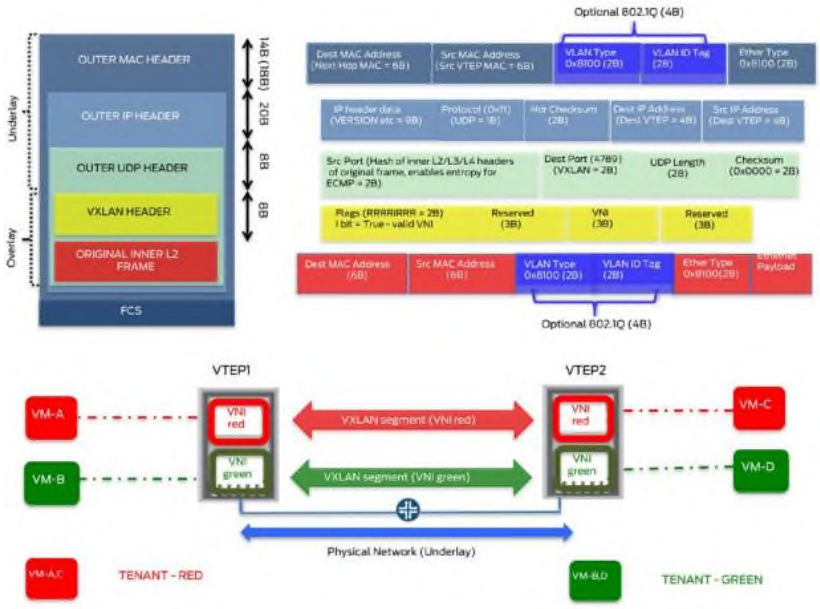


Рис. 1.5 Формат кадру VXLAN і термінологія

1.7 Варіанти VXLAN

Протокол площини керування визначає, як пристрої формують стосунки, обмінюються інформацією про топологію та приймають рішення вибору найкращого шляху. Це, у свою чергу, впливає на поведінку площини даних, яка визначає, як пакети даних транспортуються через мережевий шлях. Поточний стандарт VXLAN не визначає площину управління. У результаті обмін інформацією про топологію або вивчення адреси виконується в площині даних, подібно до поширення й вивчення рівня 2. VTEP дізнається MAC-адресу локального кінцевого хоста (локальне вивчення), він створює запис у своїй таблиці пересилання. За відсутності протоколу площини керування він не розповсюджує цю інформацію до будь-якого із віддалених VTEP. Кожен VXLAN вузол повинен виконувати власне незалежне вивчення адреси. Коли VTEP отримує пакет VXLAN (віддалене вивчення), він створює асоціацію, яка відстежує віддалений MAC, VNI для якого він був отриманий, та віддалений VTEP, з якого він був надісланий. Ці записи необхіднідля того, щоб він знав, як у майбутньому пересилати трафік, призначений на цей віддалений MAC. Для багатоадресного трафіку (Broadcast, UnknownUnicast і Multicast), він вдається до механізму фладінгу (англійською *flooding*, наповнення). Варіанти VXLAN відрізняються своїми можливостями та тим, як вони дозволяють вивчати MAC і обробляти трафік BUM.

- Multicast /VXLAN: Віддалене вивчення MAC виконується в площині даних. Кожен VXLAN сегмент або VNID відображає групу багатоадресної адреси. VTEP, які мають кінцеві хости в а даний VNID приєднається до тієї самої багатоадресної групи (IGMP). Групи багатоадресної передачі таким чином використовується для обмеження обсягу фладінгу трафіку BUM лише необхідним VTEP. Для невідомоїодноадресної або широкомовної, VTEP розішле пакет

VXLAN шляхом надсилання лише для групи багатоадресної адреси, пов’язаної з даним сегментом VNID або VXLAN. При базовій реплікації пакети реплікуються на найдальшій точці у фізичній топології, що є ефективним для інтенсивного багатоадресного трафіку. Однак для underlay виконання багатоадресної передачі може бути значною в операційних накладні витратах, особливо в середовищах, які сьогодні не використовують групову розсилку і мали би це зробити лише за підтримки VXLAN. Хоча багатоадресна передача надає деякі ефективності використання ресурсів, це створює проблеми масштабованості. Незважаючи що VXLAN може підтримувати 16 млн логічних мереж рівня 2, неможливо підтримувати відображення 1:1 між такою кількістю VNID і групами багатоадресної передачі. Якщо дана багатоадресна група повинна підтримувати більше ніж один VNID, це може призвести до непотрібного флуду, коли незацікавлені у отриманні розсилки VTEP все одно повинні будуть виконувати певну обробку, щоб відкинути трафік від небажаних VNID.

- Unicast (Static) VXLAN: як і Multicast/VXLAN, вивчення MAC адрес здійснюється в площині даних. Завдяки використанню вхідної реплікації вихідний VTEP буде копіювати пакет BUM локально та надсилати одноадресну копію кожному з інших VTEP які беруть участь у тому самому сегменті VXLAN. Для певного VNI можна налаштувати кілька одноадресних VTEP. Статична конфігурація, водночас простіша в управлінні та забезпечує детальний контроль над областю розсилки, блокує автоматичне виявлення динамічних змін топології. Крім того, додаткова обробка на головному вузлі це може бути не ідеальним вибором для інтенсивного багатоадресного трафіку.
- OVSDB/VXLAN: OVSDB забезпечує функціональність рівня керування для VXLAN, тобто дистанційне вивчення MAC тепер

виконується в площині керування. OVSDB зазвичай використовується в поєднанні з накладним контролером. Розповсюдження трафіку BUM обробляється контролером, який використовує OVSDB, щоб увімкнути вивчення MAC-адреси та обробляє VTEP забезпечення.

- EVPN/VXLAN: як і OVSDB, EVPN також є технологією рівня керування для VXLAN. Він використовує MP-BGP, щоб увімкнути дистанційне вивчення MAC у площині керування та може використовувати з контролером або без нього.
-

1.8 Апаратна частина VTEP

Більшість середовищ центрів обробки даних складається з комбінації віртуалізованих і невіртуалізованих обчислювальних ресурсів. Невіртуалізовані сервери (англійською, *Bare-MetalServers(BMS))* не підтримують серверну або мережеву віртуалізацію. Вони не підтримують і не використовують VXLAN VLAN для сегментації мережі. Зазвичай вони використовуються високопродуктивними програмами, які потребують виділених ресурсів. Часто виникає потреба забезпечити зв’язок між віртуальними та фізичними ресурсами. Деякі поширені приклади включають взаємодію багаторівневої програми, міграція з фізичного на віртуальний (P-V), віртуальні машини повинні спілкуватися з брандмауером або пристроями балансування навантаження, розгорнутими на BMS. Окрім підключення віртуального робочих навантажень, шлюзи VTEP також можуть обробляти VLAN до VXLAN з’єднання фізичних і віртуальних робочих навантажень.

Виходячи з того, де знаходиться кінцева точка віртуалізації мережі (NVE), VTEP можна класифікувати, як зазначено в таблиці 1.1

Таблиця 1.1

Класифікація VTEP

	Програмні шлюзи VTEP	Апаратні шлюзи VTEP
Опис	Коли функцію VTEP увімкнено в програмному забезпеченні (гіпервізор із підтримкою VXLAN) на сервері. Це часто використовується в поєднанні з контролером SDN (наприклад, JuniperContrail або VMware NSX).	Коли функцію VTEP увімкнено на апаратному мережеврмупристрії. Таке рішенняможна використовувати зSDN контролером або без нього та функціоналом аналогічному програмним VTEP. Апаратні VTEP зазвичай використовують протокол рівня керування, наприклад OVSDB або EVPN для взаємодії з контролером.
Масштабованість, продуктивність	Хоча програмні шлюзи можуть бути економічно ефективнішими. Коли вимоги до трафіку не дуже високі, це робить їх хорошим вибором для невеликих розгортань.	Апаратні шлюзи VTEP, пропонують високу щільність, продуктивність і кращу масштабованість. Для значного розгортання, якщо існує велика кількість BMS, що спілкуватися з віртуалізованими робочими навантаженнями, або якщо потрібно обробляти великий трафік. У рішення можна використовувати зв’язки

Продовження таблиці 1.1

		шлюзів VTEP: фізично-фізичний (BMS-BMS), віртуально-віртуальний (VM-VM) або віртуально-фізичний (VM-BMS).
Відмовостійкість	З програмними шлюзами VTEP кінцеві хости/VM є розташовані разом із NVE/VTEP у тому самому фізичному сервері і зв'язки між ними віртуальні. Зазвичай мультидомінгне є можливим, і навіть якщо він є суто локальний для фізичного сервера і невидимий для будь-який іншого VTEP, що знаходиться на іншому сервері. В цьому рішеннямеханізми резервування обмежені.	З використанням EVPN як площини керування, апаратні шлюзи VTEP можуть підтримувати різні режими резервування, такі як All-Active або Single-Activemultihoming.

1.9 EVPN

Мета розробки NetworkVirtualizationOverlay (NVO) — це мільйони віртуальних мереж для загальної фізичної інфраструктури, а тому властивості масштабування площини керування NVO є надзвичайно важливими. EVPN, описані тут розширення, розроблено з урахуванням потреби цього рівня масштабованості:

- Вивчення рівня керування та механізм ARP suppression: EVPN розрізняє, як MAC-адреси вивчаються залежно від того, чи пристрій CE безпосередньо підключено до PE-пристрою, чи пристрій CE знаходиться за іншим PE-пристроєм. Коли пристрій CE

безпосередньо підключено до PE, здійснюється вивчення здійснюється відповідно IEEE MAC , у цьому разі можна використовувати методи вивчення через площину даних. Це називається локальним вивчення. Коли CE пристрій підключено до віддаленого PE-пристрою, а локальний PE-пристрій має вивчити MAC-адресу пристрою CE через віддалений пристрій PE - це називається дистанційне вивчення. На відміну від традиційного з'єднання, вивченняMAC між PE в EVPN відбувається в площині керування, а не площині даних. Вони не вивчаються через пересилка самих кадрів, а вивчення відбувається в площині керування за допомогою MP-BGP. Зі збільшенням широкомовних доменів відбувається збільшення трафіку ARP. Ця функція EVPN призводить до зменшення фладінгу для невідомого одноадресного трафіку. Більшість кінцевих хостів надсилають GratuitousARP (GARP) запити, коли вони з'являються в мережі, що дозволяє локальному вивченню EVPN PE далі розповсюджувати отримані MAC- та IP-адреси іншим PE за допомогою MP-BGP через площина управління для дистанційного вивчення. В полі NLRI (маршрутні анонси MAC/IP) використовуються для розповсюдження MAC/IP-адрес між PE, і як така доступність анонсована, пристрої знають куди надсилати трафік. EVPN також передбачає цю можливість щоб придушити ARP-флуд, активувавши функціональність Proxy-ARP. Коли кінцевий хост ініціює запит ARP для іншого віддаленого кінцевого хоста, його перехоплює EVPN PE, який виконує пошук проксі-ARP за запитуваною IP-адресою. Якщо відбувається збіг, то він надсилає відповідь ARP від імені віддаленого кінцевого хоста, без непотрібного заповнення запитів ARP, що споживають пропускну здатність мережі.

- Активний/Активний (A/A) Multi-homing і Aliasing: Коли клієнтський пристрій або мережа підключається до двох або більше

PE через декількаEthernetканалів (називається Ethernet сегмент), і всі багатоадресні PE можуть пересилати відомий одноадресний трафік до/від нього через VLAN Ethernet-сегменту, тоді таке багатонапрямлення називається All Активний. Підтримка EVPN для A/A multihoming забезпечує кращу доступність сайту та балансування навантаження, оскільки всі канали є активні та використовуються. Багатодомні PE налаштовані з однаковими параметрами LACP для даного ESI, що, утворює єдину систему з точки зору CE. Це дозволяє CE надсилати різні потоки до обох PE для той же CE-VID (VLAN-ID). Коли кілька PE EVPN підключено до одного сегментуEthernet (наприклад, підключений до того самого сервера через LAG), це можливо лише коли один PE дізнається MAC-адреси від кінцевого хоста. У результаті віддалені PE отримують оголошення MAC/IP маршрути для цих адрес від одного PE, навіть якщо підключено кількома PE до того самого сегменту Ethernet. Aliasing дозволяє віддаленим PE ефективно балансувати навантаження трафік між усіма багатодомними PE для певного сегмента Ethernet (той самий ESI). Ethernet Маршрути A-D використовуються для передачі до віддалених PE, з яких доступні MAC-адреси усіх PE з декількома адресами для певного сегмента Ethernet, навіть якщо доступність не анонсована всіма ними.

- Швидка конвергенція та масове відкликання: EVPN визначає новий механізм відновлення від збоїв зв’язку PE-CE, а також збоїв багатодомних вузлів PE, що швидко повідомляє віддаленим PE про необхідність оновлення їхніх таблиць пересилання. Коли EVPN PE втрачає з’єднання з певним сегментом Ethernet, він відкликає відповідний маршрут Ethernet A-D (A-D на ES). Це призводить до того, що всі віддалені вузли негайно очищаються від FIB, усі MAC-адреси, отримані від невдалого вузла на цьому ESI, замість того, щоб чекати, поки всі маршрути MAC-адрес будуть відкликані,

таким чином увімкнувши швидка реконвергенція. Як наслідок, реконвергенція мережі після збою не залежить від кількість MAC-адрес, отриманих PE. Це важливо в контексті програм віртуалізації, які сприяють збільшенню обсягу MAC адрес, які будуть оброблені мережею.

- Мобільність MAC: для підтримки мобільності робочого навантаження та міграції підтримується EVPN MAC Mobility (MAC переміщення). Це відноситься до можливості кінцевого хоста (визначається MAC-адресою), щоб перейти від одного сегмента Ethernet до іншого. У ході переміщення MAC-адреси може бути доступною через кілька Ethernet сегментів, оскільки може бути кілька анонсів маршрутів MAC/IP. На допомогу PEsдля відстеження та обробки подій мобільності, маршрути анонсів MAC/IP позначені тегами з атрибутом MAC mobilityextendedcommunity із порядковим номером. EVPN також підтримує можливість пов’язувати MAC-адреси з певними сегментамиEthernet (stickyMAC, тобто статичні MAC-адреси, які не підлягають MAC переміщенню). Переміщення для таких закріплених MAC-адрес заборонено, а трафік з будь-якого іншого сегмента Ethernet відкидається. Це допомагає обмежити несанкціонований трафік(підроблені адреси) у мережі для цієї MAC-адреси.
- Легкість ініціалізації: простота ініціалізації стає найважливішою з розширенням розгортання підприємств і центрів обробки даних. EVPN представляє єдину технологію VPN рішення для VPN рівня 3 і рівня 2 для досягнення спрощення розгортання. В на додаток до автоматичного виявлення PE, що належать до певного екземпляра VPN, EVPN також підтримує автоматичне виявлення PE, що належать до певної багатодомної групи також автоматизовані вибори призначеного DesignatedForwarder (DF) серед багатодомних PE для запобігання зацикленню трафіку BUM. Атрибути, необхідні

для налаштування сервісу, для Наприклад, RouteDistinguisher, RouteTarget, тощо можуть бути автоматично отримані без необхідності явного налаштування. Існуючі та добре зрозумілі концепції архітектури, такі як фільтрація маршрутів BGP і обмежений розподіл маршрутів для обмеження та розповсюдження маршруту тільки зацікавленим PE,відображення маршрутів BGP для створення повнозв'язнихтопологій, може бути застосовується для створення рішень EVPN.

- Розподілений anycastgateway: EVPN підтримує інтегровану маршрутизацію та перемикання, дозволяючи переадресацію як всередині підмережі (рівень 2), так і між підмережами (рівень 3). Розподілена функція будь-якого шлюзу забезпечує оптимальну переадресацію з кінця хости можуть використовувати свій локальний шлюз для надсилання трафіку за межі своєї підмережі IP. IRB інтерфейси, які діють як шлюзи рівня 3 для орендаря (VNI), можуть бути розподілені між собою на різних PE.
- Оптимізація трафіку: EVPN забезпечує оптимізовану доставку вхідного та вихідного трафіку. Синхронізація MAC-IP хоста відноситься до процесу PE, який має дізнався прив'язку MAC/IP (перегляд ARP), щоб передавати інший анонс маршруту MAC, який містить як MAC-адресу, так і IP-адресу для кінцевого хоста. Приймаючий PE встановлює MAC-адресу в EVI (MAC-VRF) та IP-адресу у відповідний IP-VRF. Служби з підтримкою L2 і L3 можуть бути розтягнуті між ними. Ця можливість додатково розширює підтримку для VMTO (VirtualMachineTrafficOptimization), що запобігає трафіку щодо міграції VM. Віртуальна машина, яка перемістилася, могла не скинути таблицю ARP оскільки він продовжує надсилати пакети між підмережами до свого попереднього шлюзу за замовчуванням, який може більше не бути оптимальним або навіть локальним шлюзом за замовчуванням після

переміщення.Як в результаті вихідний трафік від віртуальної машини до зовнішнього пункту призначення може призвести до перешкод через попередньо налаштований шлюз за замовчуванням. Цей ефект на виході можна усунути за допомогою функції розподіленого шлюзу будь-якої адреси (адреса віртуального шлюзу), який гарантує, що кожен маршрутизатор у межах VLAN має активний екземпляр зашлюза замовчуванням. Для певної VLAN/підмережі той самий шлюз за замовчуванням (та сама IP/MAC-адреса) реалізується одночасно на відповідних пристроях, тим самим забезпечуючи замовчування шлюз для кінцевих хостівякий завжди знаходиться в найближчій точці. Тромбування трафіку також може спостерігатися у вхідному напрямку, коли є зовнішній отримувач через WAN намагаючись досягти кінцевого хоста, який перейшов з DC-1 на DC-2 (та сама IP-підмережа). Початкова досяжність могла бути встановлена за допомогою шлюзів DC-1 і DC-2, що рекламують зведені маршрути. Кінцевий хост через WAN може віддати перевагу анонсу від шлюзу DC-1 для певної підмережі IP. З того часу як зникає видимість маршрутів хосту або видимості точного розташування хоста призначення, щоб дістатися дохоста, який перейшов з DC-1 на DC-2 трафік від віддаленого кінцевого хоста, може передаватися через DC-1.

- Підтримка різних площин даних: EVPN роз’єднує площину керування та даних. Це відокремлення дозволяє створити узгоджену базу даних пересилання трафіку керування через мережу, використовуючи вивчення площині керування замість того, щоб покладатися на фладінгчерез площину даних. MP-BGP забезпечує функціональність рівня керування, яка є однаковою для всіх інкапсуляцій площини даних. [4]

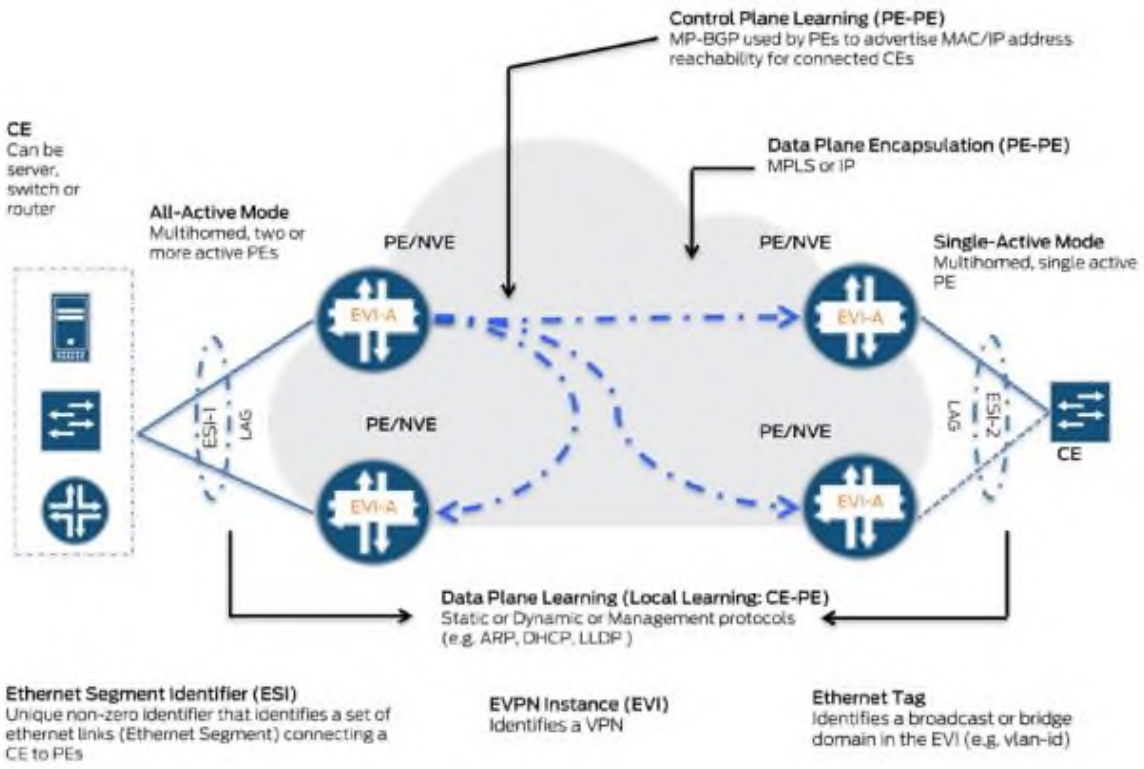


Рис. 1.6 Короткий виклад основних концепцій EVPN

2 ВИЗНАЧЕННЯ SDN

Програмно-визначена мережа (SDN) — це ідея, яка відносно нещодавно відновила інтерес дослідників щодо програмованих мереж і змістив увагу мережевої спільноти до цієї теми, пообіцявши зробити процес проектування та управління мережами більш інноваційнимта спрощеним порівняно з традиційним, але негнучкий поточний підходом.

Проектування та керування комп’ютерними мережами може стати дуже важким завданням високого рівня складності. Тісний зв'язок між мережами площини управління (де приймаються рішення щодо обробки трафіку) і площини даних (де відбувається фактичне пересилання трафіку) викликають різні проблеми, пов’язані з його управління та еволюцією. Операторам мереж необхідно вручну трансформувати політики високого рівня в команди конфігурації низького рівня, процес досить складний мережі можуть бути справді складними та схильними до помилок. Представляючи нові функції для мережі, як системи виявлення вторгнень табалансувальники навантаження, втручання в інфраструктуру мережі безпосередньо впливає на її логіку Розгортання нових протоколів може бути повільним процесом, що вимагає років стандартизації і тестування для забезпечення взаємодії між реалізаціями наданими різними постачальники.

Ідея програмованих мереж була запропонована як засіб для виправлення цієї ситуації, сприяючи інноваціям в управлінні мережею та розгортанні мережевих послуг за допомогою програмування базових мережевих об’єктів використовуючи API. Це призводить до створення гнучких мереж, здатних працювати відповідно до потреб користувача у прямій аналогії з мовами програмування, що використовується для перепрограмування комп’ютерів, щоб виконувати низку завдань без необхідність постійної модифікації основної апаратної платформи.

SDN є відносно новою парадигмою програмованої мережі, яка змінює спосіб проектування та управління мережами шляхом введення абстракції, яка відокремлює елемент керування від площини даних, як показано на рисунку 2.1. При такому підході програма керування програмним забезпеченням, яка називається контролером, має огляд мережівцілому та відповідає за прийняття рішень, а апаратне забезпечення (маршрутизатори, комутатори тощо) просто відповідає за пересилання пакетів до місця призначення за інструкціями контролера, як правило, це набір правил обробки пакетів.

Відокремлення логічно централізованого керування від базової площини даних швидко стало центром яскравого дослідницького інтересу мережевої спільноти оскільки це значно спрощує керування мережею та її еволюцію кількома способами. Нові протоколи та програми можна тестувати та розгортати в мережі без них вплив на непов'язаний мережевий трафік; додаткова інфраструктура може бути введена без зайвого клопоту; проміжні блоки можна легко інтегрувати в програмне керування, дозволяючи пропонувати нові потенційні рішення для проблем, які давно існують у центрі уваги, наприклад, керування надзвичайно складним ядром стільникових мереж.

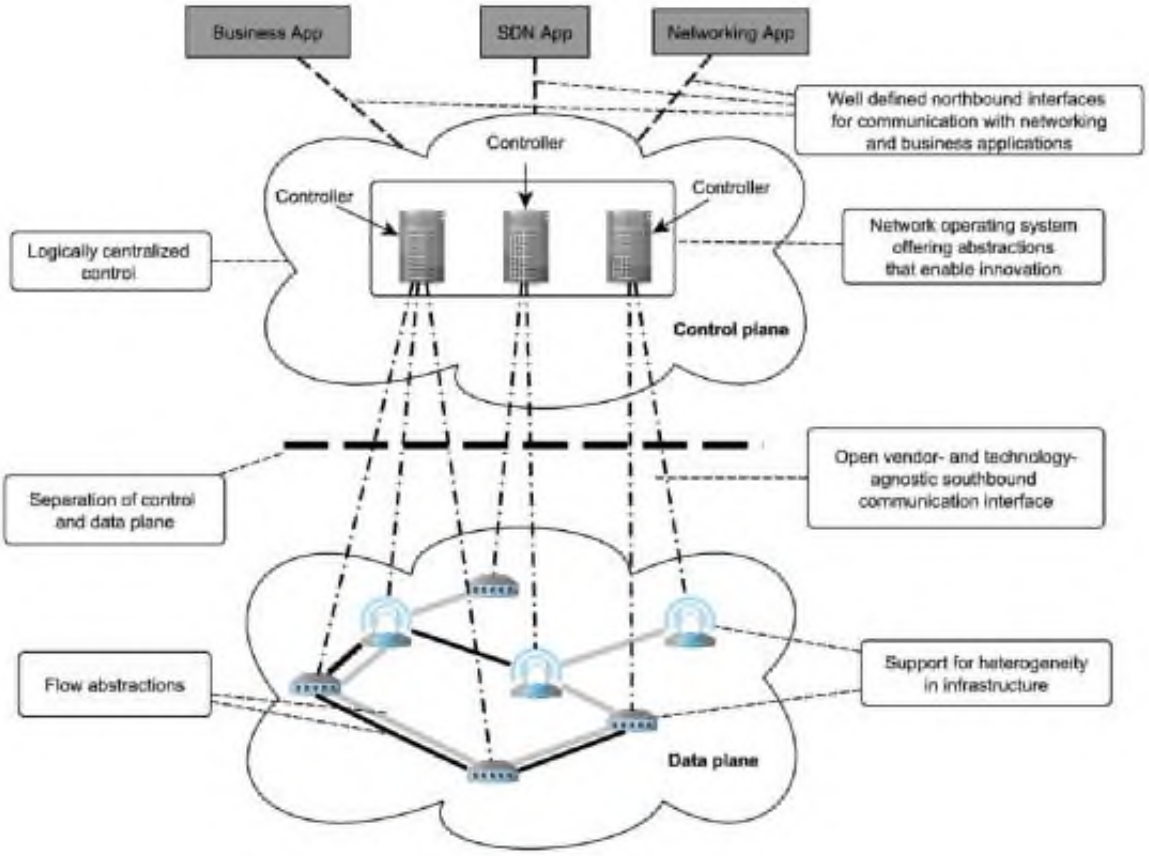


Рис.2.1 SDN в двох словах - ключові ідеї, що лежать в основі парадигми SDN

Цей розділ є загальним оглядом SDN для читачів, які щойно познайомилися до парадигми SDN, а також для тих, хто потребує огляду її минулого, теперішнього та майбутнього. Через обговорення та приклади викладені далі пропонується можливість зрозуміти, чому і як SDN змінює парадигми щодо проектування та управління мережами та розуміння переваг, які вони мають може запропонувати низку цікавих сторінок, наприклад операторам ЦОД.

2.1 Парадигма та додатки SDN

Як уже згадувалося, підхід SDN дозволяє керувати мережею послуги через абстракцію нижчого рівня. Замість того, щоб займатися низькорівневими деталями мережевих пристроїв щодо способу передачі пакетів і потоків керування, мережевим адміністраторам тепер потрібно лише використовувати абстракції, доступні в архітектурі SDN. Спосіб, яким

це досягається, полягає в роз’єднанні контрольної площини з площиною даних відповідно до багаторівневої архітектури, зображеної на рисунку 1.7.

На нижньому шарі ми можемо спостерігати площину даних, де знаходиться мережа інфраструктура (комутатори, маршрутизатори, бездротові точки доступу тощо). В контексті SDN ці пристрої були позбавлені всієї логіки керування (наприклад, алгоритмів маршрутизації як BGP), просто реалізуючи набір операцій пересилання для маніпулювання мережевих пакетів даних і потоків, забезпечуючи абстрактний відкритий інтерфейс для спілкування з верхніми шарами. У термінології SDN це пристрої зазвичай називаються мережними комутаторами.

Переходячи до наступного рівня, ми можемо спостерігати контрольну площину, де знаходиться сутність називають рівнем контролера. Ця сутність інкапсулює логіку мережі та відповідає за надання програмного інтерфейсу мережі, що повинен впроваджувати нову функціональність і виконувати різноманітні завдання управління. На відміну від попередніх підходів, такі як ForCES, площина керування SDN повністю вилучається з мережевих пристроїв і вважається логічно централізованою, тоді як фізично він може бути як централізованимтак і децентралізованим, що знаходиться на одному або кількох серверах, які контролюють інфраструктуру мережі вцілому.

Важливий аспект, який відрізняє SDN від попереднього програмованого мережевих спроб полягає в тому, що він ввів поняття мережевої операційної системи абстракції . Попередні зусилля, такі як активна мережа, пропонували деякі свого роду операційні система вузла (наприклад, NodeOS) для керування основним обладнанням. Мережева операційна система пропонує більш загальну абстракцію стану мережі комутаторів, що відкриває спрощений інтерфейс для керування мережею. Ця абстракція передбачає логічно централізовану модель керування, у якій програми переглядають мережу як єдину систему. Іншими словами, мережева операційна система діє як проміжний рівень, відповідальний за підтримку узгодженого перегляду стану мережі, який потім

використовується логікою керування для надання різноманітних мережевих послуг топологічного відкриття, маршрутизації, управління мобільністю та статистикою тощо.

У верхній частині стеку SDN лежить прикладний рівень, який включає всі програми, які використовують послуги, надані контролером, для виконання завдання, пов’язані з мережею, як-от балансування навантаження, віртуалізація мережі тощо. Одне з найбільш важливих особливостей SDN є відкритість, яку вона забезпечує для сторонніх розробників через абстракції, які він визначає для легкої розробки та розгортання нового програми в різних мережевих середовищах від центрів обробки даних і глобальних мережах для бездротових та стільникових мереж. Крім того, архітектура SDN усуває цю потребу для виділених проміжних пристроїв, таких як брандмауери та системи виявлення вторгнень (IDS) в мережевій топології, оскільки тепер можна реалізувати їх функціональні можливості форма програмного забезпечення, яке контролює та змінює стан мережі служби мережевої операційної системи. Очевидно, існування цього шару додає велика цінність для SDN, оскільки це створює широкий спектр можливостей для інновацій, робить SDN привабливим рішенням як для дослідників, так і для промисловості.

Зв’язок контролера з площиною даних і додатком рівня можна досягти за допомогою добре визначених інтерфейсів (API). Ми можемо виділити два основні API в архітектурі SDN: 1) південний API для зв’язку між контролером і мережевою інфраструктурою; і 2) північний API визначення інтерфейсу між мережевими програмами та контролером. Це є подібно до того, як досягається зв’язок між апаратним забезпеченням, операційною системою та простором користувача в більшості комп’ютерних систем.

Ознайомившись із загальним оглядом архітектури SDN, настав час для поглибленого обговорення кожного зі щойно представлених будівельних блоків.

2.2 Комутатори SDN

У звичайній парадигмі мереж розглядається мережева інфраструктура є найбільш невід'ємною частиною мережі. Кожен мережевий пристрій інкапсулює всі функціональні можливості, необхідні для роботи мережі. Наприклад, а маршрутизатор повинен забезпечити належнуTernaryContentAddressableMemory(TCAM) для швидкого пересилання пакетів, а також складне програмне забезпечення для виконання розподілених протоколів маршрутизації, таких як BGP. Так само бездротовій точці доступупотрібно належне обладнання для бездротового підключення так як і програмне забезпечення для пересилання пакетів, контролю доступу тощо. Однак динамічна зміна поведінки мережевих пристроїв не є тривіальним завданням через їх закритий характер.

Трирівнева архітектура SDN, представлена вище, змінює це шляхом роз’єднання площини управління від операцій пересилання, що спрощує управління мережею пристроїв. Як уже згадувалося, усі пристрої пересилання зберігають апаратне забезпечення, яке відповідає за зберігання таблиць пересилання (наприклад, інтегрованих для конкретної програми схеми - ASIC - з TCAM), але позбавлені своєї логіки. Контролер диктує комутаторам, як пакети повинні пересилатися шляхом встановлення нового правила пересилання через абстрактний інтерфейс. Кожен раз, коли пакет надходить до комутатора перевіряється його таблиця пересилання, і пакет пересилається відповідно.

Незважаючи на те, що в наведеному вище огляді SDN булапредставлена чиста трирівнева архітектура, залишається незрозумілим, які межі між контролером і даними повинні бути. Наприклад, ActiveQueueManagement(AQM), планування і конфігурація — це операції, які все ще вважаються частиною площини даних навіть у корпус комутаторів SDN. Однак немає жодної внутрішньої проблеми, яка б запобігла цьому функції від того, щоб стати частиною площини керування, запровадивши певний вид абстракції, що дозволяє контролювати поведінку низького рівня в

комутаційних пристроях. Такий підхід може виявитися корисним, оскільки він спростить розгортання на нові більш ефективні схеми роботи комутаторів низького рівня.

З іншого боку, при цьому всі контрольні операції переходять до логічно централізованого контролера, що має перевагу в легшому керуванні мережею, він також може підвищувати проблеми з масштабованістю, якщо фізична реалізація контролера також централізована. Таким чином, може бути корисним зберегти частину логіки в свічах, наприклад у випадку DevoFlow,який є модифікацією OpenFlow, моделі потоків пакетів поділяються на дві категорії: малі («мишачі») потоки обробляється безпосередньо комутаторами та великими («слоновими») потоками, які потребують втручання контролера. Аналогічно в контролері DIFANE комутатори використовуються для зберігання необхідних правил, а до контролеру віднесено просте завдання розподілу правил за комутаторами.

Інша проблема комутаторів SDN полягає в тому, що правила пересилання використовуються у випадку SDN є складнішими, ніж у звичайних мережах, для яких використовуються символи підстановки пересилання пакетів, враховуючи кілька полів пакета, наприклад джерело та адреси призначення, порти, програми тощо. У результаті комутаційне обладнання не може легко впорається з керуванням пакетами та потоками. Для пересилання для швидкої роботи потрібні ASIC з використанням TCAM. На жаль, таке спеціалізоване апаратне забезпечення є дорогим і енерговитратним, і, як наслідок, лише обмежена кількість записів пересилання для схем пересилання на основі потоку можуть підтримуватися кожним комутатором, що перешкоджає масштабуванню мережі. Спосіб впоратися з цим - запровадити допоміжний ЦП (Центральний Процесор)до комутатора або десь поблизу, щоб виконувати не лише площину керування, а й функціональні можливості площини даних, наприклад, дозволити ЦП пересилати потоки «мишей» або запроваджувати нові архітектури, які були б

більш виразними та дозволяли виконати більше дій, пов’язаних з обробкою пакетів.

Проблема апаратних обмежень стосується не лише фіксованих мереж, але й поширюється також на бездротові та мобільні домени. Площина бездротової передачі даних потребує обробки, щоб запропонувати більше корисних абстракцій подібно до того, що сталося з площиною даних фіксованих мереж. У той час як абстракції площини даних пропонуються такі протоколи, як OpenFlow, підтримують ідею відокремлення керування від площиниданих, вони не можуть бути поширені на бездротові та мобільні мережі, якщо тільки не лежить в основі апаратного забезпечення (наприклад, комутатори в магістральних стільникових мережах і бездротові точки доступу) починає надавати настільки ж складні та корисні абстракції.

Незалежно від способу реалізації комутаторів SDN, це має бути зроблено. Очевидно, що для того, щоб нова парадигма набула популярності, потрібно забезпечити зворотну сумісність. Також є чисті комутатори SDN, які повністю не інтегровані для роботи лише з контролером, це гібридний підхід (тобто підтримка SDN разом із традиційними операції та протоколи), які, мабуть, були б найуспішнішими на цих ранніх етапах ступенях розвитку SDN. Причина в тому, що в той час як функції SDN є переконливими рішення для багатьох реалістичних сценаріїв, інфраструктура в більшості корпоративних мереж все ще дотримується традиційного підходу. Тому проміжна гібридна форма, ймовірно, полегшить перехід на SDN.

2.3 Контролери SDN

Як уже згадувалося, однією з основних ідей філософії SDN є наявність мережевої операційної системи, розміщеної між мережевою інфраструктурою і прикладним рівнем. Ця мережева операційна система відповідає за координацію та управління ресурсами всієї мережі та для виявлення абстрактного уніфікованого представлення всіх компонентів до

програм, які виконуються поверх нього. Це ідея аналогічна тій, що дотримується типова комп'ютерна система, де операційна система знаходиться між апаратним забезпеченням і простором користувача і відповідає за управління апаратними ресурсами та надання загальних послуг для програм користувача. Подібним чином мережеві адміністратори та розробники тепер мають а однорідне середовище легше програмувати та налаштовувати.

Логічно централізоване керування та узагальнена мережева абстракція, яка робить модель SDN застосовною для більш широкого спектру додатків і гетерогенною мережевих технологій порівняно зі звичайною мережевою парадигмою. Наприклад, розглянемо неоднорідне середовище, що складається з фіксованого та бездротового зв'язку, мережа складається з великої кількості пов'язаних мережевих пристроїв (маршрутизаторів, комутаторів, бездротові точки доступу, проміжні пристрої тощо). У традиційній мережевій парадигмі кожен мережевий пристрій вимагатиме індивідуальної низькорівневої конфігурації мережею адміністратора для належної роботи. Крім того, оскільки кожен пристрій націлений на іншу мережеву технологію, вона матиме своє власне управління,що вимагає надзусиль до конфігурації, для адміністратора, щоб уся мережа працювала належним чином. З іншого боку з логічно централізованим керуванням SDN, адміністратору не доведеться турбуватися про деталі низького рівня. Натомість керування мережею здійснюватиме визначення належної політики високого рівня, залишаючи відповідальність за операційну систему мережі для зв'язку та налаштування роботи мережевих пристроїв.

2.4 Централізація управління в SDN

Як уже обговорювалося, архітектура SDN визначає, що мережева інфраструктура логічно контролюється центральним органом, відповідальним за управління і забезпечення виконання політики. Проте слід

уточнити, що логічно централізований контроль не обов'язково також передбачає фізичну централізацію.

Були різні пропозиції щодо фізично централізованих контролерів, наприклад для приклад NOX та Maestro. Фізично централізований дизайн керування спрощує реалізацію контролера. Всі комутатор управляються однієюфізичною особою, що означає, що мережа не підлягає проблемам, пов’язаним із узгодженістю, при цьому всі програми бачать однаковий стан мережі (який походить від того самого контролера). Незважаючи на свої переваги, цей підхід має ті ж недоліки, що й це роблять усі централізовані системи, тобто контролер діє як єдина точка відмови для вся мережа. Щоб подолати це, підключіть кілька контролерів до комутаторів, що дозволяє резервному контролеру взяти на себе роботу в разі збою. В цьому випадку, усі контролери повинні мати узгоджене уявлення про мережу, інакше програми може не працювати належним чином. Крім того, централізований підхід може підвищити масштабованість занепокоєння, оскільки всіма мережевими пристроями має керувати один і той самий об’єкт.

Один підхід, який додатково узагальнює ідею використання кількох контролерів мережі повинен підтримувати логічно централізоване, але фізично децентралізовану площину управління. У цьому випадку кожен контролер відповідає за управління лише однією частиною мережі, але всі контролери спілкуються та підтримують загальний вигляд мережі. Тому додатки сприймають контролер як єдине ціле, а насправді операції управління виконуються розподіленою системою. Перевага цього підходу, крім того, що більше немає жодної точки відмови, це підвищення продуктивності і масштабованість, оскільки кожен повинен керувати лише частиною мережі окремий компонент контролера. Деякі відомі контролери, які належать до цього категорії Onix і HyperFlow. Один потенційний мінус децентралізації управління ще раз пов'язане з узгодженістю стану мережі між контролерами. Оскільки стан мережі є розподіленим, можливо, що програми,

які обслуговуються різними контролерами, можуть мати різний погляд на мережі, через що вони можуть працювати неправильно.

Варто використовувати гібридне рішення, яке намагається охопити як масштабованість, так і послідовність два рівні контролерів, як це робить контролер Kandoo. Нижній шар є складається з групи контролерів, які не мають знання про цілий стан мережі. Ці контролери виконують лише контрольні операції, які потребують знання стану окремого комутатора (лише локальна інформація). З іншого боку верхній шар целогічно централізований контролер, відповідальний за виконання операцій у всій мережі які потребують знання всього стану мережі. Ідея полягає в тому, що локальні операції таким чином можна виконувати швидше й не створювати додаткового навантаження на високий рівень центрального контролеру, що ефективно підвищує масштабованість мережі.

Окрім ідей, пов’язаних із рівнем фізичної централізації контролерів, були й інші запропоновані рішення, пов’язані з їх логічною децентралізацією. Ідея логічної децентралізації походить безпосередньо з ранньої ери програмування мережах і з проекту Tempest. Нагадаємо, що архітектура Tempest дозволила кілька мереж віртуальних банкоматів для роботи поверх одного набору фізичних комутаторів. Подібним чином були пропозиції щодо проксі-контролерів SDN, таких як FlowVisor які дозволяють декільком контролерам використовувати одну площину пересилання. Мотивація оскільки ця ідея полягала в тому, щоб уможливити одночасне розгортання експериментальних і корпоративні мережі через ту саму інфраструктуру, не впливаючи одна на одну.

Перш ніж завершити опис про ступінь централізації з SDN контролерів, важливо вивчити занепокоєння, які можуть виникнути щодо їх продуктивність і застосовність у великих мережевих середовищах.

Однією з найбільш частих проблем, які викликають скептики SDN, є здатність SDN мережі для масштабування та реагування у випадках високого навантаження на мережу. Це занепокоєння походить головним чином з того

факту, що в новій парадигмі управління виходить за межі мережі пристроїв і входить до єдиного органу, відповідального за керування всією мережею. Мотивовані цією проблемою, дослідження продуктивності контролера SDN реалізації показали, що навіть фізично централізовані контролери можуть працюють дуже добре, маючи дуже низький час відгуку. Наприклад, було показано що навіть примітивні однопотокові контролери, такі як NOX, можуть впоратися із середнім значенням робоче навантаження до 200 тисяч нових потоків на секунду з максимальною затримкою 600 мс для мереж, що складаються з 256 комутаторів.Наприклад, NOXMT може обробляти 1,6 мільйона нових потоків на секунду в мережі з 256 комутаторами де середній час відгуку становить 2 мс у звичайної восьмиядерної машини з процесорами 2 ГГц. Новіші конструкції контролерів, орієнтовані на великі промислові сервери, обіцяють покращити продуктивність ще більше. Наприклад, контролер McNettle стверджує, що може для обслуговування мереж до 5000 комутаторів за допомогою одного контролера з 46 ядер з а пропускна здатність понад 14 мільйонів потоків за секунду та затримка менше 10 мс.

Інше важливе занепокоєння щодо продуктивності виникає у випадку фізичного децентралізованої площини керування – це спосіб розміщення контролерів у мережі, оскільки на продуктивність мережі може значно вплинути кількість і фізична розташування контролерів, а також за алгоритмами їх координації. Щоб вирішити цю проблему, були запропоновані різні рішення, починаючи з перегляду розміщення контролерів як задачі оптимізації для встановлення з’єднань цієї проблеми до областей локальних алгоритмів і розподілених обчислень для розробка ефективних протоколів координації контролерів.

Останнє питання, яке виникає у випадку фізично розподілених контролерів SDN пов'язані з узгодженістю стану мережі, що підтримується на кожному контролері, коли виконання оновлень політики через проблеми паралельності, які можуть виникнути через схильність до помилок розподіленої природи логічного контролера. Рішення такої проблеми можуть

бути подібними до транзакційних баз даних, з розширеним контролером транзакційний інтерфейс, що визначає семантику для повного введення політики оновлення або переривання.

2.5 Менеджмент трафіку

Інше дуже важливе питання проектування контролерів SDN пов’язане з тим, як це зробити регулювати трафік. Рішення щодо управління трафіком можуть мати прямий вплив на продуктивність мережі, особливо у випадках великих мереж, що складаються з багато комутаторів і з високим трафіком. Ми можемо розділити проблеми, пов'язані з управління трафіком на дві категорії; контроль деталізації та виконання політики.

- Контроль деталізації: деталізація контролю, що застосовується до мережевого трафіку, стосується того, наскільки дрібними чи грубими повинні бути операції перевірки контролера щодо проходження пакетів мережі. У звичайних мережах кожен пакет надходить на комутатор перевіряється індивідуально і приймається рішення про маршрутизацію щодо того, куди повинен відправлятися пакет пересилатися залежно від інформації, яку він містить (наприклад, адреса призначення). Поки цей підхід зазвичай працює для звичайних мереж, але цього не можна сказати про SDN. У цьому випадку пакетний підхід стає неможливим для реалізації поперек будь-якої великої мережі, оскільки всі пакети повинні проходити через контролер який мав би побудувати маршрут для кожного з них окремо. Через проблеми з продуктивністю, пов’язані з пакетним підходом, більшість SDN контролери дотримуються підходу на основі потоку, де кожен пакет призначається певному потоку відповідно до певної властивості (наприклад, адреси відпрвника та отримувча пакета, додаток з яким він пов’язаний). Контролер встановлює новий потік, досліджуючи перший пакет, що надходить для цього потоку, і

відповідно налаштовує комутатор. Щоб додатково розвантажити контролер, додатковим грубим підходом було б примусити виконувати керування на основі об’єднання потоку замість використання окремих потоків. Основним компромісом при дослідженні рівня деталізації є навантаження в контролер проти QoS, що пропонується мережевим програмам.Чим точніший контроль, тим вищий QoS. У пакетному підході контролер завжди може приймати найкращі рішення для маршрутизації кожного окремого пакета, що призводить до покращеного QoS. З іншого боку, посилення контролю над агрегуванням потоків означає, що рішення контролера щодо пересилання пакетів не приймаються повністю до мережі. У цьому випадку пакети можуть бути переслані через неоптимальний маршрут, що призводить до погіршення QoS.

- Виконання політики: друге питання в управлінні трафіком пов'язане з тим, як ця мережа політики застосовуються контролером до мережевих пристроїв. Один підхід, а потім такі системи, як Ethane, повинні мати модель реактивного керування, де комутаційний пристрій консультується з контролером кожного разу, коли потрібно прийняти рішення щодо нового потоку. У цьому випадку політика для кожного потоку встановлюється лише для комутаторів коли виникає реальний попит, що робить керування мережею більш гнучким. А потенційним недоліком цього підходу є погіршення продуктивності через час необхідний для того, щоб перший пакет потоку відправився до контролера для перевірки. Це зниження продуктивності може бути значним, особливо у випадках контролерів, які є фізично розташовані далеко від комутатора. Альтернативним підходом до застосування політики було б використання проактивної моделі контролю. У цьому випадку контролер завчасно заповнює таблиці потоків для будь-якого трафіку, який може проходити через комутатори, а потім надсилає правила всім комутаторам мережі. Використовуючи цей підхід,

комутатор більше не повинен запитувати вказівки контролер, щоб налаштувати новий потік і натомість виконати простий пошук у таблиці, що вже збережено в TCAM пристрою. Перевага проактивного контролю полягає в тому це усуває затримку, спричинену зверненням до контролера для кожного потоку.

-

2.6 Інтерфейси програмування SDN

Як уже згадувалося, зв'язок контролера з іншими рівнями є досягається через південний API для взаємодії контролера та комутатора та через північний API для взаємодії контролера та програми.

Південне сполучення

Комунікація на південь дуже важлива для маніпулювання поведінкоюкомутаторів SDN контролером. Це спосіб, яким намагається SDN «програмувати» мережу. Найвидатніший приклад стандартизованого південного руху API — OpenFlow. Більшість проектів, пов’язаних із SDN, припускають, що комунікація контролеру із комутаторами базується на OpenFlow, тому важливо зробити детальну презентацію підходу OpenFlow. Однак це слід зробити зрозуміло, що OpenFlow є лише одним (досить популярним) із багатьох можливих реалізацій взаємодій контролер-комутатор. Інші альтернативи, наприклад DevoFlowтакож існують, намагаючись вирішити проблеми продуктивності, з якими стикається OpenFlow.

Огляд OpenFlow. Дотримуючись принципу SDN роз’єднання площини керування та даних, OpenFlow забезпечує стандартизований спосіб керування трафіком у комутаторах та обміну інформації між комутаторами та контролером, як показано на рисунку 2.2. Комутатор OpenFlow складається з двох логічних компонентів. Перший компонент містить одну або більше таблиць потоків, відповідальних за збереження необхідної інформації комутатором для пересилання пакетів. Другим компонентом є OpenFlow

клієнт, який, по суті, є простим API, що забезпечує зв’язок комутатора з контролером.

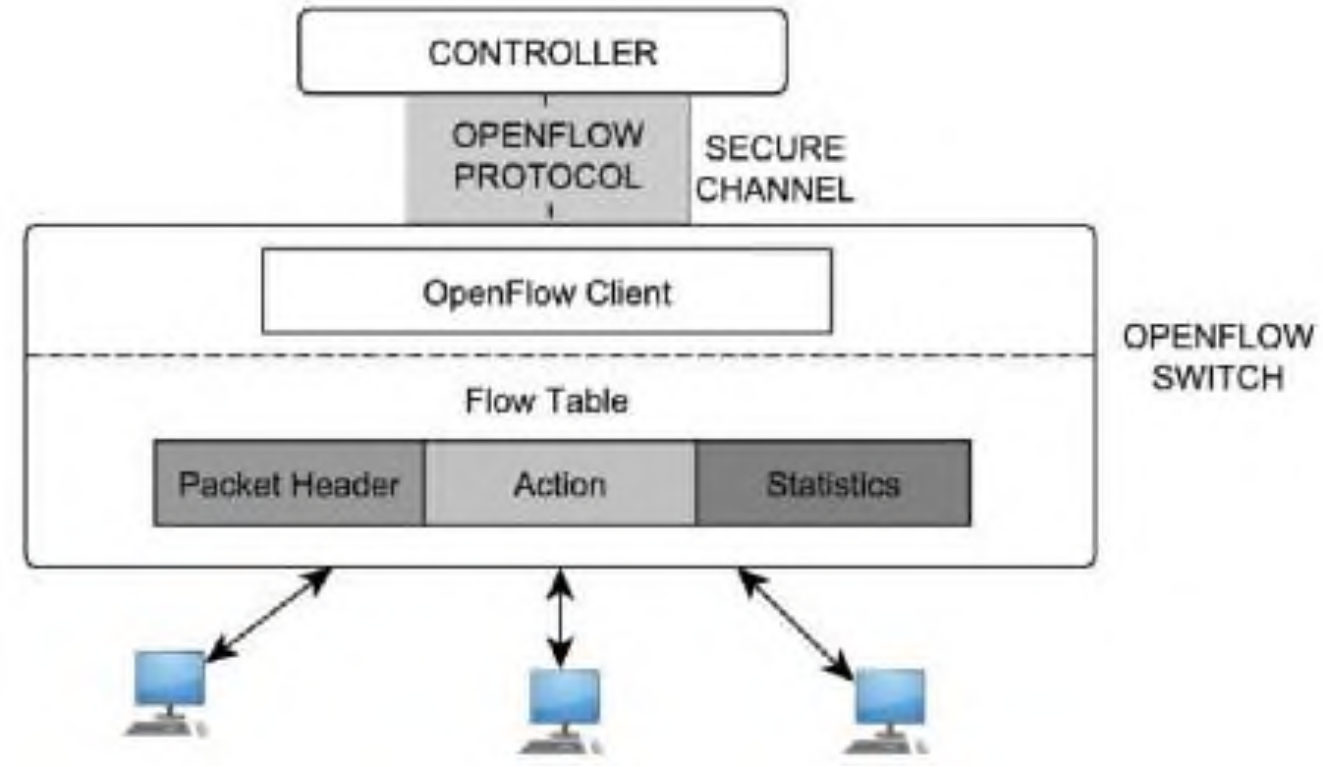


Рис. 2.2 Дизайн комутатора OpenFlow і зв'язок з контролером

Таблиці потоку складаються із записів потоку, кожна з яких визначає набір правил визначення того, як пакети, що належать до цього конкретного потоку, керуватимуться комутатором (тобто як вони будуть оброблятися та пересилатися). Кожен запис у таблиці потоків має три поля: 1) Заголовок пакета, що визначає потік, 2) Дія, що визначає, як пакет має бути оброблений і 3) Статистика, яка відстежує інформацію, наприклад кількість пакетів і байтів кожного потоку та час після пакета потоку було який було передано востаннє.

Коли пакет надходить на комутатор OpenFlow, його заголовок перевіряється, а пакет зіставляється з потоком, який має найбільш схоже поле заголовка пакета. Якщо знайдено відповідний потік, виконується дія, визначена в полі Дія. Ці дії включають пересилання пакета на певний порт для маршрутизації через мережу, пересилання пакета для перевірки контролером або відхилення пакета. Якщо пакет не можна зіставити з

жодним потоком, воно обробляється відповідно до дії, визначеної в записі потоку таблиці.

Обмін інформацією між комутатором і контролером відбувається шляхом надсилання повідомлень через безпечний канал у стандартизований спосіб, визначений протокол -OpenFlow. Таким чином, контролер може маніпулювати потоками, знайденими в таблиці потоків комутатора (тобто додати, оновити або видалити запис потоку) або проактивно, або реактивно, як було викладено в основних принципах контролера. Так як контролер вміє спілкуватися з комутатором за допомогою протоколу OpenFlow більше немає необхідності щоб мережеві оператори могли безпосередньо взаємодіяти з комутатором.

Північний API

Як уже обговорювалося, однією з основних ідей парадигми SDN є існування мережевої операційної системи, що лежить між мережевою інфраструктурою і службами та додатками високого рівня, подібно до комп'ютерної операційної системи, що лежить між апаратним забезпеченням і простором користувача. Припускаючи таку централізовану координаційну сутність та базування на основних принципах операційної системи, чітко визначений інтерфейс також повинен існувати в архітектурі SDN для взаємодії контролеру з додатками. Цей інтерфейс повинен надавати програмам доступ до базового обладнання, керувати системними ресурсами та дозволяти їх взаємодію з іншими програмами, не маючи жодних знань про мережеву інформацію низького рівня.

На відміну від південного сполучення, де взаємодія між комутаторами та контролером чітко визначені через стандартизований відкритий інтерфейс (тобто OpenFlow), наразі не існує прийнятого стандарту для взаємодії контролеру із додатками. Тому кожна модель контролера потребує забезпечення власними методами для здійснення зв'язку між контролером і додатком. Крім того, навіть інтерфейси, які реалізують поточні контролери, забезпечують абстракції дуже низького рівня (тобто потоком маніпуляції),

які ускладнюють реалізацію програми та часто різні відмінні цілі, які базуються на більш високому рівні концепції. Як приклад розглянемо програму керування живленням і брандмауером. Програма керування живленням має перенаправити трафік, використовуючи якомога менше посилань можливо, щоб вимкнути неактивні комутатори, тоді як брандмауеру можна отримати ці додаткові функції перемикається на маршрутизацію трафіку, після чого вони найкраще відповідають правилам брандмауера. Залишаючи програмісту вирішення цих конфліктів може стати дуже складним і громіздким процесом.

Для вирішення цієї проблеми було запропоновано багато ідей, які пропагують використання мовного мережевого програмування високого рівня, відповідних для політики перекладу на низький рівень. обмеження потоку, який, у свою чергу, використовується контролером для керування SDN комутаторами. Ці мови мережевого програмування також можна розглядати як проміжні рівень в архітектурі SDN, розміщений між прикладним рівнем і контролером подібно до того, як мови програмування високого рівня, такі як C++ і Python існує поверхневий асемблер для зберігання складних деталей низького рівня мови асемблера від програміста. Деякі приклади такого високого рівня мови мережевого програмування включають Frenetici Pyretic.

Домени програм SDN. Щоб продемонструвати застосовність SDN у широкому діапазоні мереж доменів, ми коротко наведемо два характерних приклади, в яких SDN може бути вигідними: центри обробки даних і стільникові мережі. Звичайно, список SDN додатків обмежені не лише цими доменами, але й розширені в багатьох інших, від корпоративних мереж, WLAN і гетерогенних мереж до оптичних мереж та інтернету речей.

2.7 Мережі Центрів Обробки Даних

Однією з найважливіших вимог до мереж центрів обробки даних є пошук способів масштабованості для підтримки сотень тисяч серверів і мільйонів віртуальних машини. Однак досягнення такої масштабованості може бути складним завданням для мережі. По-перше, разом із збільшенням кількісті серверів збільшується розмір таблиць пересилання, що призводить до потреби в більш складних і дорогих пристроїв пересилання. Крім того, управління трафіком і забезпечення виконання політики можуть стати дуже важливими та критичними проблемами, оскільки очікується, що центри обробки даних постійно знаходяться в стані високого рівня продуктивності.

У традиційних центрах обробки даних вищезазначені вимоги зазвичай виконуються шляхом ретельного проектування та конфігурації основної мережі. Це операція в більшості випадків виконується вручну шляхом визначення бажаних маршрутів для трафіку та шляхом розміщення проміжних блоків на стратегічних вузьких точках у фізичній мережі. Очевидно, що такий підхід суперечить вимозі масштабованості, оскільки вручну конфігурація може стати дуже складним і схильним до помилок завданням, особливо зізростанням розміру мережі. Крім того, стає все важче зробити центр обробки даних, що працює на повну потужність, оскільки він не може динамічно адаптуватися до вимоги додатків.

Переваги, які SDN пропонує для керування мережею, заповнюють ці прогалини. Завдяки відокремленню керування від площини даних пристроїв пересилання стає набагато більше простіше і тому дешевше. При цьому вся логіка управління делегується одному логічно централізоване утворення. Це дозволяє динамічно керувати потоками, навантаженням балансування трафіку та розподіл ресурсів у спосіб, який найкраще регулює функціонування дата-центру до потреб запущених програм, що в свою чергу призводить до підвищення продуктивності. Розміщення проміжних блоків у

мережі більше непотрібне, оскільки тепер можна забезпечити виконання політики через сутність контролеру.

3 ДОСЛІДЖЕННЯ ТА ПОРІВНЯННЯ СУЧАСНИХ ПРОГРАМНО-КОНФІГУРОВАНИХ РІШЕНЬ ДЛЯ МЕРЕЖ ЦОД ПРОВІДНИХ ВЕНДОРІВ

3.1 CiscoACI

Маючи уявлення про застосовані технології та архітектуру ми можемо ознайомитися з провідними рішеннями від виробників мережевого обладнання. Компанія Ciscoпропонує наступний варіант побудови VXLANSDNфабрики. Рішення складається з високопродуктивних комутаторів лінійки Nexusпід управління NX-OSта кластеру контролерів APIC.

Комутатори лінійки Nexus 9Кмають високопродуктивні ASICs та інтерфейси швидкостей від 10-25 Гб/сек, зазвичай використовують для підключення серверів, та від 40-100та навіть 400Гб/сек для зв’язку з комутаторами фабрики, мережевими екранами, зовнішніми L3 мережами.

Комутатори серії CiscoNexus 9500 розроблені з резервованими супервізорами, системними контролерами, джерелами живлення та панелями вентиляторів, що усуває єдину точку збою в шасі. Ці комутатори також підтримують до 6 модулів фабрики, які забезпечують резервування та плавне зниження комутаційної здатності у разі збою модуля фабрики. [5]

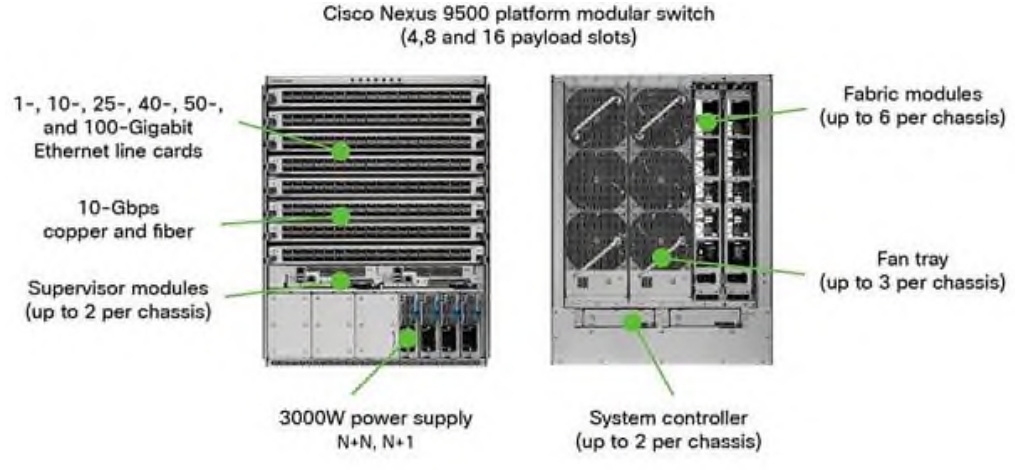


Рис. 3.1 Модульний комутатор Nexus 9500

Кластер контролерів APICможе бути як апаратний так і віртуальний. Необхідна потужність серверу де розміщений APICзалежить кількості пристроїв фабрики.APICне приймає участі в передачі трафіку. Фабрика самостійно приймає рішення щодо передачі потоків трафіку. Після втрати зв’язку з APICфабрика продовжить передавати трафік згідно з застосованою до нього востаннє конфігурацією. Площинаданих (інкапсуляція VXLAN) та площина управління (vPC, iBGP, EVPN, ACL)розташовані на комутаторах. APIC відіграє функцію системи керування, щополягає узастосуванні політик передачі трафіку та конфігурації до комутаторів фабрики. Конфігурація фабрики, що складається з низки пристроїв представлена як єдина сутність.Також система дає змогу застосовувати політики безпеки на основі заголовка рівня 4. Є можливість інтеграції з продуктами віртуалізації (VMwarevCenterandvShield, Microsoft Hyper-V та інші).Якщо ЦОДів, де розгорнутий CiscoACI 2 або більше, система може бути доповнена оркестраторомCiscoACIMulti-SiteOrchestrator.

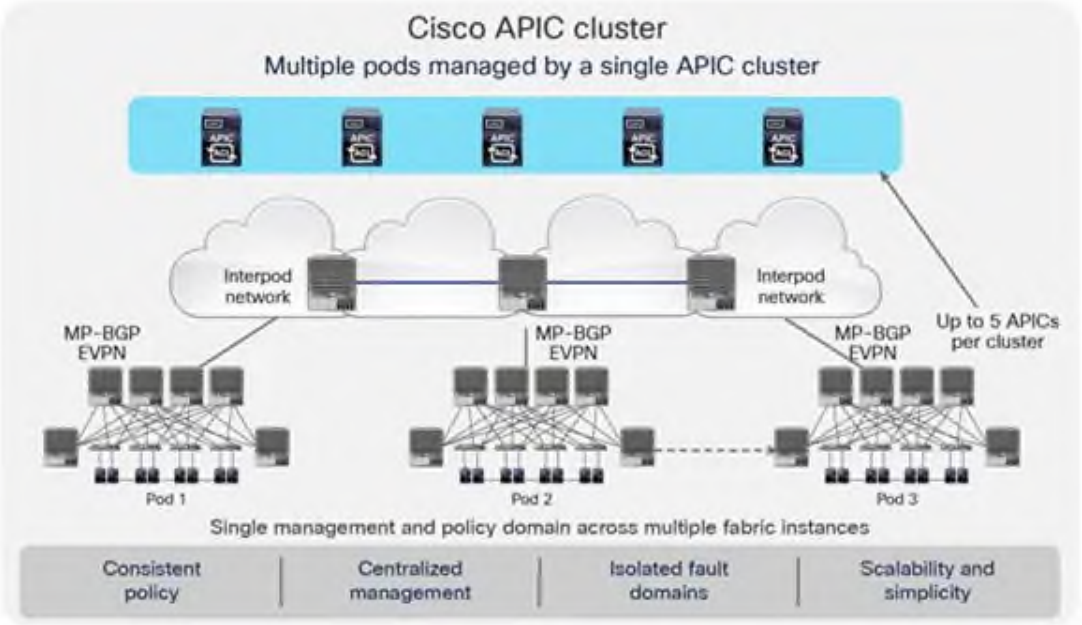


Рис.3.2 CiscoACI фабрика

3.2 Aruba Fabric Composer

Компанія ArubaNetworkstакож пропонує своє SDNрішення для побудови мережі ЦОД. ArubaFabricComposer це контролер який здатний будувати мережеві фабрики з комутаторів під управлінням Aruba CX. Наприклад близькою за характеристиками комутаторів Nexus 9Kє лінійкаArubaCX 8325, рисунок 2.3.Ця лінійка має аналогічні показники продуктивності інтерфейси, функції та протоколи, що застосовуються у побудові фабрики.[6]



Рис. 3.3 Aruba CX 8325Spineта Leaf комутатори

ArubaFabricComposer може оркеструвати окремий набір комутаторів AOS-CX як мережеву структуру через єдиний інтерфейс, який керує мережею. Прискорює надання інфраструктури за допомогою програмно-визначеної автоматизації та оркестрації. Доступна наскрізна мережева видимість підключених хостів,віртуальних машин. Система інтегрується з сучасними засобами віртуалізації.

3.3 Apstra

Компанія JuniperNetworks пропонує SDNрішення Apstra. JuniperApstra — це програмне, багатовендорне мережеве рішення на основі намірів, яке забезпечує замкнуту автоматизацію забезпечуючи рішення для повного керування структурою фабрики. Контролер має шаблони, що створюють розгортання та керування фабрикою побудованою на архітектурі Spine-Leaf.

Систему можна інтегрувати з VMware. Також є функціонал по створенню політик на основі заголовку рівня 3.

Комутатори серії QFXмають аналогічні параметри апаратної та функціональної частини, що в рішення описаних вище.



Рис. 3.4 Juniper QFX5200Spineта Leaf комутатори

3.4 Матриця порівняння рішень

При виборі будь якої системи, її майбутньому користувачеві доводитися обрати те чи інше рішення керуючись певними критеріями. Одним з таких може бути ціна рішення або характеристики апаратної частини, що показані в таблиці 3.1.

Таблиця 3.1

Матриця порівняння апаратної частиниLeaf комутаторів фабрики

	Cisco N9K-C93180YC-FX3	Aruba 8325-48Y8C	Juniper QFX5120-48YM-AFO
SFP28	48	48	48
QSFP28	6	8	8
Прод.	3,6 Тб/сек	6,4 Тб/сек	4Тб/сек
Прод.	1,2 Млн п/сек	2 Млн п/сек	2 Млн п/сек

Таблиця 3.2

Матриця порівняння апаратної частини Spine комутаторів фабрики

	Cisco N9K-C9332C	Aruba 8325-32C	Juniper QFX5120-32C-AFO
QSFP28	32	32	32
Прод.	6,4 Тб/сек	6,4 Тб/сек	6,4 Тб/сек
Прод.	4,4 Млн п/сек	2 Млн п/сек	2 Млн п/сек

Таблиця 3.3

Матриця вартості компонентів фабрики запропонованої виробниками

	Cisco	Aruba	Juniper
Spine	$42 870	$49 243	$32 820
Leaf	$70 000	$47 083	$33 566
Контролер	$37 129	$32 552	$70 880
Модулі	~ $10 000	~ $10 000	~ $10 000
Загальна вартість	$272 869	$235 204	$213 652

Мінімально фабрика складається з двох Spineта Leafкомутаторів, контролеру або кластеру контролерів та оптичних модулів якими з’єднується обладнання. Рядок загальної вартості в таблиці 2.3містить суму від мінімально необхідної кількості обладнання для побудови фабрики.

ВИСНОВКИ

На сьогоднішній день для модернізації та вдосконалення роботи ЦОД широко застосовуються технології віртуальних накладених мереж (Overlaynetwork)у поєднання з SDN (Software-definedNetworking) контролером.

Як результат, така мережа є легко масштабованою, досить простою у розгортанні. Керування мережею значно спрощується завдяки використанню контролера у порівнянні з традиційними мережами, де необхідно конфігурувати кожен вузол окремо. Такий підхід дозволяє автоматизувати розгортання нових сервісів, тим самим спростивши експлуатацію мережі, що дозволить збільшити доступність сервісу та зосередитися на потребах бізнесу, не відволікаючись на траблшутінг.

Основним обладнання мережі є комутатори, що підтримують інкапсуляцію VXLAN. Вони містять високопродуктивні процесори ASIC, для пересилання трафіку. Основним інтерфейсом для підключення до серверів – SFP28, підключення між мережевим обладнанням фабрики – QSFP28.Більшість рішень від провідних вендорів мають підтримку утворення каналів між декількома ЦОД з єдиною точкою керування. Завдяки використанню топології Clos масштабування мережі відбувається досить просто. При необхідності додати інтерфейсів доступу додається один Leafкомутатор та підключається до кожного Spine, якщо необхідно наростити пропускну швидкість, то необхідно додати Spinekомутатор та підключити з кожним Leaf вузлом. Як результат оператори ЦОД мають масштабоване, автоматизоване мережеве рішення, що забезпечує високу доступність сервісів.

ПЕРЕЛІК ПОСИЛАНЬ

1. https://www.fortunebusinessinsights.com/cloud-computing-market-102697
2. https://uk.wikipedia.org/wiki/Virtual_Extensible_LAN
3. https://www.juniper.net/us/en/research-topics/what-is-vxlan.html#:~:text=VXLAN%20is%20an%20encapsulation%20protocol,the%20use%20of%20virtual%20networks.
4. https://www.juniper.net/documentation/en_US/day-one-books/TW_DCDeployment.v2.pdf
5. https://www.cisco.com/c/en/us/products/collateral/cloud-systems-management/application-policy-infrastructure-controller-apic/datasheet-c78-739715.html
6. https://www.arubanetworks.com/products/switches/core-and-data-center/fabric-composer/

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ

Магістерська кваліфікаційна робота
на тему:

“Методика вдосконалення роботи ЦОД на базі застосування програмно-конфігурованих мереж ”

Виконав: студент групи ТСДМ-63 Тетеря Р.В.
Керівник: КТН, доцент Лаврінець К.Г.

Київ 2025

Мета та завдання

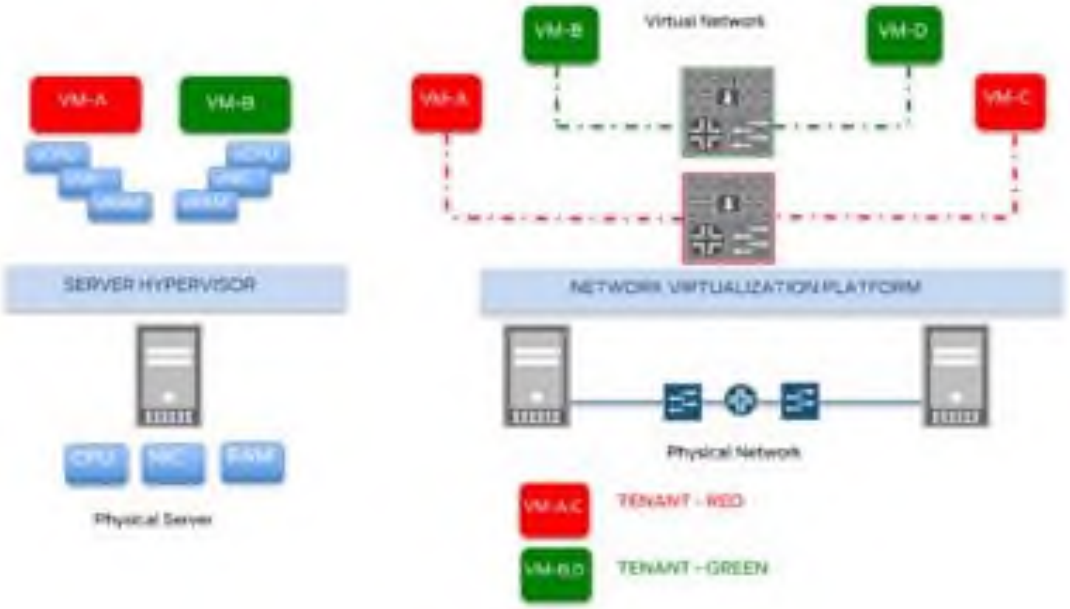
Об`єкт дослідження – побудова моделі SDN мережі ЦОД з інкапсуляцією VXLAN.

Предмет дослідження –телекомунікаційна мережа ЦОД.

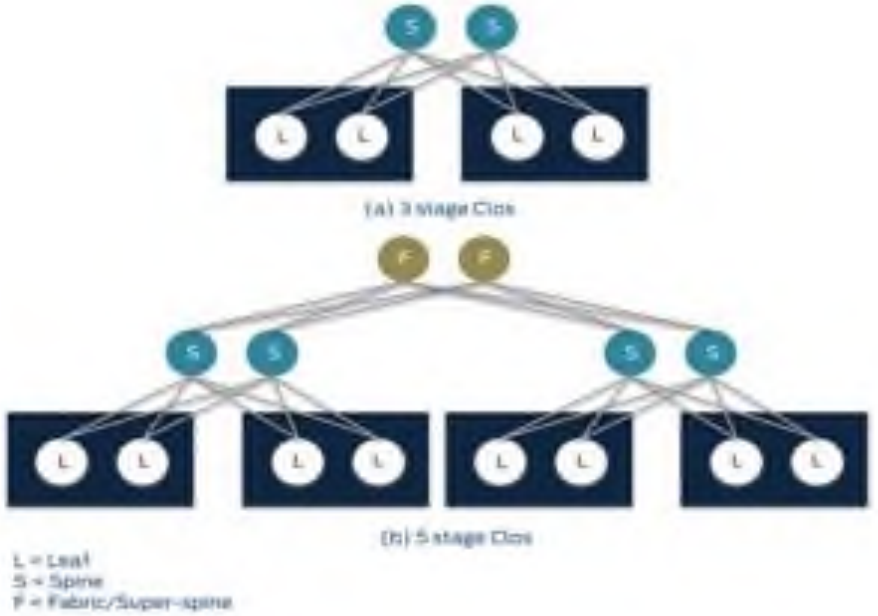
Мета роботи – дослідження вдосконалення роботи телекомунікаційних мереж ЦОД на основі сучасних рішень.

Метод дослідження – теорія електрозв’язку, теоретичної радіотехніки, математичного та комп’ютерного імітаційного моделювання.

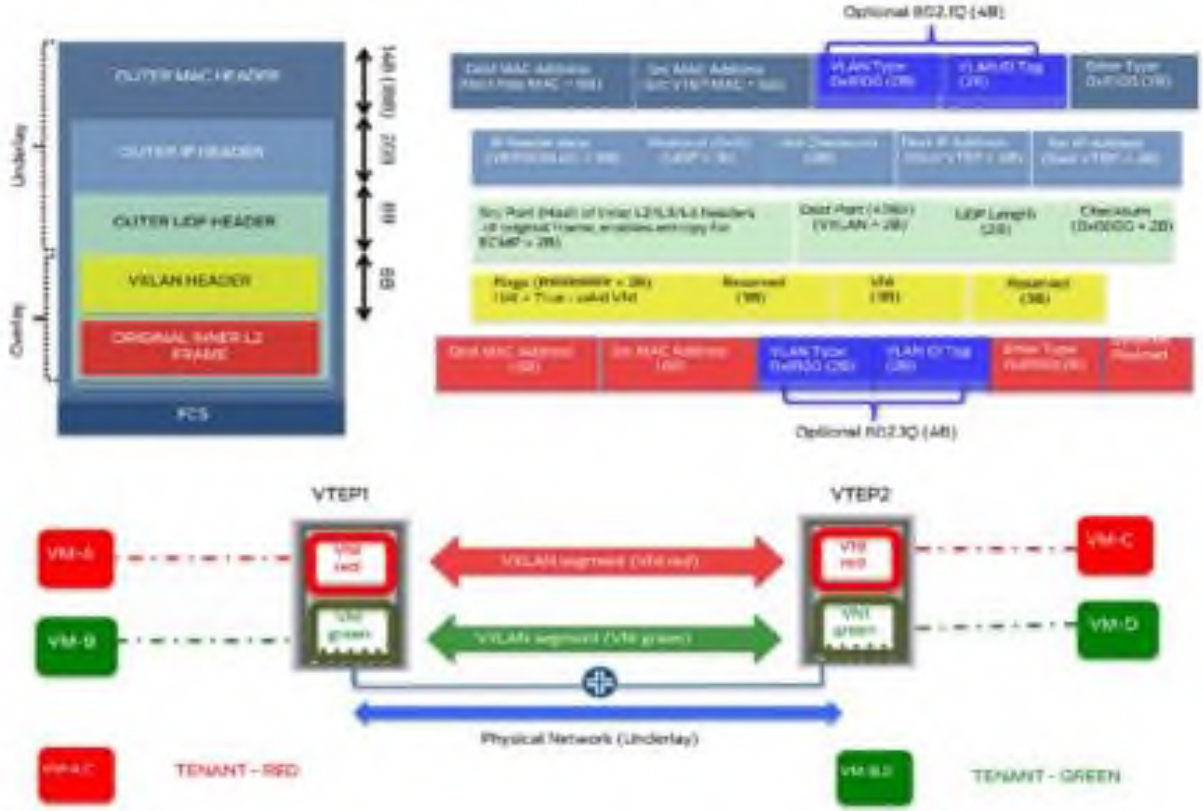
Серверна та мережеві інфраструктури

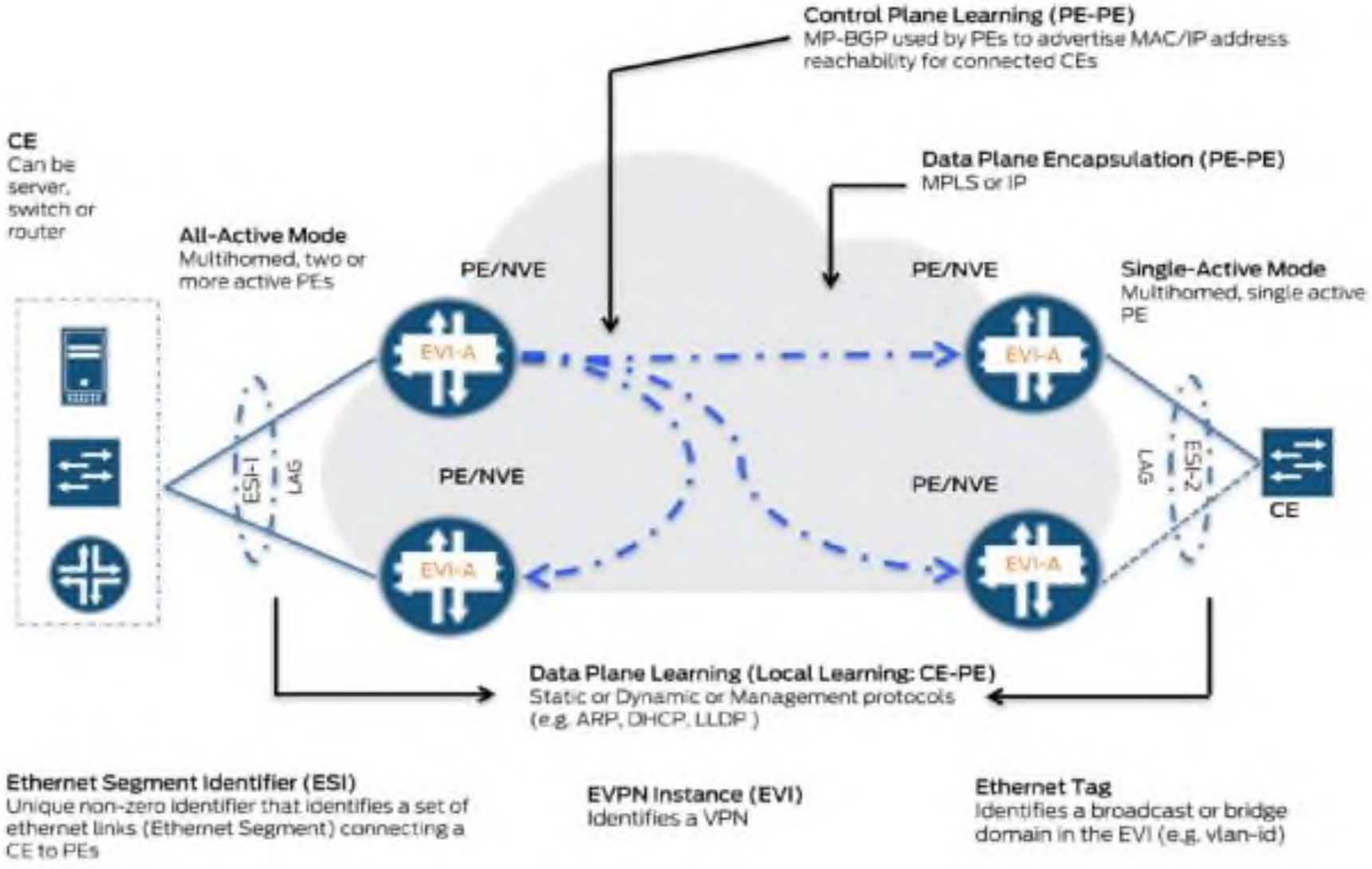


3/ 5-рівнева Clos архітектура IP-фабрики

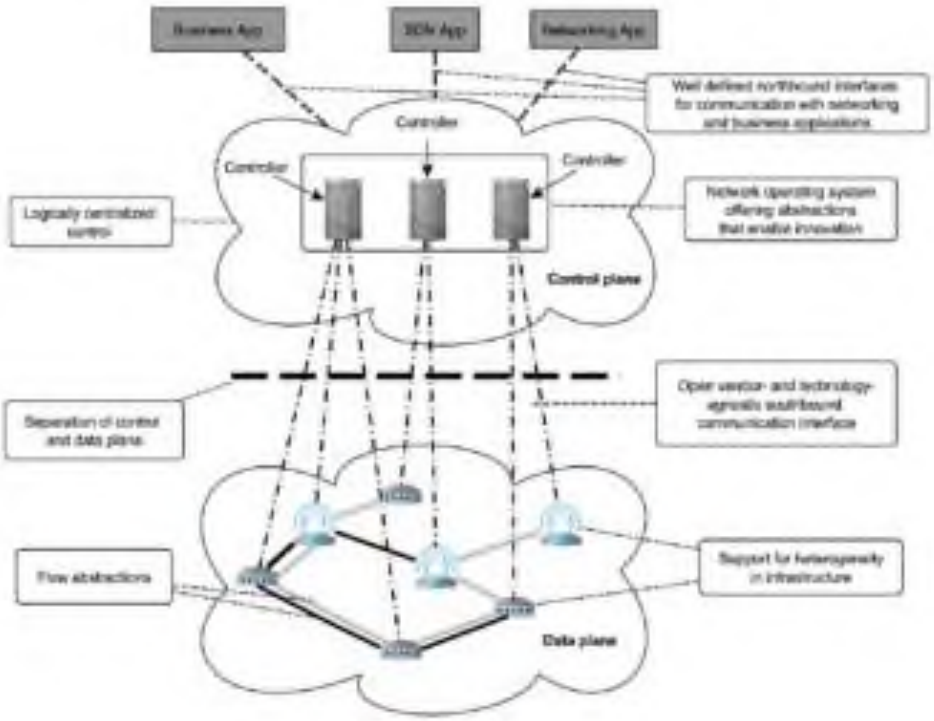


Формат кадру VXLAN і термінологія

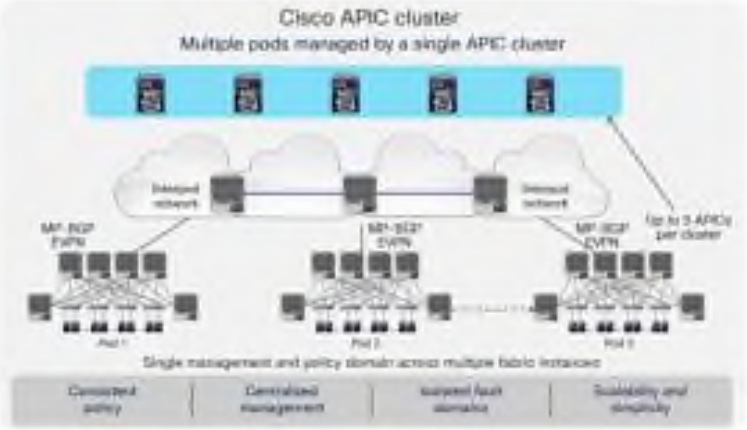



Control Plane Learning (PE-PE)
MP-BGP used by PEs to advertise MAC/IP address reachability for connected CEs
CE
Can be server, switch or router
Data Plane Encapsulation (PE-PE)
MPLS or IP
All-Active Mode
Multihomed, two or more active PEs
PE/NVE
PE/NVE
Single-Active Mode
Multihomed, single active PE
EVI-A
EVI-A
ESI-1
LAG
PE/NVE
PE/NVE
LAG
ESI-2
CE
EVI-A
EVI-A
Data Plane Learning (Local Learning: CE-PE)
Static or Dynamic or Management protocols
(e.g. ARP, DHCP, LLDP)
Ethernet Segment Identifier (ESI)
Unique non-zero identifier that identifies a set of ethernet links (Ethernet Segment) connecting a CE to PEs
EVPN Instance (EVI)
Identifies a VPN
Ethernet Tag
Identifies a broadcast or bridge domain in the EVI (e.g. vlan-id)

Концепція SDN мережі



Cisco ACI


Cisco APIC cluster
Multiple pods managed by a single APIC cluster
Single management and policy domain across multiple fabric instances


1-, 10-, 25-, 40-, 50-, and 100-Gigabit Ethernet line cards
10-Gbps copper and fiber
Supervisor modules (up to 2 per chassis)
Fabric modules (up to 6 per chassis)
Fan tray (up to 3 per chassis)
System controller (up to 2 per chassis)

Аналоги від Aruba Networks та Juniper Networks



Aruba 8325



QFX5200-32C



QFX5200-48Y

Матриця порівняння апаратних потужносте та вартості обладнання

	Cisco N9K-C93180YC-FX3	Aruba 8325-48Y8C	Juniper QFX5120-48YM-AFO
SFP28	48	48	48
QSFP28	6	8	8
Прод.	3,6 Тб/сек	6,4 Тб/сек	4 Тб/сек
Прод.	1,2 Млн п/сек	2 Млн п/сек	2 Млн п/сек

	Cisco N9K-C9332C	Aruba 8325-32C	Juniper QFX5120-32C-AFO
QSFP28	32	32	32
Прод.	6,4 Тб/сек	6,4 Тб/сек	6,4 Тб/сек
Прод.	4,4 Млн п/сек	2 Млн п/сек	2 Млн п/сек

	Cisco	Aruba	Juniper
Spine	$42 870	$49 243	$32 820
Leaf	$70 000	$47 083	$33 566
Контролер	$37 129	$32 552	$70 880
Модулі	~ $10 000	~ $10 000	~ $10 000
Загальна вартість	$272 869	$235 204	$213 652

Висновок

На сьогоднішній день для модернізації та вдосконалення роботи ЦОД широко застосовуються технології віртуальних накладених мереж (Overlay network) у поєднання з SDN (Software-defined Networking) контролером.
Як результат, така мережа є легко масштабованою, досить простою у розгортанні. Керування мережею значно спрощується завдяки використанню контролера у порівнянні з традиційними мережами, де необхідно конфігурувати кожен вузол окремо. Такий підхід дозволяє автоматизувати розгортання нових сервісів, тим самим спростивши експлуатацію мережі, що дозволить збільшити доступність сервісу та зосередитися на потребах бізнесу, не відволікаючись на траблшутінг.
Як результат оператори ЦОД мають масштабоване, автоматизоване мережеве рішення, що забезпечує високу доступність сервісів.