



ДОСЛІДЖЕННЯ УПРАВЛІННЯ НАДІЙНІСТЮ ТА ВІДМОВСТІЙКІСТЮ В ІНФОКОМУНІКАЦІЙНИХ МЕРЕЖАХ: МОДЕЛЮВАННЯ ТА ТЕСТУВАННЯ ПРОТОКОЛІВ РЕЗЕРВУВАННЯ ШЛЮЗІВ ЗА ЗАМОВЧУВАННЯМ

О. Єременко^[ORCID: 0000-0003-3721-8188], Р. Савченко^[ORCID: 0009-0005-6693-8733], К. Яковенко^[ORCID: 0009-0003-1980-6063],
С. Шестопапов^[ORCID: 0009-0008-8643-0752]

Харківський національний університет радіоелектроніки, пр. Науки, 14, Харків, 61166, Україна

Відповідальна за рукопис: О. Єременко (e-mail: oleksandra.yeremenko@nure.ua).

(Подано 28 лютого 2025)

Стаття присвячена дослідженню механізмів управління надійністю та відмовостійкістю в інфокомунікаційних мережах, зосереджуючись на моделюванні та тестуванні протоколів резервування шлюзів. У роботі розглянуто протоколи VRRP (Virtual Router Redundancy Protocol) та GLBP (Gateway Load Balancing Protocol), які забезпечують безперервну роботу мережі у випадку відмови основного шлюзу. Метою дослідження є порівняльний аналіз цих протоколів з використанням віртуальних середовищ EVE-NG та Containerlab, а також їх тестування на фізичних та віртуальних пристроях. У статті виконано аналіз характеристик і механізмів роботи обох протоколів, програмну реалізацію та конфігурацію протоколів на пристроях Cisco у середовищах емуляції, а також проведено тестування ефективності в умовах відмов шлюзів. Результати показали, що VRRP забезпечує мінімальні втрати пакетів та швидке переключення між активним і резервним шлюзами, що робить його оптимальним для мереж з високими вимогами до безперервності з'єднання. Натомість протокол GLBP забезпечує балансування навантаження між шлюзами, але супроводжується більшими втратами пакетів та довшим часом переключення, що обмежує його застосування в критичних мережах. Зокрема, тестування у віртуальних середовищах продемонструвало, що при використанні віртуалізованих каналів і ресурсів затримки та втрати пакетів у VRRP значно менші порівняно з GLBP. Отже, тестування у віртуальних середовищах є корисним для попереднього аналізу конфігурацій, але для точного визначення ефективності протоколів необхідне тестування на реальних пристроях. Також розглядаються можливості використання GLBP у великих мережах для забезпечення не лише резервування, а й оптимізації балансування трафіку. На основі отриманих результатів сформульовано пропозиції щодо перспектив подальших досліджень, які включають оптимізацію часу перемикання між шлюзами та розробку гібридних рішень комбінованого використання VRRP та GLBP, що поєднують переваги обох протоколів для підвищення ефективності управління трафіком та резервуванням і створення більш гнучких і стійких до відмов інфраструктур.

Ключові слова: надійність, відмовостійкість, шлюз за замовчуванням, VRRP, GLBP, моделювання, тестування

УДК: 621.391

Вступ

Надійність і відмовостійкість у сучасних інфокомунікаційних мережах є надзвичайно важливими аспектами, які визначають якість і безперервність надання послуг [1–3]. Зростання кількості користувачів, обсягів оброблюваних даних і складність мережних структур призводять до

збільшення кількості викликів у забезпеченні стабільної роботи мереж. В умовах, коли збоїв в роботі мережних пристроїв чи каналів передачі даних можуть призвести до фінансових втрат, порушення бізнес-процесів або навіть до серйозних наслідків для безпеки, гарантування високого рівня надійності та відмовостійкості стає ключовою задачею для мережних інженерів. Важливо відзначити, що надійність мережі безпосередньо впливає на довіру користувачів та загальну репутацію компанії, підприємства чи провайдера, адже навіть короткочасна недоступність сервісів може мати негативні наслідки.

Однією з головних задач забезпечення надійності та відмовостійкості є мінімізація простоїв та гарантування безперервності бізнес-процесів. Це особливо актуально для критично важливих систем, як-от банківські сервіси, системи управління виробництвом, медичні установи або служби екстреної допомоги тощо. Крім того, надійність мережі сприяє підтримці високого рівня безпеки, оскільки незаплановані відключення можуть створити вразливості для атак або втрати даних. Враховуючи щораз більшу залежність від цифрових технологій, забезпечення безперебійної роботи інфраструктури набуває стратегічного значення для успішного функціонування підприємств та організацій.

Різноманітні приклади реалізації забезпечення надійності та відмовостійкості в мережах включають як апаратні, так і програмні підходи [1–3]. Серед апаратних рішень можна виділити системи дублювання компонентів, які автоматично переключаються у випадку відмови основного елемента. Це можуть бути резервні сервери, додаткові лінії зв'язку, мережні адаптери та інші критичні компоненти. Такі рішення дозволяють зменшити залежність від одного вузла та зберегти цілісність роботи мережі навіть при виникненні апаратних збоїв [2]. Також широко застосовуються програмні технології, такі як кластеризація серверів, реплікація баз даних, віртуалізація ресурсів та балансування навантаження, що дозволяють адаптуватися до змін у навантаженні та швидко відновлювати роботу після збоїв.

Особливе місце в реалізації мережної відмовостійкості займають протоколи резервування шлюзів за замовчуванням (або захисту шлюзів), які забезпечують резервування маршрутизаторів та безперервну роботу у випадку відмови окремих вузлів [1]. За допомогою протоколів сімейства First-Hop Redundancy Protocols (FHRPs) створюється віртуальний шлюз, через який проходить увесь трафік, що дозволяє в разі збоїв одного з маршрутизаторів автоматично переключатися на інший, мінімізуючи втрату пакетів та переривання з'єднання. Серед найбільш відомих протоколів такого типу можна виділити Hot Standby Router Protocol (HSRP), Gateway Load Balancing Protocol (GLBP) та Virtual Router Redundancy Protocol (VRRP), кожен з яких має свої переваги та особливості реалізації [4–13]. Наприклад, HSRP та GLBP є пропрієтарними рішеннями компанії Cisco [4, 5], тоді як VRRP є відкритим стандартом [6], що забезпечує більшу сумісність у різноманітних мережах.

Відмовостійкість мережного обладнання не обмежується лише протоколами захисту шлюзів. Багато сучасних комутаторів та маршрутизаторів підтримують різні техніки підвищення стійкості, такі як агрегування каналів (Port Channel, EtherChannel), резервування електроживлення, використання резервних копій програмного забезпечення та інтеграцію з системами моніторингу, які дозволяють передбачати й запобігати можливим відмовам [3]. Резервування каналів дозволяє об'єднати декілька фізичних з'єднань в єдиний логічний, що не лише збільшує пропускну здатність, але й створює запаси на випадок відмови окремих ліній. Системи моніторингу та аналізу дозволяють адміністратору оперативно реагувати на проблеми, що виникають у мережі, забезпечуючи швидке відновлення роботи навіть за складних сценаріїв збоїв.

Ключовим аспектом реалізації надійності є також балансування навантаження, яке сприяє рівномірному розподілу трафіку між кількома вузлами мережі [1–3, 8, 9]. Це дозволяє не тільки покращити продуктивність системи, але й зменшити ризик перевантаження окремих компонентів, що зі свого боку підвищує загальну стійкість мережі. Балансування навантаження може здійснюватися на різних рівнях: від балансування на рівні фізичних ліній зв'язку до розподілу запитів на рівні застосунків. Використання таких підходів дозволяє будувати мережі, здатні

адаптуватися до змін трафіку та зберігати стабільність навіть у випадку несподіваних навантажень або відмов окремих елементів інфраструктури.

У контексті захисту шлюзу відмовостійкість реалізується через спеціалізовані протоколи, які дозволяють пристроям автоматично виявляти відмови та переключати трафік на резервні канали чи вузли. Протоколи захисту шлюзу забезпечують створення віртуальних адрес, якими користуються кінцеві пристрої, водночас фактично трафік маршрутизується через робочі фізичні шлюзи [3–7]. Це сприяє тому, що навіть у разі відмови одного з маршрутизаторів мережа продовжує функціонувати без переривань, адже інший маршрутизатор автоматично бере на себе роль обробника трафіку. Такий підхід не тільки підвищує надійність, але й спрощує адміністрування мережі, оскільки кінцеві пристрої не потребують переналаштування своїх параметрів після виникнення проблем зі шлюзом.

Отже, метою цієї статті є дослідження механізмів управління надійністю та відмовостійкістю в інфокомунікаційних мережах шляхом моделювання та тестування протоколів захисту шлюзу. Основна увага приділяється порівняльному аналізу протоколів VRRP (Virtual Router Redundancy Protocol) та GLBP (Gateway Load Balancing Protocol) у різних середовищах емуляції та їх тестуванню на фізичних і віртуальних пристроях.

Для досягнення поставленої мети у статті вирішуються такі завдання:

- порівняння характеристик і механізмів роботи протоколів VRRP та GLBP;
- програмна реалізація та конфігурація VRRP та GLBP на пристроях Cisco у середовищах EVE-NG і Containerlab;
- тестування відмовостійкості VRRP та GLBP на фізичних і віртуальних пристроях;
- обговорення отриманих результатів та формулювання висновків щодо ефективності досліджених підходів.

2. Порівняння характеристик і механізмів роботи протоколів VRRP та GLBP

Розглядаючи різноманітні реалізації механізмів захисту та резервування, можна навести приклади використання таких протоколів, як HSRP, VRRP та GLBP [3–13]. Кожен із цих протоколів вирізняється підходами до реалізації відмовостійкості та резервування шлюзу. HSRP, розроблений компанією Cisco, створює активний і резервний маршрутизатор, які працюють у парі, забезпечуючи безперервність зв'язку [4]. VRRP, як відкритий стандарт, дозволяє налаштовувати кілька резервних маршрутизаторів у гетерогенних мережах, де присутнє обладнання від різних виробників [6]. GLBP надає додаткову можливість балансування навантаження між кількома вузлами, що підвищує ефективність використання ресурсів мережі [5].

Виконуючи порівняння протоколів відмовостійкості, важливо враховувати, що вибір конкретного рішення часто залежить від вимог мережі, доступного обладнання та сценаріїв використання [10–12]. Кожен із протоколів має свої переваги й недоліки. Наприклад, HSRP простий у налаштуванні та добре працює в однорідних мережах із обладнанням Cisco, однак його пропріетарна природа може обмежити застосування в мультивендорних середовищах. З іншого боку, VRRP, як відкритий стандарт, забезпечує гнучкість та сумісність із різними виробниками, що робить його більш універсальним вибором, особливо в умовах наявності гетерогенного парку обладнання. GLBP додає до цього можливості балансування навантаження, що може бути корисним у великих мережах із великим обсягом трафіку, де критично важливо оптимізувати використання мережевих ресурсів.

Ретельний аналіз протоколів HSRP, VRRP та GLBP (таблиця 1) дозволяє обрати придатний метод захисту шлюзу залежно від конкретних умов експлуатації мережі. Під час вибору слід оцінити не лише технічні характеристики кожного протоколу, але й їхню сумісність з наявною інфраструктурою, можливості масштабування, а також вимоги до балансування навантаження.

Такий підхід дозволяє побудувати стійку та гнучку мережну архітектуру, здатну ефективно реагувати на збої та забезпечувати безперервність роботи навіть у найскладніших умовах.

Отже, HSRP та GLBP забезпечують резервування шлюзу в мережах з обладнанням Cisco. Механізм HSRP базується на концепціях Active- і Standby-маршрутизаторів: основний вузол обробляє трафік, а резервний очікує відмови першого. Через закритість стандарту використання HSRP в мультивендорних середовищах ускладнене [3, 4]. GLBP, на відміну від HSRP, підтримує балансування навантаження між кількома шлюзами. Усі пристрої в групі можуть брати участь у маршрутизації, що підвищує ефективність використання ресурсів. Проте, GLBP залишається пропрієтарним рішенням, що обмежує його застосування у мережах з обладнанням різних виробників [5]. VRRP – відкритий стандарт, підтримуваний багатьма вендорами, що робить його оптимальним вибором для гетерогенних мереж. Його принцип роботи схожий на HSRP: головний маршрутизатор (Master) обробляє трафік, а резервні (Backup) готові його замінити. Завдяки відкритості VRRP забезпечує гнучкість та сумісність між обладнанням різних виробників, що робить його найкращим рішенням для мереж зі змішаною інфраструктурою [3, 6]. Вибір між HSRP, GLBP та VRRP залежить від особливостей мережі та доступного обладнання. HSRP і GLBP ефективні в інфраструктурі Cisco, водночас VRRP є універсальним рішенням для мереж з пристроями від різних вендорів. Порівняння ключових характеристик цих протоколів наведено в таблиці 1 [3–13].

Таблиця 1

Порівняння протоколів HSRP, GLBP та VRRP

Характеристика	HSRP	GLBP	VRRP
Стандарт	Пропрієтарний (Cisco)	Пропрієтарний (Cisco)	Відкритий (IETF)
Розподіл навантаження	Немає	Є, балансування трафіку між вузлами	Немає
Типи ролей	Active, Standby	Active Virtual Gateway, Active Virtual Forwarder	Master, Backup
Сумісність із різними вендорами	Тільки Cisco	Тільки Cisco	Широка (різні вендори)
Можливість пріоритетів	Наявна	Наявна	Наявна
Базова ідея	Резервування основного шлюзу	Резервування з балансуванням	Резервування з відкритим стандартом
Сценарії використання	Однорідні Cisco-мережі	Однорідні Cisco-мережі з необхідністю розподілу навантаження	Гетерогенні середовища, мультивендор
Швидкість переключення	Висока	Висока	Висока

Підсумовуючи порівняння протоколів відмовостійкості, можна зробити висновок, що VRRP та GLBP є найбільш доцільними для подальшого дослідження, оскільки вони забезпечують не лише резервування шлюзу, а й ефективний розподіл навантаження (у випадку GLBP). VRRP, як відкритий стандарт, є універсальним рішенням для гетерогенних мереж, тоді як GLBP надає додаткові можливості в інфраструктурі Cisco. Реалізація та тестування цих протоколів дозволяє оцінити їхню ефективність у різних сценаріях, що є критично важливим для забезпечення безперервності роботи мережі.

Для практичного дослідження обрано дві платформи моделювання: EVE-NG та Containerlab [14, 15]. EVE-NG є популярним середовищем для емуляції мережного обладнання, що дозволяє створювати реалістичні тестові конфігурації на віртуальних пристроях [14]. Водночас Containerlab забезпечує більш гнучкий та контейнеризований підхід до розгортання мережних сценаріїв, що є важливим у сучасних хмарних середовищах [15]. Використання обох платформ дозволяє оцінити VRRP та GLBP у різних умовах, зокрема на фізичних і віртуальних пристроях, що дає змогу отримати комплексну картину роботи цих протоколів у реальних мережах.

3. Програмна реалізація VRRP та GLBP на пристроях Cisco в середовищі EVE-NG

Конфігурація VRRP

Для експериментальної перевірки механізмів відмовостійкості було вирішено налаштувати VRRP у підмережі 192.168.88.0/24. У ній передбачено два маршрутизатори Cisco (Router1 і Router2), комутатор (Switch) та клієнтський ПК (Client). Router1 отримав IP-адресу 192.168.88.253, а Router2 – 192.168.88.252. Для користувачів цієї мережі віртуальним шлюзом став 192.168.88.254, який налаштовано за допомогою VRRP. Усі пристрої об'єднані в один VLAN 100 на комутаторі, а ПК, зокрема, має адресу 192.168.88.100 із маскою 255.255.255.0 та вказаним шлюзом 192.168.88.254. Топологія мережі для дослідження продемонстрована на рисунку 1.

Основна ідея протоколу VRRP полягає в тому, що Master-маршрутизатор обробляє весь трафік до віртуальної IP-адреси, доки залишається доступним у мережі. У конфігурації використано групу VRRP з ідентифікатором 1. Пріоритет Router1 встановлено вищим, наприклад 120, що дає йому право бути основним маршрутизатором. На Router2 задається нижчий пріоритет (наприклад 100), і він відповідає за резервування. Параметр preempt дає змогу маршрутизатору з вищим пріоритетом автоматично відновлювати роль Master, якщо він повернувся до роботи після збою.

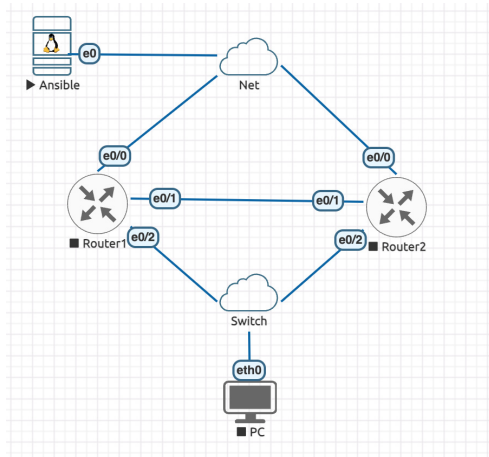


Рис. 1. Топологія мережі для дослідження

Під час налаштування вручну кожен маршрутизатор отримує команди для визначення IP-адреси на інтерфейсі, увімкнення VRRP та встановлення пріоритету. Однак для прискорення цього процесу та запобігання можливим помилкам доцільно застосовувати Ansible [16]. У межах дослідження було підготовлено playbook (збірку завдань), що описує базову адресацію, активує VRRP та налаштовує пріоритети на Router1 і Router2. Запуск playbook запропоновано виконувати за допомогою команди *ansible-playbook -i hosts cisco.yaml* (рис 2).

```
[root@ansible ansible]# ansible-playbook -i hosts cisco.yaml

PLAY [Configure VRRP on Cisco routers] *****

TASK [Створити VLANи на пристрої] *****
[WARNING]: To ensure idempotency and correct diff the input configuration lines should be similar to how they appear if present in the running configuration on device
changed: [Router1]
changed: [Router2]

TASK [Router1: Налаштування фізичних інтерфейсів] *****
changed: [Router1]

TASK [Router1: Налаштування фізичних інтерфейсів] *****
changed: [Router2]

PLAY RECAP *****
Router1      : ok=2    changed=2    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
Router2      : ok=2    changed=2    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

Рис. 2. Лог виконання *playbook cisco.yaml*

Результатом виконання *playbook* стало налаштування протоколу VRRP на маршрутизаторах Cisco, що дозволило створити відмовостійкий шлюз для підмережі 192.168.88.0/24. Успішне налаштування було підтверджено через аналіз логів виконання команд Ansible, де кожен крок був відображений із зазначенням статусу виконання.

Конфігурація маршрутизаторів, відображена на рисунках 3 та 4, демонструє успішне виконання налаштувань після запуску *playbook*. Усі основні параметри, включно з VRRP, адресацією інтерфейсів та активними сервісами, були застосовані відповідно до очікуваних вимог.

```
ip dhcp pool VLAN100
 network 192.168.88.0 255.255.255.0
 default-router 192.168.88.254
 dns-server 1.1.1.1
ip domain-name ownhub.network
no ip cef
no ipv6 cef
spanning-tree mode pvst
spanning-tree extend system-id
interface Ethernet0/0
 description Uplink_to_main_router
 switchport mode access
interface Ethernet0/1
 switchport access vlan 100
 switchport mode access
interface Ethernet0/2
 switchport access vlan 100
 switchport mode access
interface Ethernet0/3
 switchport access vlan 100
 switchport mode access
interface Vlan1
 ip address 10.10.150.10 255.255.255.0
interface Vlan100
 ip address 192.168.88.253 255.255.255.0
 vrrp 1 ip 192.168.88.254
 vrrp 1 priority 120
ip forward-protocol nd
ip http server
ip route 0.0.0.0 0.0.0.0 10.10.150.254
ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
control-plane
line con 0
 logging synchronous
line aux 0
line vty 0 4
 login local
 transport input ssh
end
```

Router1#

Рис. 3. Конфігурація Router1 після виконання *playbook cisco.yaml*

```
ip dhcp pool VLAN100
 network 192.168.88.0 255.255.255.0
 default-router 192.168.88.254
 dns-server 1.1.1.1
ip domain-name ownhub.network
ip cef
no ipv6 cef
spanning-tree mode pvst
spanning-tree extend system-id
interface Ethernet0/0
 description *** Uplink to main router ***
 switchport mode access
interface Ethernet0/1
 switchport access vlan 100
 switchport mode access
interface Ethernet0/2
 switchport access vlan 100
 switchport mode access
interface Ethernet0/3
 switchport access vlan 100
 switchport mode access
interface Vlan1
 ip address 10.10.150.11 255.255.255.0
interface Vlan100
 ip address 192.168.88.252 255.255.255.0
 vrrp 1 ip 192.168.88.254
ip forward-protocol nd
ip http server
ip route 0.0.0.0 0.0.0.0 10.10.150.254
ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
control-plane
line con 0
 logging synchronous
line aux 0
line vty 0 4
 login local
 transport input ssh
end
```

Router2#

Рис. 4. Конфігурація Router2 після виконання *playbook cisco.yaml*

Router1 було налаштовано як Master із пріоритетом 120, а Router2 як Backup із пріоритетом 100. Це означає, що у разі нормальної роботи мережі весь трафік до віртуальної IP-адреси 192.168.88.254 обробляється Router1. У разі виходу Router1 з ладу, контроль над трафіком автоматично передається Router2 без втрати зв'язку для кінцевих пристроїв. Варто зазначити, що у конфігурації VRRP відсутнє явне зазначення команди *vrrp 100 preempt*. Це пояснюється тим, що функція автоматичного перемикавання ролі Master увімкнена за замовчуванням на пристроях Cisco, тому її відсутність у конфігурації не впливає на роботу протоколу.

Також було реалізовано базову DHCP-конфігурацію для надання IP-адрес клієнтським пристроям у підмережі VLAN 100. Це включало в себе визначення мережі, шлюзу за замовчуванням і DNS-сервера. Усі ці параметри було успішно впроваджено автоматизованим способом, що значно скоротило час конфігурації та зменшило ризик помилок.

Загалом результати виконання експерименту підтвердили працездатність рішення, а також доцільність використання Ansible для автоматизації налаштування мережних пристроїв. Це дозволило забезпечити високу надійність, мінімізувати ручні налаштування та забезпечити швидке переключення між маршрутизаторами у разі збою основного вузла.

Конфігурація GLBP

Для забезпечення відмовостійкості та балансування навантаження в одній і тій самій підмережі 192.168.88.0/24 можна застосовувати і протокол GLBP. Він створює віртуальний шлюз, який складається з декількох маршрутизаторів, що працюють як одна група. Порівняно з VRRP, GLBP дає змогу розподіляти трафік між усіма вузлами, а не зберігати пасивний режим для резервних пристроїв.

Топологія залишається ідентичною тій, що показана на рисунку 1. У підмережі використовуються два маршрутизатори Cisco (Router1 і Router2), комутатор (Switch) і клієнтський ПК (Client). Router1 має IP-адресу 192.168.88.253, Router2 – 192.168.88.252, а для користувачів віртуальним шлюзом слугує адреса 192.168.88.254. Пристрої об'єднані у VLAN 100, а клієнтська станція отримує адресу 192.168.88.100 із маскою 255.255.255.0 та налаштованим шлюзом 192.168.88.254.

GLBP визначає роль Active Virtual Gateway, яку виконує маршрутизатор з найвищим пріоритетом, і роль Active Virtual Forwarders, що забезпечує реальне балансування трафіку. Якщо основний вузол стає недоступним, його місце може посісти резервний. Так досягається і відмовостійкість, і рівномірне завантаження всіх учасників групи.

Під час налаштування вручну на кожному маршрутизаторі зазвичай прописують IP-адреси інтерфейсів VLAN, створюють GLBP-групу з визначеним ідентифікатором, задають пріоритети та режим preempt. Це дає змогу маршрутизатору з вищим пріоритетом автоматично повертати собі роль Active Virtual Gateway, якщо він знову стає доступним. У лабораторному експерименті пріоритет Router1 налаштовано вищим (наприклад, 120), щоб саме він був головним вузлом для групи GLBP, а Router2 діє як резервний з пріоритетом 100. Обидва маршрутизатори отримують команду load-balancing round-robin для розподілу клієнтських запитів.

```
(root@ansible) # ansible-playbook -i hosts cisco_glbp.yaml

PLAY [Configure GLBP on Cisco routers] *****

TASK [Create VLAN on both routers] *****
[WARNING]: To ensure idempotency and correct diff the input configuration lines should be similar to how they appear if present in the running configuration on device
changed: [Router2]
changed: [Router1]

TASK [Configure GLBP on Router1] *****
changed: [Router1]

TASK [Configure GLBP on Router2] *****
changed: [Router2]

PLAY RECAP *****
Router1      : ok=2   changed=2   unreachable=0   failed=0   skipped=1   rescued=0   ignored=0
Router2      : ok=2   changed=2   unreachable=0   failed=0   skipped=1   rescued=0   ignored=0
```

Рис. 5. Лог виконання *playbook cisco_glbp.yaml*

Аналогічно до сценарію налаштування VRRP було застосовано Ansible, щоб виконувати конфігурацію GLBP централізовано й виключити ризик людських помилок. У підготовленому *playbook* містяться команди для задання адрес VLAN, створення відповідної GLBP-групи та визначення пріоритетів. Запуск відбувається за командою на кшталт *ansible-playbook -i hosts cisco_glbp.yaml*, після чого Ansible послідовно підключається до Router1 і Router2 та вносить усі потрібні зміни. Запуск *playbook* запропоновано виконувати за допомогою команди *ansible-playbook -i hosts cisco_glbp.yaml* (див. рис. 5). У результаті застосування цих конфігурацій Router1 і Router2

утворюють групу GLBP зі спільним віртуальним шлюзом 192.168.88.254, що представлено на рисунках 6 і 7.

```
ip dhcp pool VLAN100
 network 192.168.88.0 255.255.255.0
 default-router 192.168.88.254
 dns-server 1.1.1.1
ip domain-name ownhub.network
no ip cef
no ipv6 cef
spanning-tree mode pvst
spanning-tree extend system-id
interface Ethernet0/0
 description Uplink_to_main_router
 switchport mode access
interface Ethernet0/1
 switchport access vlan 100
 switchport mode access
interface Ethernet0/2
 switchport access vlan 100
 switchport mode access
interface Ethernet0/3
 switchport access vlan 100
 switchport mode access
interface Vlan1
 ip address 10.10.150.10 255.255.255.0
interface Vlan100
 ip address 192.168.88.253 255.255.255.0
 glbp 1 ip 192.168.88.254
 glbp 1 priority 120
 glbp 1 preempt
ip forward-protocol nd
ip http server
ip route 0.0.0.0 0.0.0.0 10.10.150.254
ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
control-plane
 line con 0
 logging synchronous
 line aux 0
 line vty 0 4
 login local
 transport input ssh
end
```

Router1#

Рис. 6. Конфігурація Router1 після застосування *playbook cisco_glbp.yaml*

```
ip dhcp pool VLAN100
 network 192.168.88.0 255.255.255.0
 default-router 192.168.88.254
 dns-server 1.1.1.1
ip domain-name ownhub.network
ip cef
no ipv6 cef
spanning-tree mode pvst
spanning-tree extend system-id
interface Ethernet0/0
 description *** Uplink to main router ***
 switchport mode access
interface Ethernet0/1
 switchport access vlan 100
 switchport mode access
interface Ethernet0/2
 switchport access vlan 100
 switchport mode access
interface Ethernet0/3
 switchport access vlan 100
 switchport mode access
interface Vlan1
 ip address 10.10.150.11 255.255.255.0
interface Vlan100
 ip address 192.168.88.252 255.255.255.0
 glbp 1 ip 192.168.88.254
 glbp 1 preempt
ip forward-protocol nd
ip http server
ip route 0.0.0.0 0.0.0.0 10.10.150.254
ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
control-plane
 line con 0
 logging synchronous
 line aux 0
 line vty 0 4
 login local
 transport input ssh
end
```

Router2#

Рис. 7. Конфігурація Router2 після застосування *playbook cisco_glbp.yaml*

За штатних умов обидва маршрутизатори беруть участь у маршрутизації трафіку, розподіляючи навантаження між собою, а у разі відмови одного з вузлів інший миттєво перебирає його роль. Такий підхід дозволяє одночасно підвищити надійність та оптимізувати використання пропускну здатності мережі.

Централізована автоматизація цього процесу за допомогою Ansible підтвердила доцільність та зручність впровадження. Адміністраторам не потрібно вручну виконувати десятки конфігураційних команд, а всі зміни зберігаються у версіонованому репозиторії. Успішне налаштування підтверджене коректною роботою маршрутизаторів і відсутністю збоїв під час тестування відмовостійкості та балансування навантаження.

4. Тестування відмовостійкості шлюзів у разі використання VRRP та GLBP на фізичних і віртуальних пристроях

Тестування відмовостійкості VRRP та GLBP на фізичних і віртуальних пристроях у межах цієї роботи мало на меті визначити швидкість перемикання між маршрутизаторами у разі виходу з ладу основного вузла та порівняти ефективність роботи мережі у випадку застосування різних протоколів відмовостійкості. Одним із ключових аспектів було дослідження того, наскільки відрізняються показники у середовищі EVE-NG та на реальному мережному обладнанні Cisco. Для цього було налаштовано VRRP і GLBP на двох комутаторах третього рівня Cisco, а сама топологія залишалася аналогічною до схеми, яка зображена на рисунку 1. У віртуальному середовищі вдалося швидко змінювати параметри та емулювати відмови, що дало змогу отримати більш точні дані про затримки та втрати пакетів у різних випадках роботи мережі. Вихідні параметри фізичного експерименту представлено в таблиці 2.

Таблиця 2

Вихідні дані для тестування відмовостійкості шлюзів у разі використання VRRP та GLBP на фізичному обладнанні

Категорія	Опис
Мета тестування	Визначення швидкості перемикання між маршрутизаторами при відмові основного вузла та порівняння ефективності VRRP і GLBP у середовищі EVE-NG та на фізичних пристроях.
Протоколи	VRRP, GLBP
Віртуальне середовище	EVE-NG
Фізичне середовище	Два маршрутизатори Cisco, з'єднані відповідно до топології рисунку 1
Комутатори третього рівня	Cisco WS-C3750G-24TS-1U, Cisco WS-C3560-24PS
Прошивка комутаторів	15.0(2)SE11
Тип з'єднання	Мідні литі патчкорди для мінімізації затримок і втрат сигналу
Налаштування STP	Spanning-Tree PortFast для зменшення часу навчання MAC-адрес
Метод емуляції відмов	У віртуальному середовищі – зміна параметрів; у фізичному – відключення живлення одного пристрою
Досліджувані показники	Швидкість перемикання між вузлами, втрати пакетів при відмові основного шлюза

Отже, в ході експерименту використано два комутатори Cisco WS-C3750G-24TS-1U та Cisco WS-C3560-24PS, які встановлені поруч і з'єднані між собою короткими мідними литими патчкордами. Такий вибір патчкордів дав змогу мінімізувати можливі втрати сигналу і запобігти виникненню додаткових затримок на фізичному рівні. Комутатори працювали з прошивкою 15.0(2)SE11 та були налаштовані у режимі Spanning-Tree PortFast для зменшення часу, необхідного для навчання MAC-адрес. Для отримання більш репрезентативних результатів у фізичному середовищі обрано сценарій, коли один з пристроїв повністю знеструлювався від'єднанням кабелю живлення. Під час цього досліджувалося, наскільки швидко інший маршрутизатор VRRP чи GLBP приймає на себе обробку трафіку і скільки пакетів при цьому втрачається.

У віртуальному середовищі EVE-NG, де проводився аналогічний експеримент, також використовувалася та сама топологія, але вже з віртуальними маршрутизаторами на базі образу i86bi_LinuxL2-AdvEnterpriseK9-M_152_May_2018.bin. Цю конфігурацію можна побачити на рисунку 8.

В параметрах Edit node було виставлено NVRAM у розмірі 512 Кб та оперативну пам'ять 128 Мб, щоб віртуальна лабораторія за своїми характеристиками не поступалася фізичним пристроєм і була максимально наближена до реальних умов експлуатації. Відмінність полягала лише в тому, що замість фізичних з'єднань тут застосовувалися віртуальні, що загалом зменшує затримки. Завдяки цьому результати роботи мережі в EVE-NG показали менше втрачених пакетів при відмові одного з маршрутизаторів. Фактично віртуальна інфраструктура забезпечила високу швидкість перемикання та вищу масштабованість ресурсів, тому час конвергенції був мінімальним.

Віртуальна лабораторія була розгорнута у середовищі Free EVE Community Edition Version 6.2.0-4, де використовується Qemu версії 2.4.0. Сама інсталяція EVE-NG виконана на VMware ESXi 8.0.3 з номером збірки 24280767, встановленій на хості з процесором 13th Gen Intel Core i3-13100, 128 Гб DDR5 та твердотільним NVMe-накопичувачем Kingston KC3000 на 2 Тб. Всередині гіпервізора EVE-NG було розгорнуто на операційній системі Ubuntu 22.04.5 LTS з ядром GNU/Linux 6.7.5-eveng-6-ksm x86_64. Віртуальній машині, на якій працює EVE-NG, виділено 2 віртуальні ядра, 4 Гб оперативної пам'яті та 100 Гб дискового простору. Для підключення до мережі застосовано інтерфейс vmxnet 3. Параметри конфігурації ресурсів, наданих віртуальній

машині, можна побачити на рисунку 9, де також відображається режим сумісності ESXi 6.7 і більш пізньої VM version 14. Підключення здійснюється через віртуальний мережний інтерфейс DPG-EVE-NG, що дозволяє зручно інтегрувати EVE-NG у реальну або лабораторну мережну інфраструктуру.

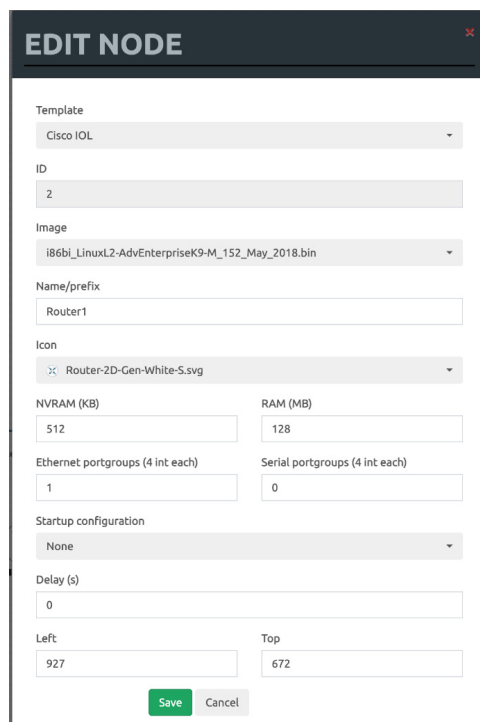


Рис. 8. Конфігурація віртуального маршрутизатора

Для емуляції збою маршрутизатора у віртуальній лабораторії застосовувався спеціальний механізм різкого вимкнення процесу, який відповідає за роботу віртуальної Cisco. На рисунку 10 показано, як використовується команда *unl_wrapper* з опцією *stop* для миттєвої зупинки віртуального маршрутизатора Router1. Це дає змогу з високою точністю фіксувати, коли саме сталася відмова, а потім вимірювати час, необхідний на повне відновлення після запуску *unl_wrapper* з опцією *start*. Повний лог цих операцій наведено нижче, де можна побачити, як система припиняє роботу вузла та потім знову активує його на тому ж місці мережної топології. Такий підхід у віртуальному середовищі є надзвичайно зручним, оскільки дозволяє відтворити умови реальної відмови і ретельно проаналізувати поведінку протоколів VRRP та GLBP.

Аналогічний підхід було запропоновано для фізичного експерименту, але там відмова вузла виконувалася шляхом відключення маршрутизатора від електромережі. Це більш жорстка модель збою, проте завдяки їй можна отримати наближені до життєвих умови роботи мережі. Протокол Spanning-Tree PortFast на комутаторах прискорює перехід у стан Forwarding, але на фізичних пристроях час конвергенції все одно дещо більший через апаратні обмеження і потребу в оновленні таблиць MAC-адрес. Також слід зазначити, що у разі відключення та повторного під'єднання живлення комутатор додатково виконує перевірку та ініціалізацію інтерфейсів, що збільшує затримку перед відновленням повної працездатності мережі.

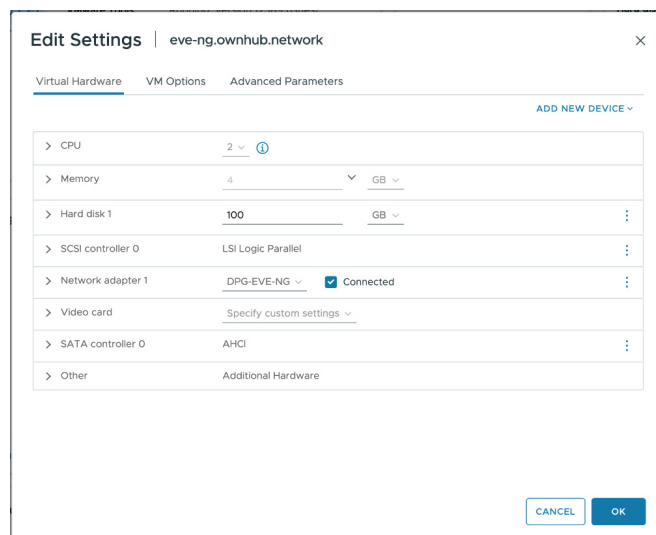


Рис. 9. Конфігурація ресурсів EVE-NG

```
[root@eve-ng:/opt/unetlab/labs# /opt/unetlab/wrappers/unl_wrapper -a stop -F /opt/unetlab/labs/Cisco.unl -T 0 -D 2
Jan 07 23:03:54 INFO: stopping fuser -k -TERM /opt/unetlab/tmp/0/ef873186-c9ae-4c01-9138-2b5cd5396bc7/2 > /dev/null 2>&1
[root@eve-ng:/opt/unetlab/labs# /opt/unetlab/wrappers/unl_wrapper -a start -F /opt/unetlab/labs/Cisco.unl -T 0 -D 2

Jan 07 23:04:35 INFO: tuncctl -u unl0 -g root -t vunl0_2_0 2>&1
Jan 07 23:04:35 INFO: sysctl -w net.ipv6.conf.vunl0_2_0.disable_ipv6=1
Jan 07 23:04:35 INFO: tuncctl -u unl0 -g root -t vunl0_2_16 2>&1
Jan 07 23:04:35 INFO: sysctl -w net.ipv6.conf.vunl0_2_16.disable_ipv6=1
Jan 07 23:04:35 INFO: tuncctl -u unl0 -g root -t vunl0_2_32 2>&1
Jan 07 23:04:35 INFO: sysctl -w net.ipv6.conf.vunl0_2_32.disable_ipv6=1
Jan 07 23:04:35 INFO: tuncctl -u unl0 -g root -t vunl0_2_48 2>&1
Jan 07 23:04:35 INFO: sysctl -w net.ipv6.conf.vunl0_2_48.disable_ipv6=1
Jan 07 23:04:35 INFO: CMD is /opt/unetlab/tmp/0/ef873186-c9ae-4c01-9138-2b5cd5396bc7/2
Jan 07 23:04:35 INFO: starting /opt/unetlab/wrappers/iol_wrapper -T 0 -D 2 -t "Router1" -F /opt/unetlab/addons/iol/bin/i86bi_LinuxL2-AdvEnterpriseK9-M_152_May_2018.bin -d 0 -e 1 -s 0 --
-n 1024 -q -n 1024 > /opt/unetlab/tmp/0/ef873186-c9ae-4c01-9138-2b5cd5396bc7/2/wrapper.txt 2>&1 &
root@eve-ng:/opt/unetlab/labs#
```

Рис. 10. Емуляція збою маршрутизатора Router1

Для оцінки доступності шлюзу застосовувалися два типи пінгування. Перший – це стандартний ping із параметрами за замовчуванням, а другий – це частий ping з інтервалом 0,1 секунди. За таких умов виконувався *ping 1.1.1.1 -i 0.1*, що дозволяло виявити неспроможність маршрутизаторів обробляти велику кількість запитів за короткий проміжок часу. У кожній конфігурації виконано по десять тестів, а отримані середні показники наведено в таблиці 3. Для проведення вимірювань було використано обладнання під управлінням RouterOS (Mikrotik). Цей вибір обумовлений тим, що реалізація ping у Mikrotik передбачає встановлення нового з'єднання для кожного пакету [17], що забезпечує більш реалістичні результати порівняно з підходами, де використовується незмінне з'єднання, як, наприклад, у ОС Windows або Linux.. На рисунку 11 зображено, як виконувався ping в інтерфейсі терміналу на Mikrotik, і за таких обставин можна побачити, що під час тестування VRRP на фізичних пристроях з інтервалом -i 0.1 втрачається приблизно тридцять чотири пакети.

Як свідчать дані, наведені в таблиці 3, EVE-NG демонструє швидшу реакцію у разі відмови вузла та менші втрати пакетів у VRRP. За стандартного інтервалу пінгу втрати становлять усього 1–2 пакети, а при інтервалі 0,1 секунди – 6–8. Це свідчить про ефективну конвергенцію та мінімальний час простою у віртуальній лабораторії. На фізичних пристроях втрати для VRRP більші: 3–4 пакети за стандартного пінгу і 25–35 при частішому. Подібна різниця вказує на суттєвий вплив реального мережного обладнання та його апаратних обмежень. Більше того, впливають додаткові затримки при переналаштуванні портів і відновленні таблиць MAC-адрес.

```

Terminal <1>
[admin@MikroTik] > ping 192.168.88.254 interval=0.1
  SEQ HOST                SIZE TTL TIME          STATUS
  ---
0 192.168.88.254          56 255 527us         success
1 192.168.88.254          56 255 428us         success
2 192.168.88.254          56 255 561us         success
3 192.168.88.254          56 255 497us         success
4 192.168.88.254          56 255 1ms98us        success
5 192.168.88.254          56 255 458us         success
6 192.168.88.254          56 255 458us         success
7 192.168.88.254          56 255 458us         success
8 192.168.88.254          56 255 458us         success
9 192.168.88.254          56 255 458us         success
10 192.168.88.254          56 255 458us         success
11 192.168.88.254          56 255 458us         success
12 192.168.88.254          56 255 458us         success
13 192.168.88.254          56 255 458us         success
14 192.168.88.254          56 255 458us         success
15 192.168.88.254          56 255 458us         success
16 192.168.88.254          56 255 458us         success
17 192.168.88.254          56 255 458us         success
18 192.168.88.254          56 255 458us         success
19 192.168.88.254          56 255 458us         success
sent=20 received=6 packet-loss=70% min-rtt=428us avg-rtt=584us max-rtt=1ms98us
  SEQ HOST                SIZE TTL TIME          STATUS
  ---
20 192.168.88.254          56 255 458us         success
21 192.168.88.254          56 255 458us         success
22 192.168.88.254          56 255 458us         success
23 192.168.88.254          56 255 458us         success
24 192.168.88.254          56 255 458us         success
25 192.168.88.254          56 255 458us         success
26 192.168.88.254          56 255 458us         success
27 192.168.88.254          56 255 458us         success
28 192.168.88.254          56 255 458us         success
29 192.168.88.254          56 255 458us         success
30 192.168.88.254          56 255 458us         success
31 192.168.88.254          56 255 458us         success
32 192.168.88.254          56 255 458us         success
33 192.168.88.254          56 255 458us         success
34 192.168.88.254          56 255 458us         success
35 192.168.88.254          56 255 458us         success
36 192.168.88.254          56 255 458us         success
37 192.168.88.254          56 255 458us         success
38 192.168.88.254          56 255 458us         success
39 192.168.88.254          56 255 458us         success
sent=40 received=6 packet-loss=85% min-rtt=428us avg-rtt=584us max-rtt=1ms98us
  SEQ HOST                SIZE TTL TIME          STATUS
  ---
40 192.168.88.254          56 255 720us         success
41 192.168.88.254          56 255 656us         success
42 192.168.88.254          56 255 449us         success
43 192.168.88.254          56 255 1ms492us       success
44 192.168.88.254          56 255 549us         success
sent=45 received=11 packet-loss=75% min-rtt=428us avg-rtt=670us max-rtt=1ms492us

```

Рис. 11. Тестування втрати пакетів VRRP на фізичних пристроях

GLBP у середовищі EVE-NG засвідчив загалом більші втрати пакетів у порівнянні з VRRP. Особливо це помітно за інтервалу 0,1 секунди, коли втрачалось 90–95 пакетів, що є досить високим показником. У фізичному середовищі втрати виявилися ще більшими: 14–19 пакетів при стандартному інтервалі та 120–170 при частішому. Така поведінка GLBP пояснюється тим, що під час розподілу навантаження між кількома маршрутизаторами виникає додаткове навантаження на кожен вузол мережі, а переналаштування портів і комутаційних таблиць на фізичних пристроях впливає ще відчутніше.

Таблиця 3

Порівняння результатів пінгування для VRRP та GLBP у різних середовищах

Тип тесту	Середня кількість втрачених пакетів з VRRP (EVE-NG)	Середня кількість втрачених пакетів з VRRP (Фізичні пристрої)	Середня кількість втрачених пакетів з GLBP (EVE-NG)	Середня кількість втрачених пакетів з GLBP (Фізичні пристрої)
Ping з інтервалом за замовчуванням	1-2	3-4	9-15	14-19
Ping з інтервалом 0,1 секунди	6-8	25-35	90-95	120-170

Отримані результати ілюструють, що VRRP є ефективнішим для швидкого відновлення мережної працездатності з мінімальними втратами пакетів. GLBP, забезпечуючи збалансоване розподілення трафіку, супроводжується більшими втратами, особливо у фізичному середовищі. Віртуальне середовище EVE-NG, завдяки віртуалізованим каналам і можливостям гнучко виділяти ресурси, виявилось більш сприятливим для тестування. Менші затримки, швидша обробка пакетів, а також відсутність великої кількості апаратних обмежень дозволили мережі працювати з кращими показниками.

Водночас потрібно враховувати, що тестування в EVE-NG не завжди може відобразити всі аспекти фізичної інфраструктури. Певні додаткові затримки, викликані роботою справжніх портів, часовими інтервалами ініціалізації обладнання, а також особливостями живлення та охолодження, у реальному середовищі можуть призвести до інших результатів. Проте з точки зору точності часу відмови і швидкості перемикання протоколів, віртуальна лабораторія значно спрощує аналіз. Тому в критичних мережах, де потрібно досягти максимальної відмовостійкості, корисно спочатку провести дослідження у EVE-NG, щоб відпрацювати схеми відновлення і налаштування, а потім вже реалізувати їх на фізичних пристроях із доопрацюванням з урахуванням апаратних особливостей.

Загалом результати випробувань підтвердили, що за наявності потужних віртуальних ресурсів, таких як виділені ядра процесора, достатньо ОЗП та швидкі SSD-накопичувачі NVMe, віртуальна інфраструктура здатна забезпечити високопродуктивну роботу навіть у складних мережних сценаріях. Детальні показники наведено у таблиці 3, де можна побачити, наскільки сильно відрізняються втрати пакетів у VRRP та GLBP при різних частотах пінгування, а також у фізичному та віртуальному середовищі. Таким чином, можна зробити висновок, що у разі підвищених вимог до надійності та швидкості переключення перевагу доцільно надати VRRP, особливо якщо пріоритетним є зменшення кількості втрат пакетів. Якщо ж потрібен автоматичний розподіл навантаження, тоді GLBP може бути доцільним, але потрібно враховувати його більшу схильність до затримок і втрат, що виявляється особливо критичним на фізичних пристроях з обмеженими ресурсами.

З огляду на викладене, можна підсумувати, що віртуальне середовище EVE-NG має перевагу в простоті налаштування та швидкості тестування, а також у здатності оперативно змінювати параметри. Воно дозволяє проаналізувати відмовостійкість та ефективність протоколів динамічного резервування шлюзів без великого ризику пошкодження реального обладнання. Якщо ж виникає потреба врахувати всі фізичні особливості та отримати більш прикладні результати, тоді рекомендовано додатково проводити тести на реальному обладнанні. Така поетапна методика тестування забезпечує найбільш комплексний підхід: спочатку віртуальна перевірка запропоновано нового рішення або зміни в конфігурації, а потім уже остаточна валідація на фізичних пристроях.

5. Тестування відмовостійкості шлюзів у разі використання VRRP та GLBP на віртуальних пристроях у середовищі Containerlab

У подальшому для тестування відмовостійкості шлюзів були розгорнуті три окремі середовища Containerlab [15, 18], які моделюють роботу VRRP та GLBP на маршрутизаторах Cisco та VyOS. У кожному з середовищ було реалізовано базову топологію з двома маршрутизаторами, комутатором і кінцевим вузлом (контейнером Ubuntu).

Containerlab – це інструмент командного рядка для оркестрації та управління мережними лабораторіями на основі контейнерів. Він дозволяє легко створювати та керувати топологіями мереж, які включають як контейнеризовані мережні операційні системи (Network Operating System, NOS), так і традиційні віртуальні машини. Containerlab надає зручний спосіб підключення контейнерів між собою, що дозволяє моделювати складні мережує сценарії. Основні переваги Containerlab включають підхід «Lab as Code» (LaC), який дозволяє описувати топології за допомогою декларативних конфігураційних файлів, швидке розгортання лабораторій на будь-якій системі з Docker, а також підтримку мультивендорних рішень. Для тестування відмовостійкості шлюзів у нашому дослідженні були розгорнуті три окремі середовища Containerlab, які моделюють роботу протоколів VRRP та GLBP на маршрутизаторах Cisco та VyOS. Конфігураційні файли цих середовищ розміщено у відкритому репозиторії Zenodo для зручності відтворення експериментів [19].

У першому середовищі було розгорнуто два маршрутизатори Cisco IOL, комутатор Cisco IOL L2 та кінцевий вузол на базі Ubuntu (таблиця 4). Водночас топологія після розгортання має вигляд, показаний на рис. 12.

Таблиця 4

Середовище 1 Cisco VRRP

Назва пристрою	Тип пристрою	Образ ОС	IPv4 адреса
clab-cisco-vrrp-lab-iol-r1	Маршрутизатор (cisco_iol)	Cisco_iol:17.15.01	172.20.32.4
clab-cisco-vrrp-lab-iol-r2	Маршрутизатор (cisco_iol)	Cisco_iol:17.15.01	172.20.32.3
clab-cisco-vrrp-lab-iol-switch	Комутатор (cisco_iol)	Cisco_iol:12.17.15.01	172.20.32.5
clab-cisco-vrrp-lab-vm-ubuntu	Контейнер під управлінням linux	Ubuntu:24.04-noble	172.20.32.2

```

root@containerlab:~/containerlab_dir# containerlab deploy -t cisco.clab.yaml
INFO[0000] Containerlab v0.61.0 started
INFO[0000] Parsing & checking topology file: cisco.clab.yaml
INFO[0000] Creating docker network: Name="clab", IPv4Subnet="172.20.32.0/24", IPv6Subnet="", MTU=1500
INFO[0000] Creating lab directory: /root/containerlab_dir/clab-cisco-vrrp-lab
INFO[0000] Creating container: "iol-r1"
INFO[0000] Creating container: "vm-ubuntu"
INFO[0000] Creating container: "iol-r2"
INFO[0001] Creating container: "iol-switch"
INFO[0001] Running postdeploy actions for Cisco IOL 'iol-r1' node
INFO[0001] Running postdeploy actions for Cisco IOL 'iol-r2' node
INFO[0001] Created link: iol-r1:eth1 (Ethernet0/1) <--> iol-switch:eth1 (Ethernet0/1)
INFO[0001] Created link: iol-r2:eth1 (Ethernet0/1) <--> iol-switch:eth2 (Ethernet0/2)
INFO[0002] Created link: vm-ubuntu:eth1 <--> iol-switch:eth3 (Ethernet0/3)
INFO[0002] Running postdeploy actions for Cisco IOL 'iol-switch' node
INFO[0012] Adding containerlab host entries to /etc/hosts file
INFO[0012] Adding ssh config for containerlab nodes
INFO[0012] 🐳 New containerlab version 0.65.1 is available! Release notes: https://containerlab.dev/rn/0.65/#0651
Run 'containerlab version upgrade' to upgrade or go check other installation options at https://containerlab.dev/install/

```

Рис. 12. Стан топології після створення віртуального середовища на базі cisco для VRRP

Після налаштування VRRP було виконано перевірку стану маршрутизаторів за допомогою команди `show vrrp`. Відповідний скріншот представлено на рис. 13.

<pre> iol-r1#show vrrp Ethernet0/1 - Group 1 State is Backup Virtual IP address is 192.168.1.254 Virtual MAC address is 0000.5e00.0101 Advertisement interval is 1.000 sec Preemption enabled Priority is 100 Master Router is 192.168.1.2, priority is 100 Master Advertisement interval is 1.000 sec Master Down interval is 3.609 sec (expires in 3.029 sec) FLAGS: 0/1 </pre>	<pre> iol-r2#show vrrp Ethernet0/1 - Group 1 State is Master Virtual IP address is 192.168.1.254 Virtual MAC address is 0000.5e00.0101 Advertisement interval is 1.000 sec Preemption enabled Priority is 100 Master Router is 192.168.1.2 (local), priority is 100 Master Advertisement interval is 1.000 sec Master Down interval is 3.609 sec FLAGS: 1/1 </pre>
---	--

Рис. 13. Результат перевірки налаштування VRRP на роутерах Cisco

У другому середовищі аналогічно було розгорнуто два маршрутизатори Cisco IOL, комутатор Cisco IOL L2 та кінцевий вузол на базі Ubuntu (таблиця 5). Тоді як топологія показана на рис. 14.

Таблиця 5

Середовище 2 Cisco GLBP

Назва пристрою	Тип пристрою	Образ ОС	IPv4 адреса
clab-cisco-glb-lab-iol-r1	Маршрутизатор (cisco_iol)	Cisco_iol:17.15.01	172.20.32.4
clab-cisco-glb-lab-iol-r2	Маршрутизатор (cisco_iol)	Cisco_iol:17.15.01	172.20.32.3
clab-cisco-glb-lab-iol-switch	Комутатор (cisco_iol)	Cisco_iol:12.17.15.01	172.20.32.5
clab-cisco-glb-lab-vm-ubuntu	Контейнер під управлінням linux	Ubuntu:24.04-noble	172.20.32.2

```

root@containerlab:~/containerlab_dir# containerlab deploy -t cisco-glbp.clab.yaml
INFO[0000] Containerlab v0.61.0 started
INFO[0000] Parsing & checking topology file: cisco-glbp.clab.yaml
INFO[0000] Creating docker network: Name="clab", IPv4Subnet="172.20.32.0/24", IPv6Subnet="", MTU=1500
INFO[0000] Creating lab directory: /root/containerlab_dir/clab-cisco-glbp-lab
INFO[0000] Creating container: "vm-ubuntu"
INFO[0000] Creating container: "iol-r1"
INFO[0000] Creating container: "iol-r2"
INFO[0001] Creating container: "iol-switch"
INFO[0001] Running postdeploy actions for Cisco IOL 'iol-r2' node
INFO[0001] Running postdeploy actions for Cisco IOL 'iol-r1' node
INFO[0001] Created link: iol-r1:eth1 (Ethernet0/1) <--> iol-switch:eth1 (Ethernet0/1)
INFO[0002] Created link: iol-r2:eth1 (Ethernet0/1) <--> iol-switch:eth2 (Ethernet0/2)
INFO[0002] Created link: vm-ubuntu:eth1 <--> iol-switch:eth3 (Ethernet0/3)
INFO[0002] Running postdeploy actions for Cisco IOL 'iol-switch' node
INFO[0012] Adding containerlab host entries to /etc/hosts file
INFO[0012] Adding ssh config for containerlab nodes
INFO[0012] 🚀 New containerlab version 0.65.1 is available! Release notes: https://containerlab.dev/rn/0.65/#0651
Run 'containerlab version upgrade' to upgrade or go check other installation options at https://containerlab.dev/install/

```

Рис. 14. Стан топології після створення віртуального середовища для GLBP

Після налаштування GLBP виконано перевірку стану маршрутизаторів командою `show glbp`. Отримані результати відображено на рис. 15.

```

iol-r1#show glbp
Ethernet0/1 - Group 1
State is Standby
  1 state change, last state change 00:05:10
Virtual IP address is 192.168.1.254
Hello time 3 sec, hold time 10 sec
Next hello sent in 1.696 secs
Redirect time 600 sec, forwarder time-out 14400 sec
Preemption enabled, min delay 0 sec
Active is 192.168.1.2, priority 100 (expires in 10.880 sec)
Standby is local
Priority 100 (default)
Weighting 100 (default 100), thresholds: lower 1, upper 100
Load balancing: round-robin
Group members:
  aabb.cc00.0110 (192.168.1.1) local
  aabb.cc00.0210 (192.168.1.2)
There are 2 forwarders (1 active)
Forwarder 1
  State is Listen
  MAC address is 0007.b400.0101 (learned)
  Owner ID is aabb.cc00.0210
  Time to live: 14398.976 sec (maximum 14400 sec)
  Preemption enabled, min delay 30 sec
  Active is 192.168.1.2 (primary), weighting 100 (expires in 10.944 sec)
  FLAGS: 0/1
Forwarder 2
  State is Active
  1 state change, last state change 00:05:03
  MAC address is 0007.b400.0102 (default)
  Owner ID is aabb.cc00.0110
  Preemption enabled, min delay 30 sec
  Active is local, weighting 100
  FLAGS: 1/1
iol-r1#

iol-r2#show glbp
Ethernet0/1 - Group 1
State is Active
  1 state change, last state change 00:05:30
Virtual IP address is 192.168.1.254
Hello time 3 sec, hold time 10 sec
Next hello sent in 1.952 secs
Redirect time 600 sec, forwarder time-out 14400 sec
Preemption enabled, min delay 0 sec
Active is local
Standby is 192.168.1.1, priority 100 (expires in 9.344 sec)
Priority 100 (default)
Weighting 100 (default 100), thresholds: lower 1, upper 100
Load balancing: round-robin
Group members:
  aabb.cc00.0110 (192.168.1.1)
  aabb.cc00.0210 (192.168.1.2) local
There are 2 forwarders (1 active)
Forwarder 1
  State is Active
  1 state change, last state change 00:05:20
  MAC address is 0007.b400.0101 (default)
  Owner ID is aabb.cc00.0210
  Redirection enabled
  Preemption enabled, min delay 30 sec
  Active is local, weighting 100
  FLAGS: 1/1
Forwarder 2
  State is Listen
  MAC address is 0007.b400.0102 (learned)
  Owner ID is aabb.cc00.0110
  Redirection enabled, 599.360 sec remaining (maximum 600 sec)
  Time to live: 14399.360 sec (maximum 14400 sec)
  Preemption enabled, min delay 30 sec
  Active is 192.168.1.1 (primary), weighting 100 (expires in 11.296 sec)
  FLAGS: 0/1

```

Рис. 15. Результат перевірки налаштування GLBP на роутерах Cisco

У третьому середовищі замість Cisco IOL були використані маршрутизатори VyOS. Крім того, мережну інфраструктуру доповнює комутатор Cisco IOL L2 та кінцевий вузол Ubuntu (таблиця 6, рис. 16).

Таблиця 6

Середовище 3 Vyos VRRP

Назва пристрою	Тип пристрою	Образ ОС	IPv4 адреса
clab-vyos-vrrp-lab-vyos1	Контейнер під управлінням linux	vyos-1.5-rolling-202501110007:latest	172.20.32.5
clab-vyos-vrrp-lab-vyos2	Контейнер під управлінням linux	vyos-1.5-rolling-202501110007:latest	172.20.32.3
clab-vyos-vrrp-lab-switch	Комутатор (cisco_iol)	Cisco_iol:12.17.15.01	172.20.32.2
clab-cisco-glbp-lab-vm-ubuntu	Контейнер під управлінням linux	Ubuntu:24.04-noble	172.20.32.4

```

root@containerlab:~/containerlab_dir# containerlab deploy -t vyos-vrrp.clab.yaml
INFO[0000] Containerlab v0.61.0 started
INFO[0000] Parsing & checking topology file: vyos-vrrp.clab.yaml
INFO[0000] Creating docker network: Name="clab", IPv4Subnet="172.20.20.0/24", IPv6Subnet="3fff:172:20:20::/64", MTU=1500
INFO[0000] Creating lab directory: /root/containerlab_dir/clab-vyos-vrrp-lab
INFO[0000] Creating container: "vyos2"
INFO[0000] Creating container: "vyos1"
INFO[0000] Creating container: "switch"
INFO[0000] Creating container: "ubuntu"
INFO[0001] Created link: vyos2:eth1 <--> switch:eth2 (Ethernet0/2)
INFO[0001] Running postdeploy actions for Cisco IOL 'switch' node
INFO[0001] Created link: ubuntu:eth1 <--> switch:eth3 (Ethernet0/3)
INFO[0001] Created link: vyos1:eth1 <--> switch:eth1 (Ethernet0/1)
INFO[0011] Adding containerlab host entries to /etc/hosts file
INFO[0011] Adding ssh config for containerlab nodes
INFO[0011] 🚀 New containerlab version 0.65.1 is available! Release notes: https://containerlab.dev/rn/0.65/#0651
Run 'containerlab version upgrade' to upgrade or go check other installation options at https://containerlab.dev/install/

```

Рис. 16. Стан топології після створення віртуального середовища на базі VyOS для VRRP

Для перевірки роботи VRRP на VyOS використана команда `run show vrrp`, результати якої представлено на рис. 17.

Name	Interface	VRRID	State	Priority	Last Transition
VRRP-1	eth1	10	MASTER	110	1m19s

Name	Interface	VRRID	State	Priority	Last Transition
VRRP-1	eth1	10	BACKUP	100	27s

Рис. 17. Результат перевірки налаштування VRRP на роутерах під управлінням VyOS

Тестування роботи механізмів відмовостійкості виконувалося за допомогою перевірки доступності віртуальної IP-адреси шлюза після примусового відключення одного з маршрутизаторів. Як метрика оцінки використовувалася кількість втрачених ICMP-пакетів (*ping*), що свідчить про час переходу управління шлюзом на резервний маршрутизатор.

Кожен тест проводився у два етапи:

- надсилання ICMP-запитів до віртуального шлюза з інтервалом 1 секунда;
- надсилання ICMP-запитів з інтервалом 0,5 секунди.

Тестування виконувалося командами:

```
ping -c 10 -i 1 192.168.1.254
```

```
ping -c 10 -i 0.5 192.168.1.254
```

```

root@vm-ubuntu:/# ping -c 10 -i 1 192.168.1.254
PING 192.168.1.254 (192.168.1.254) 56(84) bytes of data.
 64 bytes from 192.168.1.254: icmp_seq=1 ttl=255 time=3.25 ms
 64 bytes from 192.168.1.254: icmp_seq=2 ttl=255 time=2.96 ms
 64 bytes from 192.168.1.254: icmp_seq=4 ttl=255 time=3.11 ms
 64 bytes from 192.168.1.254: icmp_seq=5 ttl=255 time=3.17 ms
 64 bytes from 192.168.1.254: icmp_seq=6 ttl=255 time=2.95 ms
 64 bytes from 192.168.1.254: icmp_seq=7 ttl=255 time=3.39 ms
 64 bytes from 192.168.1.254: icmp_seq=8 ttl=255 time=3.29 ms
 64 bytes from 192.168.1.254: icmp_seq=9 ttl=255 time=2.98 ms
 64 bytes from 192.168.1.254: icmp_seq=10 ttl=255 time=3.04 ms

```

```
--- 192.168.1.254 ping statistics ---
```

```
10 packets transmitted, 9 received, 10% packet loss, time 9031ms
rtt min/avg/max/mdev = 2.946/3.126/3.386/0.149 ms
```

```

root@vm-ubuntu:/# ping -c 10 -i 0.5 192.168.1.254
PING 192.168.1.254 (192.168.1.254) 56(84) bytes of data.
 64 bytes from 192.168.1.254: icmp_seq=1 ttl=255 time=1.91 ms
 64 bytes from 192.168.1.254: icmp_seq=2 ttl=255 time=2.33 ms
 64 bytes from 192.168.1.254: icmp_seq=4 ttl=255 time=3.22 ms
 64 bytes from 192.168.1.254: icmp_seq=5 ttl=255 time=3.25 ms
 64 bytes from 192.168.1.254: icmp_seq=6 ttl=255 time=2.69 ms
 64 bytes from 192.168.1.254: icmp_seq=7 ttl=255 time=3.16 ms
 64 bytes from 192.168.1.254: icmp_seq=8 ttl=255 time=3.01 ms
 64 bytes from 192.168.1.254: icmp_seq=9 ttl=255 time=3.10 ms
 64 bytes from 192.168.1.254: icmp_seq=10 ttl=255 time=2.74 ms

```

```
--- 192.168.1.254 ping statistics ---
```

```
10 packets transmitted, 9 received, 10% packet loss, time 4527ms
rtt min/avg/max/mdev = 1.912/2.824/3.248/0.429 ms
```

Рис. 18. Результат перевірки роботи VRRP на роутерах під управлінням Cisco

Для GLBP також було виконано додатковий тест з більшою кількістю запитів:

```
ping -c 30 -i 0.5 192.168.1.254
```

Результати проведених тестів демонструють, що використання протоколу VRRP на маршрутизаторах Cisco та VyOS забезпечує мінімальні втрати трафіку під час перемикання між активним та резервним шлюзами. Втрати обмежуються одним ICMP-пакетом незалежно від встановленого інтервалу перевірки.

На відміну від цього, протокол GLBP на маршрутизаторах Cisco показав значно більший час перемикання, що призводило до втрати від 9–10 ICMP-пакетів у коротких тестах та до 17–18 пакетів при збільшенні кількості ICMP-запитів (таблиця 7).

```

root@vm-ubuntu:/# ping -c 30 -i 0.5 192.168.1.254
PING 192.168.1.254 (192.168.1.254) 56(84) bytes of data.
64 bytes from 192.168.1.254: icmp_seq=1 ttl=255 time=2.30 ms
64 bytes from 192.168.1.254: icmp_seq=20 ttl=255 time=2.35 ms
64 bytes from 192.168.1.254: icmp_seq=21 ttl=255 time=3.23 ms
64 bytes from 192.168.1.254: icmp_seq=22 ttl=255 time=2.88 ms
64 bytes from 192.168.1.254: icmp_seq=23 ttl=255 time=3.09 ms
64 bytes from 192.168.1.254: icmp_seq=24 ttl=255 time=2.47 ms
64 bytes from 192.168.1.254: icmp_seq=25 ttl=255 time=3.35 ms
64 bytes from 192.168.1.254: icmp_seq=26 ttl=255 time=3.42 ms
64 bytes from 192.168.1.254: icmp_seq=27 ttl=255 time=4.82 ms
64 bytes from 192.168.1.254: icmp_seq=28 ttl=255 time=3.07 ms
64 bytes from 192.168.1.254: icmp_seq=29 ttl=255 time=3.11 ms
64 bytes from 192.168.1.254: icmp_seq=30 ttl=255 time=3.37 ms

--- 192.168.1.254 ping statistics ---
30 packets transmitted, 12 received, 60% packet loss, time 14743ms
rtt min/avg/max/mdev = 2.303/3.121/4.820/0.635 ms

```

Рис. 19. Результат перевірки роботи GLBP на роутерах під управлінням Cisco

Отримані дані свідчать про те, що VRRP є більш ефективним рішенням для забезпечення безперервності з'єднання порівняно з GLBP. Швидкість реакції VRRP на відмову основного шлюза значно вища, що робить його більш прийнятним для використання в мережах з високими вимогами до мінімізації простоїв і затримок. Це особливо важливо для середовищ, де навіть незначні перерви у роботі мережі можуть призводити до критичних наслідків.

Таблиця 7

Порівняння результатів пінгування для VRRP та GLBP у різних середовищах

Тип тесту	Середня кількість втрачених пакетів з VRRP Cisco (Containerlab)	Середня кількість втрачених пакетів з GLBP Cisco (Containerlab)	Середня кількість втрачених пакетів з VRRP VyOS (Containerlab)
Ping з інтервалом за замовчуванням	1	9-10	2
Ping з інтервалом 0,5 секунди	1-2	18	4-5

Аналіз отриманих експериментальних даних свідчить про те, що використання VRRP забезпечує мінімальний час переходу між активним і резервним шлюзом, що підтверджується стабільними втратами на рівні одного пакета незалежно від платформи реалізації (Cisco або VyOS). Натомість GLBP демонструє довший процес переключення між маршрутизаторами, що може впливати на якість обслуговування критичних додатків, особливо в умовах високої чутливості до затримок. Враховуючи результати, VRRP є доцільнішим для застосування в мережах, де пріоритетом є мінімізація простою, тоді як GLBP може бути корисним у сценаріях із необхідністю балансування навантаження між шлюзами.

Висновки

Стаття присвячена дослідженню механізмів управління надійністю та відмовостійкістю в інфокомунікаційних мережах шляхом моделювання та тестування протоколів резервування шлюзів. В роботі виконано поставлені завдання щодо порівняльного аналізу протоколів VRRP та GLBP, їх програмної реалізації у середовищах EVE-NG і Containerlab, а також тестування на фізичних і віртуальних пристроях.

Результати проведених досліджень підтвердили, що протокол VRRP є більш ефективним для швидкого переключення між активним і резервним шлюзами порівняно з GLBP. VRRP забезпечував мінімальні втрати пакетів (1-2 пакети) незалежно від платформи реалізації (Cisco або VyOS), що робить його ідеальним вибором для мереж з високими вимогами до безперервності роботи. Натомість GLBP, попри можливість балансування навантаження через механізм Forwarder, демонстрував значно більші втрати пакетів (9-18 пакетів) і довший час переключення, що обмежує його застосування в критичних середовищах. Загалом, отримані результати дозволяють

сформулювати рекомендації щодо вибору протоколу залежно від конкретних потреб мережі. Якщо головним критерієм є мінімізація простоїв і швидке переключення між шлюзами, то VRRP є оптимальним рішенням. Якщо ж основна мета полягає у рівномірному балансуванні навантаження між кількома вузлами, то використання GLBP може бути виправданим, проте слід враховувати його більшу чутливість до втрат трафіку.

Дослідження виявило, що віртуальне середовище EVE-NG є ефективним інструментом для попереднього тестування та відпрацювання конфігурацій, оскільки воно дозволяє швидко та гнучко змінювати параметри за умови відсутності ризиків для фізичного обладнання. Водночас тестування на реальних пристроях необхідне для отримання точних оцінок продуктивності та врахування апаратних особливостей. Контейнеризоване середовище Containerlab, хоч і є легковаговим, може поступатися у точності відтворення реальних мережних сценаріїв, що важливо враховувати при виборі платформи для тестування.

Перспективи подальших досліджень у цьому науковому напрямку пов'язані з дослідженням можливості GLBP щодо балансування навантаження [20], зокрема різних механізмів, наприклад, round-robin, weighted, host-dependent, а також оптимізації часу перемикання між шлюзами. Це дозволить розробити рекомендації для використання GLBP у великих мережах, де важливе не лише резервування, але й ефективний розподіл трафіку між кількома шлюзами. Також варто розглянути інтеграцію GLBP з іншими протоколами, такими як VRRP, для створення гібридних рішень, що поєднують їхні сильні сторони.

References

- [14] Лемешко, О.В., Єременко, О.С., Невзорова, О.С. (2020), *Потокові моделі та методи маршрутизації в інфокомунікаційних мережах: відмовостійкість, безпека, масштабованість*. Харків: ХНУРЕ, 308 с. doi: <https://doi.org/10.30837/978-966-659-282-1>
- [15] Лемешко, О.В., Єременко, О.С., Євдокименко, М.О., Шаповалова, А.С., Слейман, Б. (2022), *Моделювання та оптимізація процесів безпечної та відмовостійкої маршрутизації в телекомунікаційних мережах: монографія. М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. Харків: ХНУРЕ*, 198 с. doi: <https://doi.org/10.30837/978-966-659-378-1>
- [16] Carthern, C., Wilson, W., Bedwell, R., Rivera, N. (2015), "Introduction to Availability", In: *Cisco Networks*. Apress, Berkeley, CA, pp. 383–406. doi: https://doi.org/10.1007/978-1-4842-0859-5_13
- [17] Cisco Systems (2023), "Understand the Hot Standby Router Protocol Features and Functionality", available at: <https://www.cisco.com/c/en/us/support/docs/ip/hot-standby-router-protocol-hsrp/9234-hsrpguidetoc.html>
- [18] Cisco Systems, "Gateway Load Balancing Protocol Configuration Guide", available at: https://www.cisco.com/en/US/docs/ios/12_2t/12_2t15/feature/guide/ft_glb.html
- [19] Hinden, R., and Deering, S. 2010. *Virtual Router Redundancy Protocol (VRRP)*. RFC 5798, available at: <https://www.rfc-editor.org/rfc/rfc5798>
- [20] Lemeshko, O., Mersni, A., Yeremenko, O., Omowumi, S., Volotka, V., and Al-Dulaimi, A. (2021), "Application Prospects of First Hop Redundancy Protocols for Fault-Tolerant SDN Controllers: A Survey", *2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T)*, pp. 416-420. doi: <https://doi.org/10.1109/picst54195.2021.9772141>
- [21] Єременко, О.С., Мерсні, А. (2020), "Підвищення відмовостійкості елементів сучасних інфокомунікаційних мереж із застосуванням протоколів резервування шлюзу за замовчуванням", *Проблеми телекомунікацій*, №2(27), pp.68-81. doi: <https://doi.org/10.30837/pt.2020.2.06>
- [22] Єременко, О.С., Євдокименко, М.О. (2018), "Огляд теоретичних рішень щодо відмовостійкої маршрутизації в телекомунікаційних мережах", *Проблеми телекомунікацій*, №1(22), pp.25-42. doi: <https://doi.org/10.30837/pt.2018.1.02>
- [23] Saud, N., and Mansour, M. (2023), "Performance Evaluation of First Hop Redundancy Protocols in IPv4 and IPv6 Networks", *2023 IEEE 3rd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA)*, pp. 440-445. doi: <https://doi.org/10.1109/MI-STA57575.2023.10169462>
- [24] Wibowo, F., Yulianto, F., and Nuha, H. (2023), "Performance Comparison Analysis of Virtual Router Redundancy Protocol (VRRP) with Gateway Load Balancing Protocol (GLBP) on a DMVPN Network", *2023*

- 11th International Conference on Information and Communication Technology (ICoICT), pp. 499-504. doi: <https://doi.org/10.1109/ICoICT58202.2023.10262804>
- [25] Julia, I., Suseno, H., Wardhani, L., Khairani, D., Hulliyah, K., and Muharram, A. (2020), "Performance Evaluation of First Hop Redundancy Protocol (FHRP) on VRRP, HSRP, GLBP with Routing Protocol BGP and EIGRP", 2020 8th International Conference on Cyber and IT Service Management (CITSM), pp. 1-5. doi: <https://doi.org/10.1109/CITSM50537.2020.9268799>
- [26] Hamied, M. (2022), "Gateway Redundancy Protocols in the Smart Grid", 2022 2nd International Conference on Computing and Information Technology (ICCIT), pp. 186-190. doi: <https://doi.org/10.1109/ICCIT52419.2022.9711652>
- [27] Harahus, M., Cavojský, M., Bugár, G., & Pleva, M., 2023. Interactive Network Learning: An Assessment of EVE-NG Platform in Educational Settings. *Acta Electrotechnica et Informatica*, 23, pp. 3-9. doi: <https://doi.org/10.2478/aei-2023-0011>
- [28] Packetswitch (2024), "Containerlab - Creating Network Labs Can't Be Any Easier", available at: <https://www.packetswitch.co.uk/containerlabs-intro/>
- [29] Ansible Community Documentation (2025), "Ansible for Network Automation", available at: <https://docs.ansible.com/ansible/latest/network/index.html>
- [30] MikroTik Documentation (2025), available at: https://wiki.mikrotik.com/Main_Page
- [31] Containerlab (2025), available at: <https://containerlab.dev/>
- [32] Savchenko, R. (2025), "Containerlab configs". Zenodo. doi: <https://doi.org/10.5281/zenodo.14916309>
- [33] Lemeshko, O., Yeremenko, O., Mersni, A., Yevdokymenko, M., Persikov, M., and Kruhlova, A. (2023), "Analysis of Proactive Models of Fault-Tolerant Routing under Load Balancing and Border Routers Availability" 2023 17th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), 1, pp. 28-31. doi: <https://doi.org/10.1109/CADSM58174.2023.10076525>

STUDY OF RELIABILITY AND FAULT TOLERANCE MANAGEMENT IN INFORMATION AND COMMUNICATION NETWORKS: MODELING AND TESTING OF DEFAULT GATEWAY REDUNDANCY PROTOCOLS

Oleksandra Yeremenko, Roman Savchenko, Kyrylo Yakovenko, Serhii Shestopalov

Kharkiv National University of Radio Electronics, Nauky Ave., 14, 61166, Kharkiv, Ukraine

The article is devoted to researching reliability and fault tolerance management mechanisms in infocommunication networks, focusing on modeling and testing of gateway redundancy protocols. The work considers the VRRP (Virtual Router Redundancy Protocol) and GLBP (Gateway Load Balancing Protocol) protocols, which ensure continuous network operation in the event of a primary gateway failure. The purpose of the study is to compare these protocols using the virtual environments EVE-NG and Containerlab, as well as to test them on physical and virtual devices. The article analyzes the characteristics and mechanisms of both protocols, the software implementation, and the configuration of the protocols on Cisco devices in emulation environments and tests their effectiveness in the face of gateway failures. The results showed that VRRP provides minimal packet loss and fast switching between master and backup gateways, making it optimal for networks with high requirements for connection continuity. Instead, GLBP provides load balancing between gateways but is accompanied by higher packet loss and longer switching times, which limits its use in critical networks. In particular, testing in virtual environments has shown that when using virtualized links and resources, VRRP delays and packet loss are significantly lower than GLBP. Therefore, testing in virtual environments is useful for preliminary analysis of configurations, although testing on real devices is necessary to determine the protocols' effectiveness accurately. We also consider the possibilities of using GLBP in large networks to provide not only redundancy but also optimize traffic balancing. Based on the results obtained, we formulate suggestions for future research, including optimizing switching times between gateways and developing hybrid solutions that combine the advantages of both protocols, VRRP and GLBP, to improve traffic management and redundancy and create more flexible and fault-tolerant infrastructures.

Keywords: reliability, fault tolerance, default gateway, VRRP, GLBP, modeling, testing