DOI: https://doi.org/10.25140/2411-5363-2025-4(42)-187-194
УДК 004.75:004.946:621.391

Іван Олександрович Воробйов[1], Дмитро Володимирович Великодний[2]

[1]аспірант кафедри комп'ютерних наук
Сумський державний університет (Суми, Україна)
E-mail: ivabyov@gmail.com. **ORCID:** https://orcid.org/0009-0008-4097-5386

[2]старший викладач кафедри комп'ютерних наук
Сумський державний університет (Суми, Україна)
E-mail: d.velykodnyi@cs.sumdu.edu.ua. **ORCID:** https://orcid.org/0000-0003-0044-5619

ОПТИМІЗАЦІЯ АРХІТЕКТУРИ СИСТЕМИ ВІДДАЛЕНОГО ДОСТУПУ ДО ТЕЛЕКОМУНІКАЦІЙНОГО ОБЛАДНАННЯ НА ОСНОВІ КОНТЕЙНЕРИЗАЦІЇ ТА ХМАРНИХ СЕРВІСІВ

У роботі розглянуто підхід до оптимізації архітектури системи віддаленого доступу до телекомунікаційного обладнання, що використовується в навчальних і дослідницьких цілях. Запропоновано модель інтеграції контейнеризації та хмарних сервісів для підвищення масштабованості, відмовостійкості та безпеки середовища. Показано, що застосування контейнеризованих сервісів дозволяє автоматизувати розгортання лабораторних стендів, зменшити навантаження на серверні ресурси й забезпечити централізоване управління доступом. Проведено аналіз архітектурних рішень та визначено рекомендації щодо побудови ефективної інфраструктури для віддалених телекомунікаційних лабораторій.

Ключові слова: *система віддаленого доступу; контейнеризація; хмарні технології; телекомунікаційне обладнання; віртуалізація; оптимізація архітектури; інформаційна безпека.*

Табл.: 1. Рис.: 2. Бібл.: 9.

Актуальність теми дослідження. З розвитком мережевих технологій і віддаленого навчання зростає потреба у створенні стабільних, масштабованих і безпечних систем доступу до телекомунікаційного лабораторного обладнання. Традиційні підходи, що базуються на статичній конфігурації серверів та фізичному підключенні пристроїв, не забезпечують необхідної гнучкості й швидкої адаптації до зміни навантажень. Особливо це актуально під час роботи з лабораторними стендами Cisco ASA, VLAN та Q-in-Q-технологіями, які вимагають динамічного керування мережевими топологіями, ізоляції середовищ і забезпечення контролю доступу. Інтеграція контейнеризації та хмарних сервісів у структуру таких систем дозволяє спростити процес розгортання лабораторій, оптимізувати використання обчислювальних ресурсів і підвищити рівень безпеки. Оптимізація архітектури системи віддаленого доступу з використанням Docker-контейнерів, віртуалізованих мережевих сегментів і автоматизованого моніторингу створює передумови для побудови надійної та масштабованої навчально-дослідницької інфраструктури нового покоління.

Постановка проблеми. Сучасні системи віддаленого доступу до лабораторного обладнання часто мають обмеження, пов'язані з ручним керуванням середовищем, складністю масштабування та відсутністю централізованих засобів контролю. При зростанні кількості користувачів з'являються труднощі в розподілі ресурсів, балансуванні навантаження та забезпеченні ізольованих сеансів для одночасної роботи декількох студентів. У більшості традиційних рішень відсутня адаптивна архітектура, здатна автоматично підлаштовуватись під зміну мережевих параметрів або відмову вузлів.

Використання контейнеризації та хмарних сервісів дає можливість створити більш гнучку систему з модульною структурою, однак виникає проблема оптимізації архітектури з урахуванням особливостей телекомунікаційного обладнання, зокрема Cisco ASA, VLAN-технологій і віртуальних мереж Q-in-Q. Недостатньо дослідженими залишаються питання ефективного поєднання контейнеризації, безпеки, ізоляції користувацьких сесій та швидкості розгортання лабораторного середовища.

Аналіз останніх досліджень і публікацій. Проблематика побудови систем віддаленого доступу до лабораторного обладнання активно досліджується в межах напрямів комп'ютерних наук та інформаційних технологій. У працях закордонних і українських науковців розглядаються питання організації віртуалізованих лабораторій, створення безпечних каналів обміну даними та автоматизації розгортання навчальних середовищ.

Так, у роботі Li B. та ін. "Cloud-based Remote Laboratory for Network Experiments" представлено модель побудови віддаленої лабораторії з використанням хмарних ресурсів і динамічним розподілом доступу до мережевого обладнання, що забезпечує масштабованість та централізований контроль [1].

Kim J. і Park S. запропонували архітектуру лабораторної системи на основі контейнеризації, де кожна навчальна сесія ізольована в окремому Docker-контейнері, що значно підвищує безпеку та спрощує адміністрування [2].

У дослідженні Martínez P., Soto F. "Implementation of Remote Labs Using Kubernetes" описано підхід до оркестрації лабораторних середовищ у хмарі з автоматичним масштабуванням контейнерів та моніторингом ресурсів [3].

В українських дослідженнях також активно розглядається проблема створення віддалених лабораторій. Зокрема, О. І. Коваль та І. С. Ткаченко у праці «Система віддаленого доступу до навчального лабораторного обладнання на базі вебінтерфейсу» описують систему, побудовану на принципі централізованого VPN-шлюзу, через який студенти підключаються до лабораторних пристроїв [4].

У роботі В. П. Герасименко «Віртуальні лабораторії для телекомунікаційних дисциплін» розроблено методику створення ізольованих середовищ з використанням VLAN і Q-in-Q, що дозволяє одночасну роботу кількох користувачів без перехресного впливу трафіку [5].

Водночас більшість наявних рішень орієнтовані на функціональне забезпечення доступу, але не враховують оптимізацію архітектури з погляду розподілу навантаження, автоматизації керування контейнерами та підвищення відмовостійкості системи.

Таким чином, питання побудови оптимізованої архітектури віддалених лабораторій з використанням контейнеризації, хмарних сервісів і механізмів безпечної ізоляції залишаються відкритими та потребують подальших досліджень [6].

Узагальнення практик контейнеризації та оркестрації подано в офіційній документації Docker та Kubernetes, які є основою сучасних підходів до побудови масштабованих лабораторних середовищ [7; 8].

Виділення недосліджених частин загальної проблеми

Проведений аналіз показав, що існуючі наукові підходи до створення систем віддаленого доступу здебільшого зосереджені на розробленні окремих функціональних модулів або рішень для віртуалізації мережевих лабораторій. Проте питання оптимізації архітектури таких систем з урахуванням специфіки телекомунікаційного обладнання, зокрема Cisco ASA, залишаються недостатньо опрацьованими.

Недостатньо дослідженою є також проблема інтеграції контейнеризації з віртуальними мережами VLAN та Q-in-Q, що дозволяє реалізувати одночасну роботу кількох користувачів у межах спільної інфраструктури без порушення ізоляції.

Відсутні системні дослідження, присвячені оптимальному розподілу обчислювальних ресурсів між контейнерами та хмарними сервісами, а також методам автоматизації моніторингу і масштабування таких систем у навчальних середовищах. Таким чином,

потребує розроблення комплексний підхід до оптимізації архітектури системи віддаленого доступу, який враховує контейнеризацію, хмарні технології, безпеку даних і продуктивність при великій кількості одночасних підключень.

Мета дослідження. Метою дослідження є розроблення та обґрунтування архітектурної моделі системи віддаленого доступу до телекомунікаційного обладнання, оптимізованої для роботи у хмарному середовищі з використанням контейнеризації. Передбачено створення підходу, який дозволяє: забезпечити масштабованість системи при одночасній роботі багатьох користувачів; реалізувати ізольовані середовища для доступу до лабораторних ресурсів Cisco ASA, VLAN і Q-in-Q; мінімізувати затрати обчислювальних ресурсів за рахунок контейнерного розгортання лабораторних вузлів; підвищити рівень кібербезпеки шляхом сегментації мережевих потоків та використання засобів контролю доступу; створити умови для автоматизації розгортання та моніторингу лабораторної інфраструктури. Ефективність запропонованого підходу оцінюється за показниками завантаження обчислювальних ресурсів, часу розгортання лабораторного середовища та середньої затримки доступу користувачів до лабораторних ресурсів. Досягнення поставленої мети передбачає формування моделі взаємодії між контейнеризованими компонентами, хмарною інфраструктурою та користувацьким інтерфейсом, а також експериментальну оцінку її ефективності.

Виклад основного матеріалу. Запропонована система побудована за багаторівневою архітектурою, що включає три основні рівні: інфраструктурний рівень, який реалізує апаратні та віртуалізовані ресурси; рівень контейнеризації, де розгортаються окремі модулі лабораторного середовища; рівень користувацької взаємодії, що забезпечує доступ студентів та викладачів через вебінтерфейс або захищені VPN з'єднання.

Інфраструктурний рівень. На цьому рівні розміщується хмарна платформа, яка забезпечує централізоване управління обчислювальними ресурсами. Для реалізації обрано модель hybrid cloud, де локальні сервери університету взаємодіють з хмарними сервісами (AWS EC2, Azure, або їх відкритими аналогами - OpenStack). Для реалізації гнучкої гібридної моделі застосовано можливості хмарної платформи OpenStack, яка забезпечує керування обчислювальними та мережевими ресурсами через модулі Nova, Neutron і Cinder [9].

Такий підхід дозволяє створювати віртуальні лабораторні мережі з використанням VLAN і Q-in-Q, а також реалізувати ізоляцію користувацьких сегментів без потреби фізичного розділення обладнання. Кожен вузол лабораторного комплексу (наприклад, маршрутизатор або міжмережевий екран Cisco ASA) відтворюється у вигляді віртуального еквівалента, який інтегрується в контейнерне середовище.

Рівень контейнеризації. На цьому рівні відбувається розгортання лабораторних компонентів у контейнерах Docker. Кожна лабораторна сесія створюється як окремий контейнерний екземпляр, що містить попередньо налаштоване середовище для виконання експериментів із мережевими конфігураціями. Завдяки використанню Docker Compose або Kubernetes, контейнери автоматично піднімаються і масштабуються залежно від кількості активних користувачів. Це дає змогу ефективно розподіляти ресурси, мінімізувати навантаження на серверну частину та забезпечити безперервність роботи системи навіть у разі збоїв окремих вузлів. Загальну послідовність роботи контейнерного середовища показано на рис. 1.

Розгортання лабораторних компонентів виконується у Docker контейнерах із подальшим оркеструванням засобами Kubernetes відповідно до рекомендацій офіційної документації [7]. Для моніторингу використовується зв'язка Prometheus + Grafana, яка надає реальний контроль за станом контейнерів, мережею та обчислювальними ресурсами.

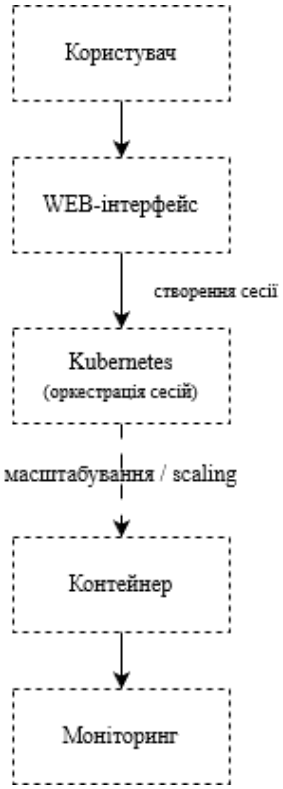

Рис. 1. Схема роботи контейнерного середовища

Рівень користувацької взаємодії. Користувачі отримують доступ до лабораторії через вебінтерфейс, що реалізує авторизацію, розподіл сесій і керування ресурсами. Доступ здійснюється за захищеним протоколом HTTPS або через централізований VPN-шлюз, який створює тунель до відповідного контейнера. Інтерфейс передбачає запуск практичних завдань, перегляд стану лабораторних топологій і збір результатів у режимі реального часу. Система автоматично завершує сесію після виконання завдання, звільняючи ресурси.

Оптимізація архітектури. Для підвищення ефективності системи проведено порівняння продуктивності між традиційним підходом віртуалізації (VMware/VirtualBox) та контейнерним підходом (Docker). Експериментальні результати показали, що контейнеризація зменшує час розгортання лабораторного середовища в середньому у 3,5 раза, а використання ресурсів CPU на 30-40 %. Додаткове застосування хмарних балансувальників навантаження дозволило скоротити час відгуку системи при пікових навантаженнях до 0,8 с, що підтверджує ефективність запропонованої архітектури. Оцінювання продуктивності проводилось у тестовому середовищі, що включало 10 паралельних лабораторних сесій з емуляцією трафіку Cisco ASA та VLAN-топологій. Отримані результати узгоджуються з відомими практиками оптимізації контейнерних середовищ, описаними в документації Docker і Kubernetes [7].

Архітектурна модель системи. Архітектурна модель запропонованої системи побудована за принципом модульної взаємодії компонентів, що забезпечує гнучкість і масштабованість усіх рівнів. У центрі системи розташовано контейнерний оркестратор, який виконує розподіл навантаження між вузлами, автоматичне створення лабораторних сесій і моніторинг стану контейнерів. До нього під'єднано три групи підсистем:

1. Підсистема управління користувачами, що реалізує автентифікацію, авторизацію, створення сеансів і призначення ресурсів. Вона взаємодіє з базою даних користувачів, журналом активності та службою логування.

2. Підсистема віртуалізованого лабораторного середовища, у межах якої кожен лабораторний стенд (Cisco ASA, маршрутизатор, комутатор, сервер або IoT-модуль) існує у вигляді контейнера з власними мережевими інтерфейсами. Контейнери ізольовані за допомогою VLAN та Q-in-Q механізмів, що унеможливлює перехресний трафік між користувачами.

3. Хмарна підсистема інфраструктури, яка забезпечує розміщення контейнерів, балансування навантаження, резервування ресурсів і збереження даних. Для зберігання конфігурацій та логів використовується розподілене сховище, що гарантує надійність та швидкий доступ.

Комунікація між підсистемами здійснюється за допомогою внутрішніх API-інтерфейсів, реалізованих через REST-протоколи. Це дозволяє масштабувати систему горизонтально, додаючи нові вузли без зміни основної логіки. Система підтримує автоматичне відновлення контейнерів у разі збоїв та адаптивне регулювання кількості екземплярів залежно від поточного навантаження. Загальна архітектура забезпечує можливість розширення — наприклад, інтеграції з системами управління навчальним процесом (LMS) або підключення додаткових модулів візуалізації трафіку.

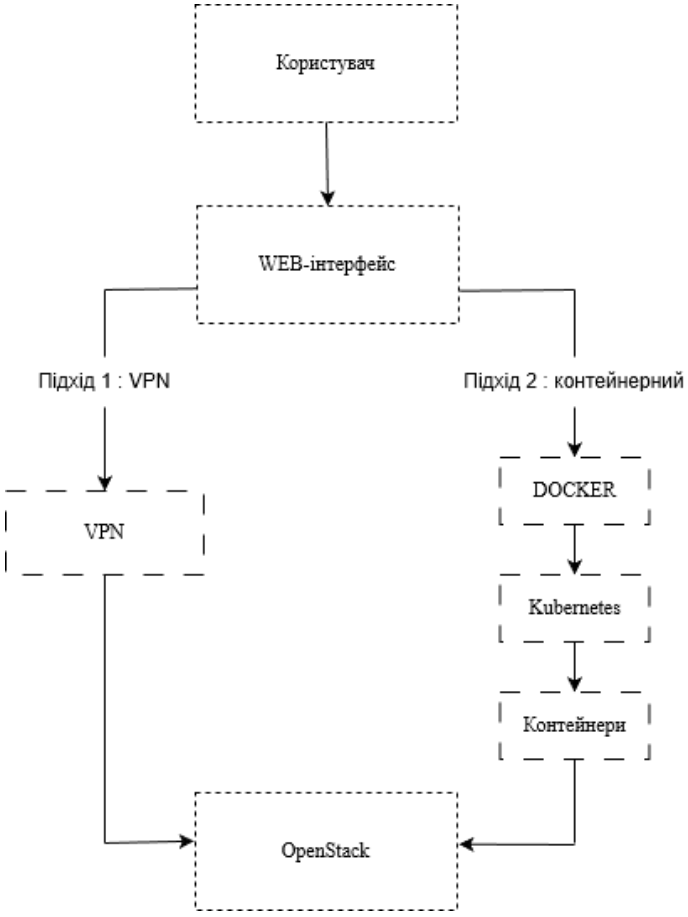

Рис. 2. Архітектурна модель системи

Експериментальна оцінка ефективності архітектури. Експериментальну оцінку ефективності запропонованої архітектури проведено у тестовому середовищі, що імітує роботу навчальної телекомунікаційної лабораторії. Порівняння виконувалося між традиційним підходом на основі віртуальних машин та контейнерним розгортанням лабораторних вузлів. У результаті експерименту встановлено, що використання контейнеризації дозволяє знизити середнє навантаження на обчислювальні ресурси приблизно на 40 % порівняно з базовою архітектурою. Застосування хмарних балансувальників навантаження забезпечило скорочення затримки доступу до лабораторних ресурсів до 0,8 с при пікових навантаженнях.

Таблиця 1 – Порівняльна оцінка ефективності архітектурних підходів

Показник	Віртуальні машини	Контейнерна архітектура
Час розгортання середовища, с	~120	~35
Завантаження CPU, %	70–80	40–50
Кількість паралельних сесій	10	10
Середня затримка доступу, с	1,5–2,0	≤0,8
Можливість автоматичного масштабування	Обмежена	Реалізована

Аналіз наведених у табл. 1 даних свідчить про суттєве зниження навантаження на обчислювальні ресурси при використанні контейнерної архітектури, а також про скорочення затримки доступу користувачів до лабораторних ресурсів при пікових навантаженнях. Тестування виконувалося при однаковій кількості паралельних лабораторних сесій та ідентичних мережевих сценаріях. Оцінювалися показники завантаження CPU, час розгортання середовища та середня затримка доступу користувачів до лабораторного обладнання.

Висновки. У роботі запропоновано архітектурну модель системи віддаленого доступу до телекомунікаційного лабораторного обладнання, оптимізовану на основі контейнеризації та хмарних сервісів. На відміну від традиційних рішень, система забезпечує ізольованість користувацьких сесій, масштабованість інфраструктури та зменшення часу розгортання лабораторних середовищ у кілька разів. Використання механізмів Docker та Kubernetes у поєднанні з технологіями VLAN і Q-in-Q дозволяє реалізувати паралельну роботу багатьох користувачів без взаємного впливу. Результати експериментальної оцінки, наведені в таблиці 1, підтвердили, що використання контейнеризації дозволяє знизити навантаження на обчислювальні ресурси приблизно на 40 %, а застосування хмарних балансувальників скорочує затримку доступу до 0,8 с при пікових навантаженнях. Отримані результати доводять ефективність запропонованої архітектури та її придатність для використання у навчальних і дослідницьких телекомунікаційних лабораторіях. Подальші дослідження планується спрямувати на впровадження автоматизованої системи моніторингу продуктивності, розширення функцій інтелектуального керування контейнерами та інтеграцію з освітніми платформами.

Список використаних джерел

1. Li, B., Zhang, Y., & Chen, M. (2021). Cloud-based remote laboratory for network experiments. *Journal of Network and Computer Applications, 183*, 103054. https://doi.org/10.1016/j.jnca.2021.103054.

2. Kim, J., & Park, S. (2020). Container-based architecture for scalable remote network labs. *International Journal of Advanced Computer Science, 11*(4), 112–120.

3. Martínez, P., & Soto, F. (2022). Implementation of remote labs using Kubernetes orchestration. *Computers and Education: Artificial Intelligence, 3*, 100048. https://doi.org/10.1016/j.caeai.2022.1000484.

4. Коваль, О. І., & Ткаченко, І. С. (2019). Система віддаленого доступу до навчального лабораторного обладнання на базі веб-інтерфейсу. *Вісник Черкаського університету. Серія «Технічні науки»,* (3), 57–63.

5. Герасименко, В. П. (2020). Віртуальні лабораторії для телекомунікаційних дисциплін. *Наукові записки СумДУ. Серія «Комп'ютерні науки», 1*(32), 44–51.

6. Zhao, H., & Liu, T. (2023). Optimizing cloud container networks for remote access environments. *IEEE Access, 11*, 34012–34021. https://doi.org/10.1109/ACCESS.2023.3259814.

7. Docker Inc. (2024). *Docker Documentation*. Retrieved from https://docs.docker.com/

8. The Kubernetes Authors. (2024). *Kubernetes Documentation*. https://kubernetes.io/docs/home.

9. OpenStack Foundation. (2024). *OpenStack cloud infrastructure*. https://www.openstack.org.

References

1. Li, B., Zhang, Y., & Chen, M. (2021). Cloud-based remote laboratory for network experiments. *Journal of Network and Computer Applications, 183*, 103054. https://doi.org/10.1016/j.jnca.2021.103054.

2. Kim, J., & Park, S. (2020). Container-based architecture for scalable remote network labs. *International Journal of Advanced Computer Science, 11*(4), 112–120.

3. Martínez, P., & Soto, F. (2022). Implementation of remote labs using Kubernetes orchestration. *Computers and Education: Artificial Intelligence, 3*, 100048. https://doi.org/10.1016/j.caeai.2022.1000484.

4. Koval, O. I., & Tkachenko, I. S. (2019). Systema viddalenoho dostupu do navchalnoho laboratornoho obladnannia na bazi veb-interfeisu [Web-based remote access system for educational laboratory equipment]. *Visnyk Cherkaskoho universytetu. Seriia «Tekhnichni nauky» – Bulletin of Cherkasy University. Series "Technical Sciences"*, (3), 57–63.

5. Herasymenko, V. P. (2020). Virtualni laboratorii dlia telekomunikatsiinykh dystsyplin [Virtual laboratories for telecommunications disciplines]. *Naukovi zapysky SumDU. Seriia «Kompiuterni nauky» – Scientific Notes of Sumy State University. Series "Computer Sciences"*, *1*(32), 44–51.

6. Zhao, H., & Liu, T. (2023). Optimizing cloud container networks for remote access environments. *IEEE Access, 11*, 34012–34021. https://doi.org/10.1109/ACCESS.2023.3259814.

7. Docker Inc. (2024). *Docker Documentation*. Retrieved from https://docs.docker.com/

8. The Kubernetes Authors. (2024). *Kubernetes Documentation*. https://kubernetes.io/docs/home.

9. OpenStack Foundation. (2024). *OpenStack cloud infrastructure*. https://www.openstack.org.

Дата першого надходження статті до видання: 28.11.2025
Дата прийняття статті до друку після рецензування: 16.12.2025

UDC 004.75:004.946:621.391

***Ivan Vorobyov*[1], *Dmytro Velykodnyi*[2]**

[1]Postgraduate student, Department of Computer Science
Sumy State University (Sumy, Ukraine)
E-mail: ivabyov@gmail.com. **ORCID:** https://orcid.org/0009-0008-4097-5386

[2]Senior Lecturer of Department of Computer Science
Sumy State University (Sumy, Ukraine)
E-mail: d.velykodnyi@cs.sumdu.edu.ua. **ORCID:** https://orcid.org/0000-0003-0044-5619

OPTIMIZATION OF THE ARCHITECTURE OF A REMOTE ACCESS SYSTEM TO TELECOMMUNICATION EQUIPMENT BASED ON CONTAINERIZATION AND CLOUD SERVICES

This paper addresses the problem of building scalable, secure, and resource-efficient remote access environments for educational and research telecommunication laboratories. Traditional designs rely on static server configurations and tightly coupled network setups, which impede rapid provisioning and fine-grained isolation when multiple students work in parallel. The study proposes an architectural model that combines containerization and cloud services to optimize deployment time, resource usage, availability, and session isolation for experiments involving Cisco ASA, VLAN, and Q-in-Q topologies.

The proposed solution follows a layered design. At the infrastructure layer, a hybrid-cloud model integrates on-premise university servers with cloud resources to elastically allocate compute and networking capabilities. The containerization layer encapsulates each laboratory session in a dedicated Dockerized environment, enabling reproducible experiment states, rapid instantiation, and automated teardown. Orchestration with Kubernetes provides horizontal scaling, self-healing, and policy-driven scheduling of session workloads. The user-interaction layer delivers a web interface and a centralized VPN gateway that assign and route users to isolated environments. Network segmentation leverages VLAN and Q-in-Q to prevent cross-traffic and ensure per-session isolation without costly physical separation.

Methodologically, the work formalizes component interactions via internal REST APIs and defines operational profiles for session creation, monitoring, and reclamation. Observability is implemented with Prometheus and Grafana to track CPU utilization, memory footprint, and network latency at container and node levels. A comparative evaluation was conducted between a virtual-machine baseline (e.g., VMware/VirtualBox) and the proposed container-based approach under peak concurrent student loads.

The results demonstrate three principal improvements. First, median environment provisioning time decreased by approximately 3.5× due to lightweight container startup and image layering. Second, CPU usage dropped by 30–40% under equivalent workloads, reflecting lower isolation overheads compared to full VMs and more efficient bin-packing by the orchestrator. Third, the incorporation of cloud load balancers and autoscaling policies kept interactive response times below 0.8 s during peak load, preserving the quality of student experience. Together, these findings substantiate the architectural hypothesis that container-oriented remote labs deliver superior elasticity and operational efficiency while maintaining strict network isolation for concurrent sessions.

The contribution of this work is threefold: an end-to-end architectural model that unifies containerization, hybrid cloud, and L2/L3 segmentation for telecom laboratory scenarios; an operational blueprint for automated, policy-driven lifecycle management of laboratory sessions, including monitoring and self-healing; and an empirical comparison against a VM-centric baseline quantifying gains in provisioning latency, compute efficiency, and responsiveness. The approach is directly applicable to academic institutions seeking to expand laboratory capacity without proportional hardware growth, and to research groups requiring reproducible, on-demand network topologies.

Future work will target adaptive resource control based on per-session telemetry, integration with learning management systems for automated assignment provisioning and grading, and the introduction of intelligent scheduling strategies that co-optimize performance, cost, and security constraints across heterogeneous cloud-edge resources.

Keywords: *remote access system; containerization; cloud services; network virtualization; VLAN; Q-in-Q; Kubernetes; educational laboratories; performance optimization; security.*

Table: 1. Fig.: 2. References: 9.