

УДК 004

Б. Подолян, магістр гр. КІ-24М,*Центральноукраїнський національний технічний університет*

ДОСЛІДЖЕННЯ ТА ПРИНЦИПИ ПОБУДОВИ СИСТЕМИ ПРОЕКТУВАННЯ ТА ПОБУДОВИ ТОПОЛОГІЇ МЕРЕЖ У ЦЕНТРАХ ОБРОБКИ ДАНИХ

У статті розроблено програмне забезпечення, яке призначено для системи проектування та побудови топології мереж у центрах обробки даних. Метою розробки є дослідження та принципи побудови системи проектування та побудови топології мереж у центрах обробки даних. Об'єктом дослідження є процес проектування та побудови топології мереж у центрах обробки даних. Предметом дослідження є методи проектування та побудови топології мереж у центрах обробки даних. Методи дослідження базуються на методах побудови комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення. Результат роботи – програмна реалізація системи проектування та побудови топології мереж у центрах обробки даних. В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

проектування та побудова топології мереж, центр обробки даних

Постановка проблеми. Надійна топологія мережі центру обробки даних є важливою для забезпечення ефективної, масштабованої та безпечної роботи мережі в сучасному цифровому середовищі.

Ви інвестували у високоякісні мережеві комутатори та маршрутизатори. Ви оптимізували свої програми для обробки запитів з мінімальною затримкою. І все ж продуктивність мережі вашого центру обробки даних все ще низька.

Виною може бути неоптимальна топологія мережі центру обробки даних. Оскільки топології мережі відіграють ключову роль у визначенні ефективності потоку мережевого трафіку в центрі обробки даних, погані топології можуть знижувати продуктивність, навіть якщо ваші програми та інфраструктура в іншому розраховані на швидкість.

Аналіз останніх досліджень і публікацій. При аналізі останніх досліджень і публікацій [1-10] було виявлено певні прогалини у забезпеченні системи проектування та побудови топології мереж у центрах обробки даних.

Мета й завдання дослідження. Метою роботи є дослідження та принципи побудови системи проектування та побудови топології мереж у центрах обробки даних.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем проектування та побудови топології мереж у центрах обробки даних.
- Дослідження системи проектування та побудови топології мереж у центрах обробки даних.
- Програмна реалізація системи проектування та побудови топології мереж у центрах обробки даних.

Об'єктом дослідження є процес проектування та побудови топології мереж у центрах обробки даних.

Предметом дослідження є методи проектування та побудови топології мереж у центрах обробки даних.

Методи дослідження базуються на методах побудови комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення.

Виклад основного матеріалу. Існує кілька способів проектування топології мережі центру обробки даних. Ось найпоширеніші:

1. Трирівнева топологія

Традиційно більшість центрів обробки даних використовували так звану трирівневу мережеву топологію. Згідно з цією схемою, мережеві ресурси організовані на три рівні:

- Рівень доступу: забезпечує пряме підключення до пристроїв кінцевих користувачів, що дозволяє здійснювати зв'язок з мережею.

- Рівень розподілу: діє як посередник, агрегуючи трафік від комутаторів доступу та забезпечуючи дотримання політик для керування маршрутизацією та безпекою.

- Базовий рівень: забезпечує високошвидкісне та надійне з'єднання між рівнями розподілу та зовнішніми мережами, обробляючи трафік магістральної мережі.

Трирівнева топологія відносно проста в реалізації. Її головним недоліком є те, що вона не дуже гнучка та нелегка в масштабуванні, оскільки відносно невелика кількість комутаторів та маршрутизаторів повинна обробляти великий обсяг мережевого трафіку. Отже, ці пристрої зазвичай є дорогими моделями високого класу, а це означає, що якщо ви хочете збільшити їхню ємність, вам доведеться додати дорожче обладнання.

Додавання такого обладнання не є ні простим, ні швидким, і може бути неефективним з точки зору витрат, якщо обсяги трафіку коливаються, а розширені можливості комутації та маршрутизації потрібні лише зрідка. У такому разі ви отримаєте дороге обладнання, яке буде використовуватися недостатньо. З цих причин трирівнева топологія зазвичай не ідеальна для центрів обробки даних, обсяги мережевого трафіку яких коливаються, хоча вона може добре працювати, коли рівні трафіку є стабільними та передбачуваними.

2. Верхня частина полиці

У топології центру обробки даних «top-of-rack» ви встановлюєте мережеві комутатори в кожній серверній стійці. Таким чином, топологія «top-of-rack» спрощує швидке та економічно ефективно додавання потужності.

Оскільки обсяг трафіку, який має обробляти кожен комутатор, менший, ніж у трирівневій топології (де кілька стійок можуть бути підключені до одного комутатора), підхід «верхня частина стійки» може використовувати менш дорогі комутатори.

Таким чином, топологія «top-of-rack» спрощує дешеве та швидке додавання потужності. Конструкції «top-of-rack» також масштабуються природним чином залежно від потужності центру обробки даних, оскільки чим більше серверних стійок ви додаєте, тим більше комутаторів ви додаєте.

Недоліком є те, що вам доведеться встановлювати та обслуговувати більше комутаторів, що збільшує експлуатаційне навантаження на команди, відповідальні за розгортання та управління мережевим обладнанням. Конструкції, що встановлюються зверху стійки, також можуть знижувати ефективність у ситуаціях, коли серверна стійка не надсилає або не отримує достатньо трафіку для використання своїх комутаторів на повну потужність.

3. Перемикання мереж

Комутована структура мереж у деяких аспектах схожа на трирівневу топологію, але з більшою кількістю комутаторів її легше масштабувати. Вона також нагадує топологію «top-of-rack», з тією ключовою відмінністю, що сервери не призначені постійно до конкретного комутатора, що дозволяє ефективніше використовувати його потужність.

Комутована структура (fabric) у деяких аспектах схожа на трирівневу топологію, але в ній є більша кількість комутаторів, що полегшує масштабування. Вона також схожа на топологію "top-of-rack", з головною відмінністю в тому, що сервери не призначені до конкретного комутатора, тому легше використовувати потужність комутатора ефективніше.

Основним недоліком комутованих тканин є їхня складність; їх важко проектувати та впроваджувати, оскільки необхідно балансувати трафік та постійно реструктуризувати з'єднання між серверами та комутаційною тканиною.

4. Гібридна топологія

Гібридна топологія – це не окремий тип, а радше комбінація кількох підходів. Наприклад, ви можете розробити топологію, де деякі серверні стійки мають виділені комутатори, а інші сервери підключаються до більш гнучкої серверної структури. Такий підхід призведе до топології «top-of-rack» для однієї частини вашої мережі та комутованої структури для інших.

Як вибрати мережеву топологію для вашого центру обробки даних

Вибираючи, яка топологія мережі підходить для вашого центру обробки даних, слід врахувати такі ключові фактори:

– Потреби масштабованості: Якщо ви очікуєте високий ступінь коливань мережевого трафіку, конструкція з верхнім розташуванням стійки або комутована структура, ймовірно, буде кращою, оскільки вона легше масштабується.

– Фінансові ресурси: Трирівневі топології можуть мати вищі початкові витрати, оскільки вам потрібно купувати дорогі комутатори та маршрутизатори корпоративного класу. З іншого боку, конструкції верхньої стійки та комутованих мережевих структур часто вимагають вищих поточних витрат, оскільки вам потрібно додавати обладнання в міру масштабування та замінювати окремі комутатори, якщо вони виходять з ладу.

– Операційні ресурси: Якщо у вас менше технічних спеціалістів центру обробки даних, доступних для керування обладнанням, розгляньте можливість використання простішої топології, наприклад, трирівневої.

– Фізичні обмеження: Хоча мережеве обладнання та кабелі, що до нього підключаються, зазвичай не займають забагато місця порівняно із серверами, фізичні обмеження простору можуть бути проблемою, особливо в невеликих центрах обробки даних або серверних кімнатах. Якщо так, то трирівнева топологія, як правило, працює краще, оскільки вона не вимагає стільки окремих мережевих пристроїв або кабелів.

Використання хмарних обчислювальних сервісів зростає щороку [1] і має багатообіцяюче майбутнє [2] [3]. З 3,0 зеттабайтами IP-трафіку на рік та очікуваним трафіком 8,6 зеттабайтів у 2029 році [1], хмарний трафік зростає майже втричі. Однак цей трафік поділяється на три типи: від центрів обробки даних до користувачів, від центрів обробки даних до центрів обробки даних та всередині центрів обробки даних. Найбільша частина цього трафіку залишається всередині центрів обробки даних і відповідає типу трафіку всередині центрів обробки даних.

Щоб керувати таким величезним трафіком, приблизно 3342 екзабайти (EB) у 2025 році та 7566 EB, що очікуються у 2029 році [1], хмарні мережі перебувають під пильною увагою з метою пошуку нових рішень, що забезпечують достатню продуктивність та масштабованість. Серед цих нових рішень ми зосереджуємося як на TRILL (прозоре взаємозв'язок багатьох каналів) [4], так і на OpenFlow [5].

Перший має децентралізовану площину керування, тоді як другий має централізовану площину керування та центральний контролер. У розділах III-B та III-A ми детально розглядаємо відповідно TRILL та OpenFlow.

Другим за важливістю типом трафіку за кількістю є трафік від центру обробки даних до користувача. Очікується, що цей трафік зростає на 25% протягом наступних чотирьох років. Це показує як збільшення кількості клієнтів хмар, так і збільшення завдань, делегованих інфраструктурі хмари.

Останній тип трафіку, між центрами обробки даних, є найменш важливим за кількістю. Однак, він матиме найсильніше зростання, зі збільшенням на 31% протягом наступних чотирьох років. У цій статті ми зосереджуємося на цьому типі трафіку, який становить приблизно 7% від загального хмарного трафіку. Крім того, цей трафік між центрами обробки даних має інші обмеження, ніж два інших. Той факт, що він є міжцентровим, має бути прихований від користувачів хмар. У більшості випадків, коли відбувається зв'язок між центрами обробки даних, центри обробки даних належать одному

хмарному постачальнику. Тому здебільшого вони використовують одну й ту саму технологію. У цих випадках для рішення TRILL ми пропонуємо в [6] рішення, яке дозволяє взаємодіяти з кількома кампусами TRILL без їх об'єднання.

Однак у деяких випадках обидва центри обробки даних використовують різні технології, оскільки вони належать різним постачальникам. У [7] наведено декілька прикладів.

Перший – це хмарна федерація. Вона об'єднує хмарних постачальників, які спільно використовують свої ресурси за допомогою спільного регулювання для задоволення потреб своїх клієнтів.

Другий приклад – міжхмарне середовище. У цьому середовищі мережі кількох постачальників хмарних послуг взаємопов'язані, а їхні центри обробки даних використовують різні технології. У цих випадках трафік між центрами обробки даних має бути модифікований, щоб його приймав кожен центр обробки даних. Ця модифікація має бути прозорою для користувачів.

Третім прикладом може бути випадок, коли користувачі бажають скористатися перевагами продуктів різних постачальників хмарних послуг залежно від їхніх пропозицій та обмежень. Користувач може обрати одного постачальника хмарних послуг для зберігання величезної кількості даних, використовуючи іншого постачальника для його обчислювальної продуктивності, і, нарешті, обрати третього постачальника з високою пропускну здатністю для розміщення своїх веб-сервісів.

Четвертим прикладом може бути компанія, яка має кілька філій. Кожна філія підключена до найближчого центру обробки даних, щоб зменшити час доступу. Ми бачимо, що якщо немає міжхмарного доступу, то користувачі не зможуть досягти такого використання хмарних сервісів.

Спосіб взаємоз'єднання центрів обробки даних можна розділити на два підходи, кожен з яких має два рішення [7]. Якщо саме постачальник послуг хоче встановити взаємозв'язок центрів обробки даних, то це підхід, орієнтований на постачальника. В іншому випадку, коли клієнт хоче використовувати хмару кількох постачальників, це підхід, орієнтований на клієнта.

Серед підходів, орієнтованих на постачальника, є два рішення: федеративна хмара та гібридна хмара. У федеративному хмарному рішенні кілька постачальників хмарних послуг домовляються про правила та політики, що дозволяють взаємодію між своїми ресурсами.

Таким чином, вони можуть обмінюватися своїми невикористаними ресурсами один з одним. Клієнт хмари може навіть не знати про цей обмін ресурсами та використовувати інфраструктуру іншого постачальника.

Друге рішення – це гібридна хмара. У цьому рішенні організація з приватною хмарою, яка потребує більшої кількості ресурсів, використовуватиме хмару третьої сторони замість того, щоб розширювати власну інфраструктуру. Це рішення має економічну привабливість для клієнтів, оскільки таким чином їм не потрібно модифікувати свою приватну хмарну інфраструктуру.

Клієнтоорієнтований підхід містить два інших рішення: Мультихмарне рішення та Агрегований сервіс від брокера.

Мультихмарне рішення перекладає все управління кількома хмарними постачальниками та кількома технологіями на клієнта. У цьому випадку клієнту, можливо, доведеться розробити адаптер для АРІ кожного постачальника. Таким чином, постачальники не знають про своє хмарне з'єднання, оскільки саме клієнт керує ним.

Останнє рішення, Агрегований сервіс від брокера, схоже на попереднє, за винятком того, що клієнт не повинен керувати абстракцією АРІ. Натомість клієнти використовують послуги третьої сторони, яка називається брокером. Він надає єдиний АРІ для кількох хмарних постачальників, абстрагуючи кожен з АРІ постачальників.

У даній роботі ми зосереджуємося на підході, орієнтованому на постачальника, а точніше на федеративному хмарному рішенні. Ми розробили рішення для з'єднання кількох

центрів обробки даних за допомогою двох різних технологій. Це рішення є шлюзом, який з'єднує центри обробки даних за допомогою TRILL або OpenFlow.

Передумови OpenFlow

OpenFlow має централізовану площину керування, один центральний контролер та стандартні комутатори OpenFlow. Контролер OpenFlow має глобальне уявлення про топологію мережі та віртуальні мережі клієнтів. Контролер приймає кожне рішення щодо маршрутизації та кожне рішення щодо комутації всередині цієї мережі. Протокол OpenFlow здатний взаємодіяти з обладнанням, сумісним з OpenFlow, для заповнення їхньої таблиці комутації.

Ці таблиці комутації дозволяють застосовувати певний набір правил до кожного потоку. У разі відсутності таких правил для потоку, цей потік надсилається комутатором до контролера. Контролер є єдиним пристроєм, здатним вирішувати, які правила застосовувати до цього потоку.

Сумісні комутатори OpenFlow можуть бути як апаратними, так і програмними. Наприклад, Open vSwitch – це віртуальний комутатор, призначений для програмних гіпервізорів. Він пропонує мережеве підключення для віртуальних машин. Open vSwitch працює через таблицю потоків, яка визначає правила та дії для кожного потоку. Для ізоляції потоків кожного орендаря, комутатори OpenFlow можуть використовувати протоколи тунелювання, такі як VLAN або GRE. Однак, OpenvSwitch також може керувати протоколом бездержавного транспортного тунелювання (STT), успадкованим від розробки Nicira.

Архітектура OpenFlow містить принаймні один контролер, підключений до одного або кількох комутаторів OpenFlow. Контролер транслює всі свої правила на підключені комутатори через певний захищений канал. Ці комутатори зберігають усі ці правила у своїх таблицях комутації. Ці таблиці потім використовуються щоразу, коли комутатор отримує пакет, щоб визначити, як обробити пакет. Такий підхід має перевагу в централізації процесу прийняття рішень лише на одному логічному пристрої.

Логічна централізація процесу прийняття рішень дозволяє краще керувати кожним потоком від початку до кінця. Це також запобігає неузгодженості в обробці потоку та дозволяє швидко транслювати оновлення процесу потоку.

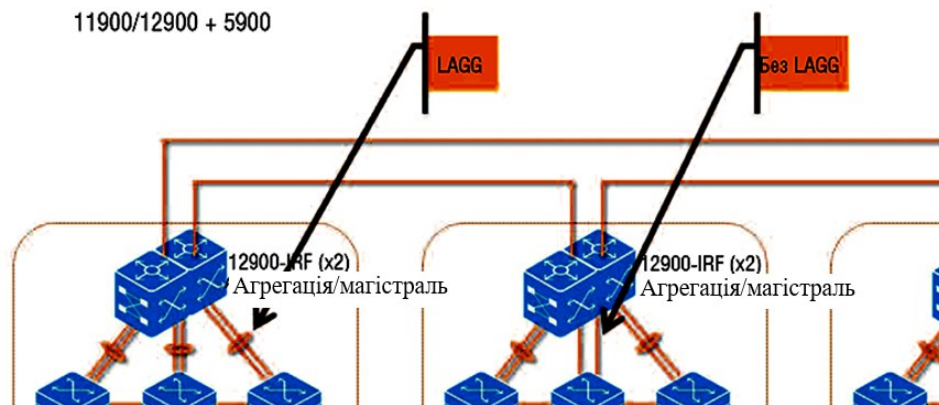


Рисунок 1 – Структурна схема системи

Передумови прозорого з'єднання багатьох каналів (TRILL)

На відміну від централізованої площини керування OpenFlow, TRILL [4] має децентралізовану площину керування. Це призводить до більшої стійкості TRILL, ніж OpenFlow. TRILL не має єдиної точки відмови, на відміну від Openflow та його контролера. TRILL – це протокол другого рівня з повною неблокуючою структурою, яка дозволяє будь-яке з'єднання в мережі. Новий компонент (RBridge) визначається специфікаціями TRILL для заміни традиційних комутаторів. Навіть якщо не всі комутатори потрібно замінити для роботи TRILL, вибір комутаторів для заміни є важливим. Фактично, RBridge діє як проксі, додаючи свою адресу до кадру.

Висновки. У статті наведені теоретичне узагальнення й рішення наукового завдання дослідження методів проектування та побудови топології мереж у центрах обробки даних. Рішення даного завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем проектування та побудови топології мереж у центрах обробки даних.
- Досліджена система проектування та побудови топології мереж у центрах обробки даних.
- На основі отриманих результатів досліджень створена програмна реалізація системи проектування та побудови топології мереж у центрах обробки даних.
- Розроблені алгоритми дозволяють успішно вирішувати завдання проектування та побудови топології мереж у центрах обробки даних. Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Список літератури

1. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.
2. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи*, 2023, том 7, № 2, С. 49-56.
3. Smirnova, T., Gnatyuk, S., Yudin, O., Sydorenko, V., Polozhentsev, A., «The Model for Calculating the Quantitative Criteria for Assessing the Security Level of Information and Telecommunication Systems». *CEUR Workshop Proceedings Volume 3156*, 2022, Pages 390-399.
4. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління*, № 2(70). 2022. С. 28-37.
5. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 3(69). С. 93-98.
6. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»*, № 2 (307). С. 46-52. 2022.
7. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.
8. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». *Communications in Computer and Information Science*, 2021, vol 1486. Springer, Cham. pp 169-184.
9. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings. Volume 2740*, 2020, Pages 102-114.
10. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.
11. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.
12. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) *Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies*, vol 48. Springer, Cham. 2021. pp 557-587.
13. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings Volume 2616*, 2020, Pages 125-136.
14. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic

- generation by a model of generator on the graph». CEUR Workshop Proceedings Volume 2616, 2020, Pages 366-379.
15. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.
 16. Smirnov O., Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.
 17. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.
 18. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.
 19. Smirnov, O., Odarchenko, R., Abakumova, A., Usik, P., Kundyz, M., «QoE optimization technique for media delivery in 5G networks». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019. P.597-601.
 20. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.
 21. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.
 22. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 353-358.
 23. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 347-352.
 24. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 618-629.
 25. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», Telecommunications and Radio Engineering. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.
 26. Смірнов О.А., Смірнова Т.В., Буравченко К.О., Кравченко С.С., Горбов В.О., «Хмарна система підтримки прийняття рішень технологічного процесу відновлення поверхонь конструкцій і деталей машин». Сучасні інформаційні системи. 2021. Т. 5, № 4. С. 79-95
 27. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у комп'ютерних системах операторів стільникового зв'язку» Вісник Черкаського державного технологічного університету. Технічні науки. №4. С. 103-110. 2020.
 28. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», Кібербезпека: освіта, наука, техніка. № 3(7). С. 43-62. 2020.
 29. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В., Поліщук Л.І. Інформаційна безпека в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2020. – 294 с.
 30. О.А. Смірнов, П.С. Усік, «дослідження перспектив використання технологічних рішень в мережах 5g» у Кібербезпека та інформаційні технології: монографія. – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.
 31. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». Центральнотрапнський науковий вісник. Технічні науки. № 2(33). с. 161-172, 2019.
 32. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В. Поліщук Л.І. Проектування комп'ютерних систем та мереж. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2019. – 264 с.