

УДК 004.051

DOI: 10.31673/2786-8362.2025.013075

Гніденко М.М.; Шепетун О.О.

ПРОБЛЕМИ МАСШТАБОВАНOSTІ ПРОГРАМНО-ВИЗНАЧЕНИХ МЕРЕЖ

Hnidenko M.M., Shepetun O.O. Scalability problems of software-defined networks. Software-defined networking (SDN) has enormous potential to transform network infrastructure, but there are significant challenges on the road to widespread adoption. And scalability is a key challenge. Given the rapid growth in data volumes, the number of connected devices, and bandwidth requirements, the ability of SDN to scale effectively is critical. If an SDN controller cannot cope with a large number of devices and data flows, this can lead to performance degradation, latency, and even network failures. Research initiatives aimed at addressing scalability issues in SDN are extremely relevant. These could include developing more efficient flow control algorithms. This would allow the controller to handle a larger number of requests without compromising performance. Addressing the scalability issue, along with other issues such as security, legacy compatibility, and management complexity, is key to realizing the full potential of software-defined networks and providing operators with the efficient, flexible, and scalable infrastructure they need. SDN scalability studies show that these problems are often not caused by SDN and are not fundamentally unique to it, and most of these problems can be solved without losing the benefits of SDN. If a network has tens of thousands of switching elements and can grow rapidly, the sheer number of control events generated in any network of this scale is enough to overwhelm any centralized controller. One way to solve this problem is to proactively enforce rules on the switches, effectively eliminating most control requests before they reach the control plane. Nodes in SDN networks can be geographically distributed, and the large diameter of these networks exacerbates the scalability problems of controllers. Physical partitioning of the network can be used to divide it into separate regions; each partition can be managed by an independent controller, and these controllers can exchange only the necessary state change events, effectively hiding most events from external controllers.

Keywords: Software-defined network (SDN), SDN controller, scalability, performance, control plane, data plane, ASIC, Open vSwitch

Гніденко М.М., Шепетун О.О. Проблеми масштабованості програмно-визначених мереж. Програмно-визначені мережі (SDN) несуть величезний потенціал для трансформації мережевої інфраструктури, але на їхньому шляху до широкого впровадження існують серйозні виклики. І проблема масштабованості є однією з ключових. З огляду на стрімке зростання обсягів даних, кількості підключених пристроїв та вимог до пропускної здатності, здатність SDN ефективно масштабуватися є критично важливою. Вирішення проблеми масштабованості, поряд з іншими викликами, такими як безпека, сумісність зі старим обладнанням та складність управління, є ключем до реалізації всього потенціалу програмно-визначених мереж і забезпечення операторам тієї ефективної, гнучкої та масштабованої інфраструктури, якої вони потребують. Дослідження масштабованості SDN показує, що ці проблеми часто викликані не SDN і не є принципово унікальними для нього, більшість цих проблем можна вирішити без втрати переваг SDN. Якщо мережа має десятки тисяч комутаційних елементів і може швидко зростати, самої кількості подій керування, що генеруються в будь-якій мережі такого масштабу, достатньо, щоб перевантажити будь-який централізований контролер. Один із способів вирішення цієї проблеми – проактивне встановлення правил на комутаторах, що ефективно усуває більшість запитів на керування, перш ніж вони потраплять до площини керування. Вузли в мережах SDN можуть бути географічно розподілені і великий діаметр цих мереж загострює проблеми масштабованості контролерів. Можна скористатися фізичним розподілом мережі, щоб розділити її на окремі регіони; кожен розділ може керуватися незалежним контролером і ці контролери можуть обмінюватися лише необхідними подіями зміни стану, ефективно приховуючи більшість подій від зовнішніх контролерів.

Ключові слова: Програмно-визначені мережі (SDN), контролер SDN, масштабованість, площина управління, площина передачі даних, ASIC, Open vSwitch

Вступ

Централізоване керування інфраструктурою на основі політики, заданої програмним додатком, є спільним визначенням сутності програмно-визначених мереж (SDN). Поділ на площину передачі даних (data plane) та площину управління (control plane) дозволяє обом рівням розвиватися незалежно та надає ряд переваг, таких як висока гнучкість, програмованість та можливість реалізації централізованого моніторингу та управління мережі [1-3]. Незалежність дозволяє площині управління розвиватися незалежно від площини даних,

що забезпечує швидші інновації, оскільки програмне забезпечення часто перевершує апаратне забезпечення за швидкістю інновацій. Крім того, логічна централізація має потенціал для спрощення роботи та управління мережею, забезпечуючи єдину точку, де можна оцінити наслідки дій управління та, можливо, відхилити їх, якщо вони призведуть до певного порушення операційних обмежень. Але незважаючи на вказані переваги, проблеми продуктивності та масштабованості завжди були притаманними для програмно-визначених мереж (SDN) з моменту їх створення. Масштабованість, як правило, охоплює широкий спектр аспектів і система масштабована лише настільки, наскільки масштабований її найменш масштабований компонент, тобто компонент, який першим стає вузьким місцем у міру збільшення розміру мережі. Дослідження, що стосуються масштабованості SDN, можна класифікувати за такими основними категоріями: площина даних, площина керування, застосування та наскрізні питання. Тобто основна проблема масштабованості знаходиться на площині даних.

В той же час, незалежно від можливостей контролера, центральний контролер не масштабується по міру зростання інфраструктури площини даних (збільшення кількості комутаторів, потоків, пропускної здатності тощо) і не зможе обробляти всі вхідні запити, забезпечуючи однакові гарантії обслуговування. Крім того, перші випробування контролерів SDN показали, що він може обробляти лише десятки тисяч ініціацій потоку в секунду, зберігаючи час встановлення потоку одиниці мілісекунд, посилили такі занепокоєння. Оскільки більшість ранніх пропозицій SDN базувалися на потоках, додаткова затримка для ініціювання потоку стала серйозною проблемою.

Постановка завдання. Для вирішення проблеми масштабування програмно-визначених мереж проведемо дослідження, у яких розкриємо сутність проблеми масштабованості в програмно-визначених мережах і це дозволить стверджувати, що вони не є унікальними для SDN. А тому вирішення проблеми її масштабованості може бути вирішено підходами, характерними для мереж взагалі. Проведемо також дослідження часто озвучених проблеми в різних умовах та обговоримо компроміси масштабованості в просторі проектування SDN. Крім того, перерахуємо недооцінені, але важливі можливості та проблеми масштабованості, що виходять за рамки загальноновживаних показників продуктивності.

Аналіз останніх досліджень. Існує велика кількість досліджень щодо вирішення проблеми масштабованості програмно-визначених мереж. Наприклад, перспективне рішення проблем масштабованості пропонується у відмовостійкій архітектурі SDN, яка масштабується до великих мереж завдяки механізму VLAN, встановленому в локальних комутаторах [4]. Описується також масштабована структура керування SDN під назвою McNettle, яка виконується на багатоядерних серверах із спільною пам'яттю та базується на Nettle. Експерименти показали можливість значного збільшення обсягу підтримуваної інфраструктури за допомогою контролера з великою кількістю ядер [5]. Описуються також вплив затримок налаштування потоку на комутаторах інфраструктури, що може стати проблемою для масштабованості мережі. Крім того, при збільшенні обсягу широкомовного мережевого трафіку та швидкому збільшенні кількості поточкових записів у таблиці потоків, обмежується масштабованість SDN.

Метою роботи є розгляд проблем, які впливають на обмеженість масштабованості програмно визначених мереж та дозволять визначити можливі шляхи їх вирішення. Деякі проблеми масштабованості SDN розглянемо більш детально, включаючи накладні витрати на налаштування потоку та стійкість до збоїв. Можна стверджувати, що, хоча ці проблеми не є специфічними для SDN, їх можна пом'якшити за допомогою альтернативного дизайну.

Виклад основного матеріалу дослідження

Що принципово відрізняє SDN від традиційних мереж передачі даних, так це відокремлення площини управління від площини даних. Це роз'єднання призводить до цікавих властивостей. Найважливіше, як показано на рисунку 1, що компоненти площини даних та управління можуть розвиватися незалежно, якщо ми визначимо стандартний інтерфейс

прикладного програмування) між ними. Елементи пересилання відповідають за комутацію і зазвичай будуються з високоспеціалізованих інтегральних схем, орієнтованих на певні програми (ASIC), які розвиваються значно повільніше, ніж компоненти площини керування, написані програмно. Крім того, можливість створення централізованого огляду мережі створює величезні потенційні можливості для спрощення програм керування, а отже, і для прискорення змін та інновацій у площині керування.

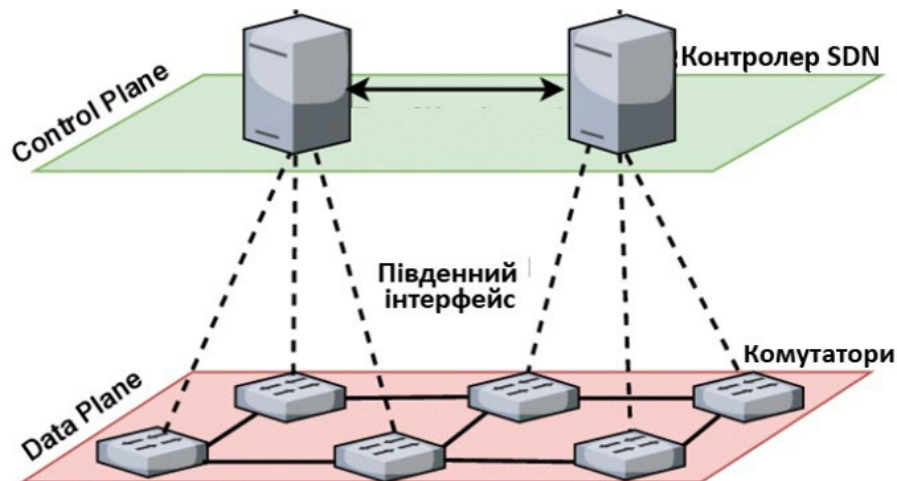


Рис. 1. Розділені площини даних та управління в SDN

Однак це роз'єднання має й свої підводні камені. По-перше, визначення стандартного інтерфейсу між двома площинами є абсолютно нетривіальним. Технічно цей інтерфейс повинен бути здатним обробляти потреби різних архітектур і повинен бути здатним сприяти незалежній еволюції обох площин. Крім того, всі або більшість постачальників комутаторів повинні використовувати один і той самий стандартний інтерфейс, щоб він був корисним; інакше мережі будуть прив'язані до певних постачальників, що може призвести до появи власницьких рівнів, що перешкоджатиме швидким змінам та інноваціям у мережах. По-друге, роз'єднання площин даних та управління створює проблеми масштабованості. Перенесення традиційних функцій локального керування на віддалений контролер потенційно може призвести до появи нових вузьких місць.

Ранні системи SDN, доручають контролеру встановлювати стан переадресації на комутаторах для кожного потоку. Навіть попри те, що ця реактивна форма обробки потоків забезпечує високий ступінь гнучкості, вона вводить затримку встановлення потоку та, залежно від реалізації, може обмежувати масштабованість. Ранні розробки, такі як Ethane та NOX, призводять до поширеного припущення, що всі системи SDN є реактивними. Однак насправді проактивні розробки, в яких записи переадресації встановлюються до початку фактичних потоків, цілком прийнятні в SDN і можуть повністю уникнути штрафу за затримку встановлення потоку.

Розглянемо процес налаштування потоку, щоб пояснити вузькі місця та показати, як хороший дизайн може їх уникнути. Як показано на рисунку 2, процес налаштування потоку складається з чотирьох кроків:

1. на комутатор надходить пакет, який не відповідає жодному запису в таблиці потоків;
2. комутатор генерує новий запит потоку до контролера;
3. контролер відповідає новим правилом переадресації;
4. комутатор оновлює свою таблицю поточкових записів.

Продуктивність на перших двох кроках і частково на останньому залежить від можливостей та ресурсів комутатора (керуючий процесор, пам'ять тощо) та продуктивності його програмного стеку. Затримка на третьому кроці визначається ресурсами контролера разом із продуктивністю програми керування. Нарешті, час оновлення таблиці поточкових записів комутатора сприяє затримці завершення процесу налаштування потоку.

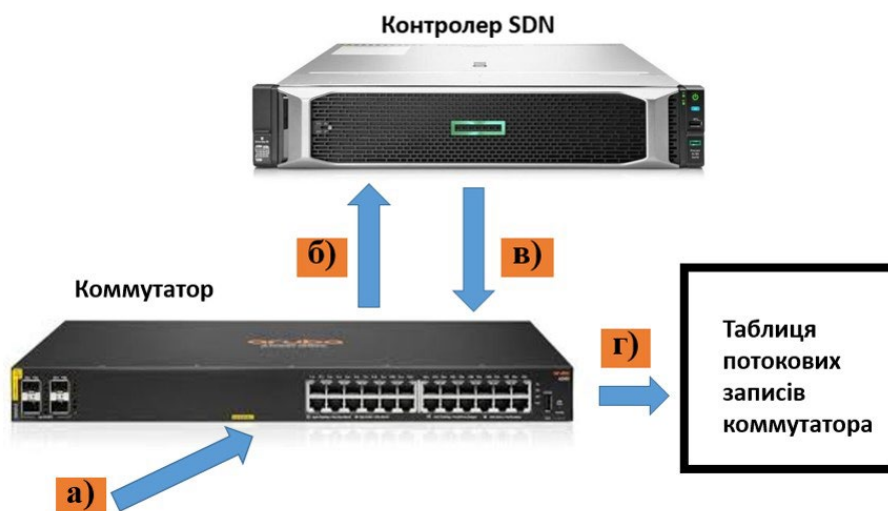


Рис. 2. Встановлення реактивного потоку в SDN

Якщо припустити, що контролери розміщені поблизу комутаторів, затримка зв'язку між контролером і комутатором є незначною. З боку контролера, навіть на звичайній машині з одним ядром процесора, сучасні контролери цілком здатні реагувати на запити на налаштування потоку протягом мілісекунди, тоді як запити на ініціацію потоку становлять близько кількох сотень тисяч за секунду.

Хоча Open vSwitch – програмний комутатор з підтримкою OpenFlow – здатний встановлювати десятки тисяч потоків на секунду з затримкою менше мілісекунди, апаратні комутатори підтримують лише кілька тисяч встановлень на секунду з затримкою менше 10 мс у кращому випадку. Така низька продуктивність зазвичай пояснюється браком ресурсів на комутаторах, поганою підтримкою високочастотного зв'язку з комутатором та процесором управління, а також неоптимальними програмними реалізаціями. Можна очікувати, що ці проблеми будуть вирішені, оскільки буде створено більш спеціалізоване обладнання. Передбачувано, що час оновлення таблиці поточкових записів стане основним фактором затримки налаштування потоку на стороні комутатора.

Хоча можна стверджувати, що контролери, а в найближчому майбутньому й комутатори, зможуть підтримувати достатню пропускну здатність з незначною затримкою для налаштування реактивного потоку, зрештою, логіка керування визначає масштабованість реактивного дизайну. Програма керування, яка встановлює наскрізний шлях для кожного потоку, не масштабується, оскільки пам'ять для кожного комутатора фіксована, але кількість записів пересилання в шляху даних зростає з кількістю активних потоків у мережі. Однак програма керування може встановлювати сукупні правила, що відповідають великій кількості мікропотоків (тим самим стикаючись з тими ж проблемами масштабованості, що й проактивний дизайн), або проактивно встановлювати правила в ядрі мережі, щоб забезпечити наскрізне з'єднання та визначити класи якості обслуговування (QoS), одночасно класифікуючи та реактивно маркуючи потоки на периферії.

Стійкість до збоїв та час конвергенції після збою завжди були ключовими проблемами продуктивності мережі. SDN не є винятком. У багатоконтролерній мережі, з відповідними механізмами виявлення контролера, комутатори завжди можуть виявити контролер, якщо такий існує в їхньому розділі. Тому, враховуючи масштабований механізм виявлення, збої контролера не створюють проблем для масштабованості SDN.

Давайте розберемо процес відновлення пошкодженого каналу або комутатора, щоб побачити, чим він відрізняється від традиційних мереж. Конвергенція у відповідь на збій каналу складається з п'яти кроків:

- комутатор виявляє зміну;
- комутатор повідомляє контролер про виявлені зміни;

- після отримання сповіщення контролер обчислює дії з відновлення;
- контролер надсилає оновлення до уражених елементів шляху даних;
- комутатор оновлюють свої таблиці пересилання.

У традиційних мережах сповіщення про збій з'єднання поширюються по всій мережі, тоді як в SDN ця інформація надсилається безпосередньо до контролера. Тому затримка поширення інформації в SDN не гірша, ніж у традиційних мережах. Також перевагою SDN є те, що обчислення виконуються на більш потужних контролерах, на відміну від слабких процесорів усіх комутаторів, незалежно від того, чи постраждали вони від збою, чи ні.

Необхідно звернути увагу, що вищенаведений аргумент був побудований на неявному припущенні, що несправний комутатор або з'єднання не впливає на канал зв'язку між комутатором і контролером. Саму мережу керування потрібно спочатку відремонтувати, якщо несправне з'єднання або комутатор був її частиною. У такому випадку, якщо мережа керування, побудована з використанням традиційного мережевого обладнання, використовує протоколи маршрутизації IGP, то IGP спочатку повинен сходитися, перш ніж комутатори зможуть зв'язатися з контролером для відновлення мережі передачі даних. Тому в цьому критичному випадку конвергенція може бути повільнішою, ніж у традиційних мережах. Якщо це виявиться проблемою, оператор мережі повинен розгорнути позасмугову мережу керування, щоб вирішити цю проблему.

Загалом, процес відновлення після збоїв у SDN не гірший, ніж у традиційних мережах. Отже, існують аналогічні проблеми масштабованості і ті ж методи, що використовуються для мінімізації часу простою в традиційних мережах, застосовні до SDN. Наприклад, проектування SDN може і повинно також використовувати локальні механізми швидкого перемикавання на резервні пристрої, доступні в комутаторах, для тимчасового перенаправлення трафіку на попередньо запрограмовані резервні шляхи, поки вирішується проблема збою. Платформа керування забезпечує основні механізми аварійного перемикавання і відновлення, які програми керування можуть повторно використовувати та на які вони можуть покладатися.

До цього часу досліджувалися найважливіші проблеми масштабованості в SDN, визначаючи різні показники, на які потенційно може вплинути зростання мережі. У подальшому використовуємо інший підхід, пояснюючи закономірності та недоліки масштабованості в різних типах мереж. Як приклади, були обрані мережі центрів обробки даних та постачальників послуг, оскільки вони мають різноманітні умови та різні вимоги. Тут не можна надати весь вичерпний перелік мереж. Мета полягає у визначенні закономірностей, які можна узагальнити та використовувати як орієнтири для розуміння проблем масштабованості в інших мережах.

Центри обробки даних. Типова мережа центру обробки даних має десятки тисяч комутаційних елементів і може швидко зростати. Самої кількості подій керування, що генеруються в будь-якій мережі такого масштабу, достатньо, щоб перевантажити будь-який централізований контролер. Один із способів вирішення цієї проблеми – проактивне встановлення правил на комутаторах, що ефективно усуває більшість запитів на керування, перш ніж вони потраплять до площини керування. Очевидною ціною тут, звичайно, є втрата точності та реактивності контролера.

Коли застосунку потрібна точна статистика потоку та/або швидкість реагування, його можна розгорнути поблизу комутаторів. Наприклад, часті події можна делегувати процесам, що працюють на кінцевих хостах, за умови мінімізації доступу до глобального стану. Враховуючи доступність обчислювальних ресурсів у мережах центрів обробки даних, такі рішення можна використовувати для досягнення довільних рівнів масштабованості. Розподілені контролери також можуть бути розумними рішеннями в мережах центрів обробки даних. Враховуючи низьку затримку в таких мережах, синхронізація стану та затримки налаштування потоку буде мінімальною та прийнятною для більшості застосунків.

Мережі постачальників послуг. Як правило, мережі постачальників послуг не мають стільки комутаторів/маршрутизаторів, скільки мають мережі центрів обробки даних; однак

вузли в таких мережах зазвичай географічно розподілені. Великий діаметр цих мереж загострює проблеми масштабованості контролерів, затримки налаштування потоків та конвергенції станів, а також вимоги до узгодженості. Можна скористатися фізичним розподілом мережі, щоб розділити її на окремі регіони; кожен розділ може керуватися незалежним контролером і ці контролери можуть обмінюватися лише необхідними подіями зміни стану, ефективно приховуючи більшість подій від зовнішніх контролерів. Враховуючи внутрішню затримку в таких мережах, усі керуючі програми повинні бути стійкими до затримки та мати слабкі вимоги до узгодженості.

Окрім високих затримок, мережі постачальників послуг зазвичай мають більшу кількість потоків, ніж інші мережі. Як наслідок, обмеження ресурсів шляху передачі даних також викликають занепокоєння. Агрегація потоків – це просте рішення, яке відбувається за рахунок деталізації керування. Зазначимо, що ці проблеми також присутні в традиційних мережах і не є унікальними для SDN.

Традиційно масштабованість мережі вивчається на основі показників продуктивності, тобто того, як змінюється певний показник продуктивності, коли масштабується мережа вздовж заданого виміру. На практиці існують інші ортогональні аспекти, які суттєво впливають на те, як система може адаптуватися до зростання. Наприклад, керованість (наскільки зручно керувати мережею, коли елементи мережі додаються, видаляються або змінюються у великих масштабах) та функціональна масштабованість (скільки зусиль потрібно для додавання нової функціональності до мережі) є такими ж важливими, як і продуктивність, для масштабованості мережі і їх не слід ігнорувати. Попередні результати поведінкових та програмних абстракцій, тестування та верифікації, а також розширюваності SDN показують, що у цьому SDN представляє значні можливості. Очевидно, що будуть серйозні труднощі в кожному з аспектів, перш ніж буде досягнуто повний потенціал SDN.

Поведінкові та програмні абстракції. Майже всі контролери SDN надають інтерфейси, які розкривають деталі, вбудовані в API між площинами даних та керування. Такий рівень деталізації є важливим для реалізації контролера, але зазвичай не є обов'язковим для керування мережею. Натомість мережевим операторам та програмістам потрібна абстракція, за допомогою якої вони можуть задати необхідну поведінку мережі з мінімальними зусиллями. З цієї метою більшість деталей та складнощів повинні оброблятися контролерами та бути прихованими від інтерфейсів керування. Такий інтерфейс забезпечить зростання мережі та полегшить керування в масштабі.

Тестування та верифікація. Без належних інструментів усунення несправностей стає серйозною проблемою для мереж у міру їх зростання. Це серйозна проблема для всіх типів мереж, включаючи SDN. Гарна новина полягає в тому, що SDN надає структурні блоки для тестування мережі, які недоступні в традиційних мережах. Більше того, враховуючи програму керування, можна змодельовати її поведінку та аналітично перевірити її для довільної мережі. Однак, враховуючи складність мереж, нелегко реалізувати інструмент, який може тестувати та верифікувати в масштабі.

Розширюваність. Поточні реалізації SDN використовують прагматичний підхід, підтримуючи лише добре відомі популярні протоколи, які легко підтримуються більшістю постачальників обладнання та мікросхем. У міру масштабування SDN нові протоколи або нові версії існуючих протоколів будуть заважати. Для підтримки цих нових протоколів необхідно модифікувати існуючі API. Рішенням цієї проблеми був би більш розширюваний та виразний API між площинами керування та даних, що є досить складним завданням. Завдання полягає в розробці виразних конструкцій, які можна реалізувати в ASIC та підтримувати непередбачені майбутні протоколи та вимоги. Хоча це не є прямою проблемою масштабованості, це може мати значний вплив на зростання та розширюваність SDN.

Висновки

Масштабованість була серйозною проблемою з моменту появи SDN. Сучасний обсяг досліджень масштабованості SDN показує, що ці проблеми не викликані SDN і не є

принципово унікальними для нього, більшість цих проблем можна вирішити без втрати переваг SDN. Поточні розгортання SDN у масштабі сховища даних підтверджують цей аргумент. У цій галузі зазвичай не враховується вплив SDN на інші обмежувальні фактори для зростання мереж, такі як мережеве програмування та складність управління. Програмно-визначені мережі додають рівень гнучкості, який може забезпечити мережеве програмування та управління в масштабі.

Список використаної літератури:

1. Rajat Chaudhary, Gagangeet Singh Aujla, Neeraj Kumar, Pushpinder Kaur Chouhan. A comprehensive survey on software-defined networking for smart communities. *Int J Commun Syst*. 2022. DOI:10.1002/dac.5296 – p. 50.
2. SDN: Software Defined Networks: An Authoritative Review of Network Programmability Technologies, Thomas D. Nadeau, Ken Gray. Copyright © 2013. All rights reserved. Printed in the United States of America. Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472. – p. 382
3. Гніденко М.П., Вишнівський В.В., Ільїн О.О. Побудова SDN мереж. – Навчальний посібник. – Київ: ДУТ, 2019. – 190 с.
4. Kim, H.J., Santos, J.R., Turner, Y., Schlansker, M., Tourrilhes, J., Feamster, N., "CORONET: Fault Tolerance for Software-Defined Networks," *Proceedings, 2012 20th IEEE International Conference on Network Protocols (ICNP)*, pp.1–2, October 30–November 2, 2018.
5. Voellmy, A., Wang, J.C., "Scalable Software-Defined Network Controllers," *Proceedings, ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 289–290, 2019.

Автори статті

Гніденко Максим – аспірант, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

ORCID: 0009-0008-4450-6472

Шепетун Олег – аспірант, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

ORCID: 0009-0006-4559-9477

Authors of the article

Hnidenko Maksym – postgraduate, State University of Information and Communication Technologies, Kyiv, Ukraine.

ORCID: 0009-0008-4450-6472

Shepetun Oleh – postgraduate, State University of Information and Communication Technologies, Kyiv, Ukraine.

ORCID: 0009-0006-4559-9477