

ОРГАНІЗАЦІЯ ВІДДАЛЕНОЇ ВЗАЄМОДІЇ СПІВРОБІТНИКІВ З ІНФОРМАЦІЙНОЮ ІНФРАСТРУКТУРОЮ ПІДПРИЄМСТВА

В даний час гострою проблемою є питання забезпечення віддаленого підключення іноземних установ і персоналу підприємства до своєї інформаційної мережі через мережу загального користування Інтернет, в той час як інші користувачі мережі загального користування не можуть отримати доступ до інформаційних ресурсів всередині підприємства. У роботі описано послідовні кроки налаштування сервера на базі Ubuntu 20.04 з використанням StrongSwan, демона IPsec, що підтримує як IKEv1, так і IKEv2. Під час процесу налаштування був створений 4096-бітний RSA ключ для підпису кореневого ЦС. Після цього було згенеровано закритий ключ VPN-сервера, який був підписаний сертифікатом VPN-сервера з використанням ключа центру сертифікації. Також було проведено налаштування IPsec, внісши необхідні зміни до конфігураційного файлу. Далі в роботі описано процес налаштування аутентифікації, включаючи настройку ідентифікатора (логіна) і пароля. Останнім етапом було налаштовано брандмауер, дозволено доступ для OpenSSH для віддаленого управління сервером, вказано стандартні порти IPsec та встановлено настройки переадресації. Цей тип зв'язку забезпечує високий рівень безпеки, оскільки сервер запитує у клієнта сертифікат та дані автентифікації. Ще однією перевагою є використання шифрування, яке сумісне з усіма платформами. Таким чином, забезпечується зашифроване та безпечне підключення до мережі, що має важливе значення для віддалених працівників, оскільки воно дає їм прямий доступ до ресурсів організації, не перебуваючи в офісі. Користувачі можуть підключатися до мережі з різних регіонів у всьому світі за допомогою своїх пристроїв.

Ключові слова: Інформаційна інфраструктура, мережі, з'єднання, захист, безпека, віддалений доступ.

Вступ

Сучасний розвиток суспільства передбачає повсякденне використання комунікаційних мереж та їх сервісів. Кожна компанія чи установа створює власну інформаційну інфраструктуру. У загальному випадку інформаційна інфраструктура являє собою взаємопов'язану сукупність мереж, служб передачі даних та інтернет-сервісів, призначених для надання єдиного захищеного мережевого простору обмеженому колу користувачів всередині підприємства. В даний час вирішується питання забезпечення віддаленого підключення іноземних установ і персоналу підприємства до своєї інформаційної мережі через мережу загального користування через мережу Інтернет, в той час як інші користувачі мережі загального користування не можуть отримати доступ до інформаційних ресурсів всередині підприємства. Вирішення цього питання особливо важливе для нашої країни під час війни.

Постановка проблеми

Основним призначенням інформаційної інфраструктури підприємств є підвищення ефективності роботи його співробітників. Мережі підтримують цілий ряд додатків і в основному мають такі основні характеристики: надійність; завадозахищеність; масштабованість; безпека. Основні функції мережі: клієнт-серверна архітектура; ефективне використання простору IP-адрес; ефективна підтримка спектру додатків, необхідних для роботи співробітників; забезпечення безпеки мережі при виникненні різних внутрішніх і зовнішніх загроз (мережеві атаки, такі як флуд MAC-адрес і DHCP-спуфінг і т.д.). У зв'язку з ситуацією, що склалася в країні, пов'язаною з війною, постає нове завдання – забезпечити можливість безпечного віддаленого доступу працівників до інформаційної інфраструктури підприємства. Щоб вирішити цю проблему, розглянемо перспективні напрямки.

Аналіз публікацій

Питання організації ефективного віддаленого доступу вже досліджувалися у численних публікаціях.

У дослідженні [1] розглянуто кілька можливих способів організації дистанційного доступу до навчальної лабораторії. Одна з архітектур полягає в перенаправленні окремих портів на один вузол локальної мережі, який має різноманітні інформаційні, апаратні та програмні ресурси. Інший варіант передбачає використання декількох вузлів, серед яких - сервер віддаленого робочого столу, web-сервер та інші ресурси з окремими IP адресами. У

роботі проаналізовано основні структурні складові системи віддаленого доступу через VPN і наведено рекомендації щодо їх налаштування. В публікації [2] проведено аналіз принципів надання дистанційного доступу через VPN, який був створений для забезпечення безпечного доступу філіям до програм організації. VPN забезпечує зашифроване та безпечне з'єднання з мережею. Особливості налаштування дистанційного доступу, які є важливими для віддалених працівників, були визначені, оскільки це дає їм прямий доступ до ресурсів організації, незалежно від місця перебування. Досліджено, що метод дистанційної роботи, що використовується багатьма компаніями, має свої переваги, але при цьому супроводжується появою нових ризиків, які можуть негативно вплинути на всю компанію.

У статті [3] описано сучасні методи та засоби створення віртуальних приватних мереж, де детально проаналізовано способи їх побудови через апаратно-програмні засоби на конкретному прикладі - приватної віртуальної мережі на базі CISCO FlexVPN. Для цієї реалізації використовувався протокол обміну ключами IKEv2 для забезпечення безпеки у віртуальних мережах. У результаті моделювання була створена віртуальна приватна мережа для корпорації, де наявні кілька захищених каналів зв'язку між структурними підрозділами, а також реалізований віддалений доступ для домашніх користувачів за допомогою технології Cisco AnyConnect. У дослідженні [4] був проведений експеримент для визначення ролі автентифікації користувачів. Цей експеримент підтвердив ефективність процесу двофакторної автентифікації (2FA) при віддаленому доступі, проте також виявив можливі вразливості віддалених пристроїв, незважаючи на забезпечену безпеку 2FA. Це може представляти серйозні ризики для організації. В результаті було рекомендовано різні підходи для підвищення безпеки зв'язку віддаленого доступу, особливо в контексті автентифікації віддалених користувачів.

Разом з тим, слід відзначити, що не зважаючи на значну кількість публікацій, допоки що немає єдиного ефективного способу організації віддаленого доступу працівників до ресурсів компанії.

Мета і задачі дослідження

Реалізація заходів щодо забезпечення безпечної взаємодії віддалених співробітників з інформаційною структурою корпоративного підприємства.

Основна частина

Потенційними загрозами конфіденційності, цілісності, доступності та спостережуваності інформації в АС (автоматизованих системах) є: 1) аварії, стихійні лиха та інші форс-мажорні обставини; 2) руйнівний вплив на обладнання та носії АС через електромережу; 3) збої в роботі АС і знищення інформації на носіях під впливом зовнішнього електромагнітного випромінювання; 4) доступ до інформації, що надається користувачами АС за допомогою стандартного програмного та апаратного забезпечення, що знаходиться поза їх повноваженнями; 5) неконтрольований доступ не-АС користувачів до апаратних та інформаційних ресурсів автоматизованої системи; 6) ненавмисні (помилкові, випадкові, необережні, без злого умислу або корисливих намірів) дії користувачів, а також співробітників, відповідальних за адміністрування, обслуговування програмно-технічних засобів АС і засобів захисту в процесі експлуатації АС; 7) навмисні (в корисливих цілях, під примусом сторонніх, зі злим умислом і т.д.) дії користувачів АС; 8) прояви програмних помилок, вірусних дій, збоїв і несправностей апаратних засобів АС.

Створення захищеного інформаційного каналу

Налаштуємо VPN-сервер IKEv2 за допомогою StrongSwan на Ubuntu 20.04. Віртуальна приватна мережа, або VPN, дає змогу надійно шифрувати трафік під час його проходження через ненадійні мережі, як-от у кав'ярні, на конференції чи в аеропорту [5]. Internet Key Exchange v2, або IKEv2, – це протокол, який забезпечує пряме тунелювання IPsec між сервером і клієнтом [6]. У реалізаціях IKEv2 VPN IPsec забезпечує шифрування мережевого трафіку [8]. IKEv2 підтримується на деяких платформах (OS X 10.11+, iOS 9.1+ і Windows 10) без додаткових програм, і він досить плавно справляється з проблемами клієнтів. У нашому

випадку ми налаштуємо VPN-сервер IKEv2 за допомогою StrongSwan на сервері Ubuntu 20.04. Далі ми розглянемо способи підключення до нього за допомогою найпоширеніших клієнтів Windows і Android.

1) Початкове налаштування сервера за допомогою Ubuntu.

Так як для свого дослідження було використано віртуальний сервер, то спосіб підключення до нього – SSH з закритим ключем для аутентифікації.

Тип ключа: RSA

Кількість бітів у згенерованому ключі: 2048

Програмне забезпечення для генерації ключів: PuTTY Key Generator (рис. 1).

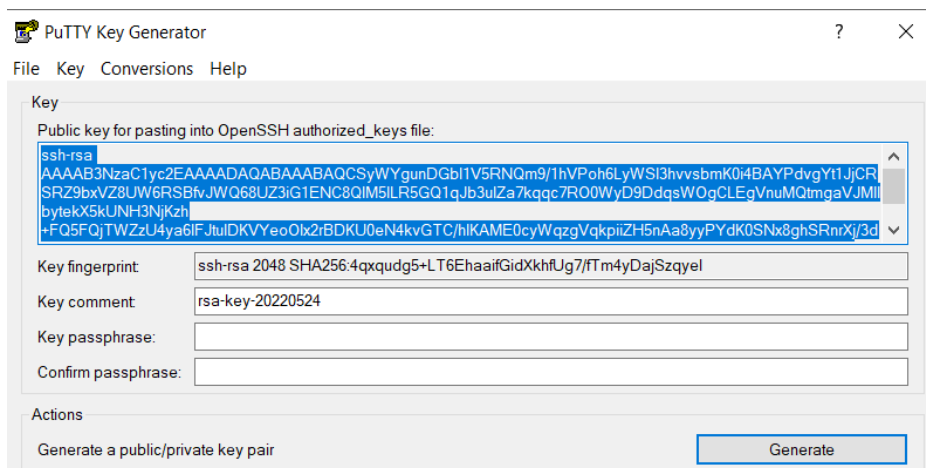


Рис. 1. Приклад генерації ключа

Ключ паролльної фрази знадобиться в подальшому, тому вкажіть його і не забудьте.

2) Налаштування PuTTY

Додайте IP-адресу сервера та деталі підключення. На екрані конфігурації PuTTY заповніть поле Host Name (або IP-адреса) IP-адресою вашого сервера. Переконайтеся, що порт встановлено на 22 і вибрано тип з'єднання SSH (рис. 2).

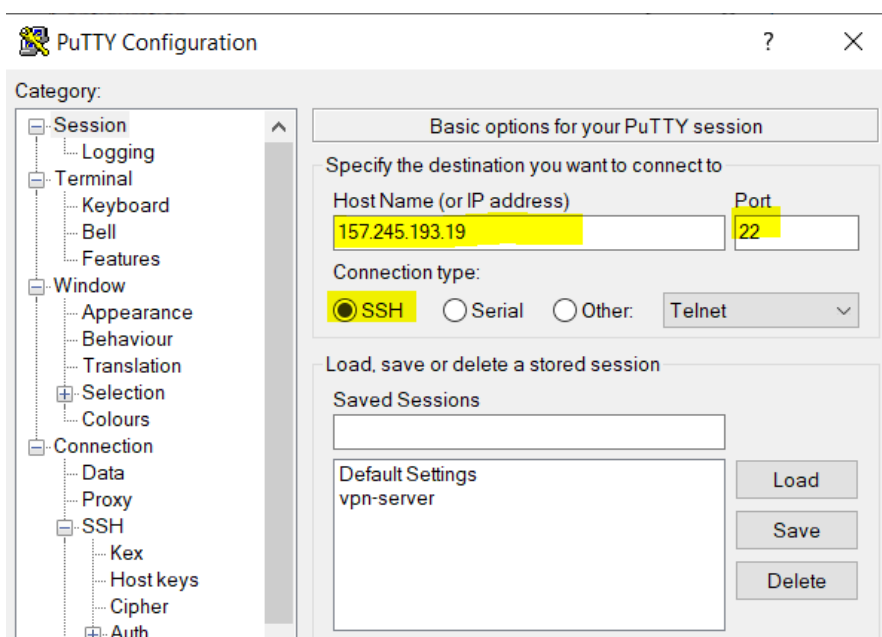


Рис. 2. Налаштування сесії PuTTY

Перевірте протокол SSH.

Далі натисніть SSH на лівій бічній панелі (у розділі «Підключення»). Переконайтеся, що вибрано 2 для версії протоколу SSH (рис. 3).

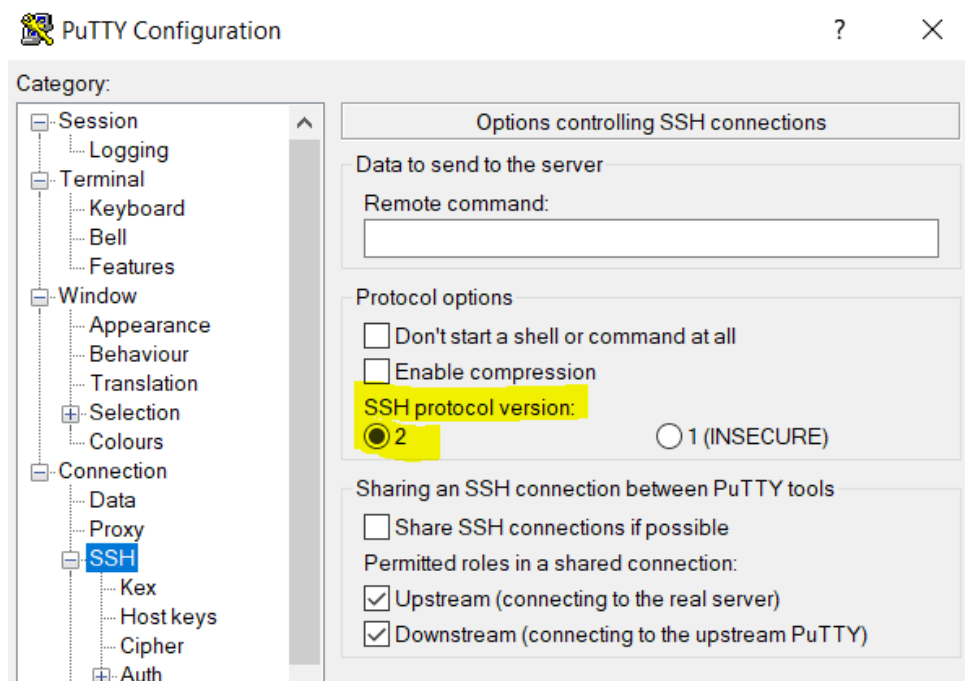


Рис. 3. Налаштування SSH PuTTY

Визначення ключа SSH

У блоці Файл закритого ключа для автентифікації натисніть кнопку Огляд (рис. 4).

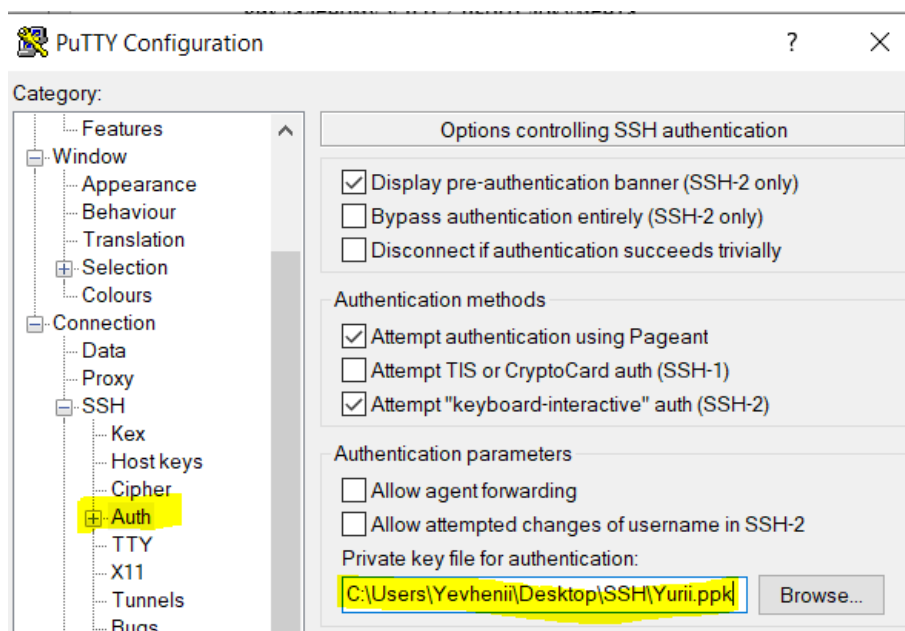


Рис. 4. Додавання закритого ключа PuTTY

Додайте ім'я користувача.

Далі в підзаголовку Connection в розділі Конфігурація даних введемо ім'я користувача нашого сервера в поле Ім'я користувача Auto-login (рис. 5).

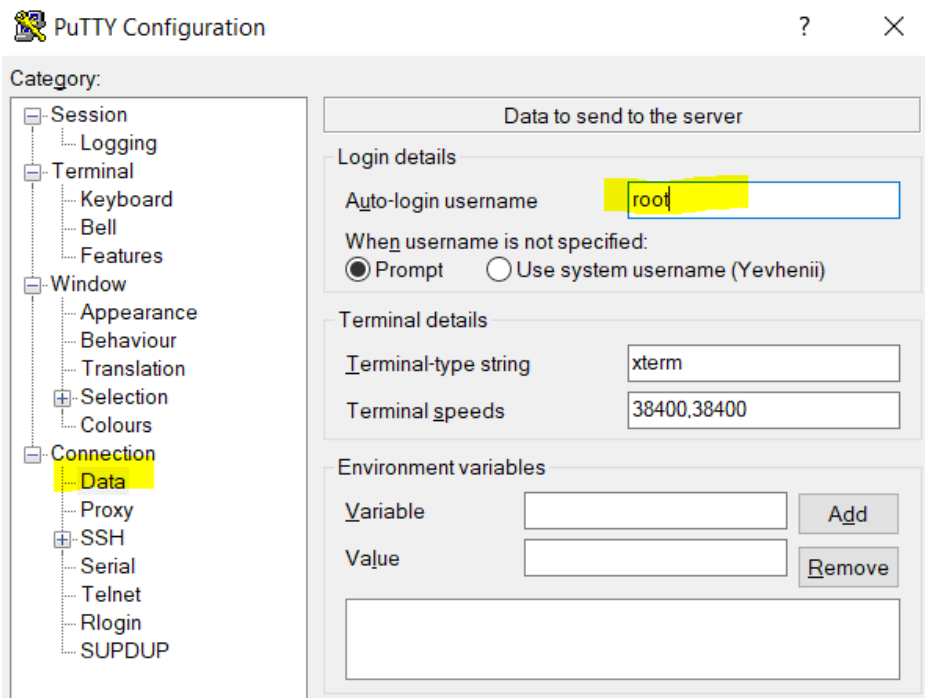


Рис. 5. Додавання логіна PuTTY

І тепер ми можемо підключитися до нашого сервера за допомогою PuTTY (рис. 6).

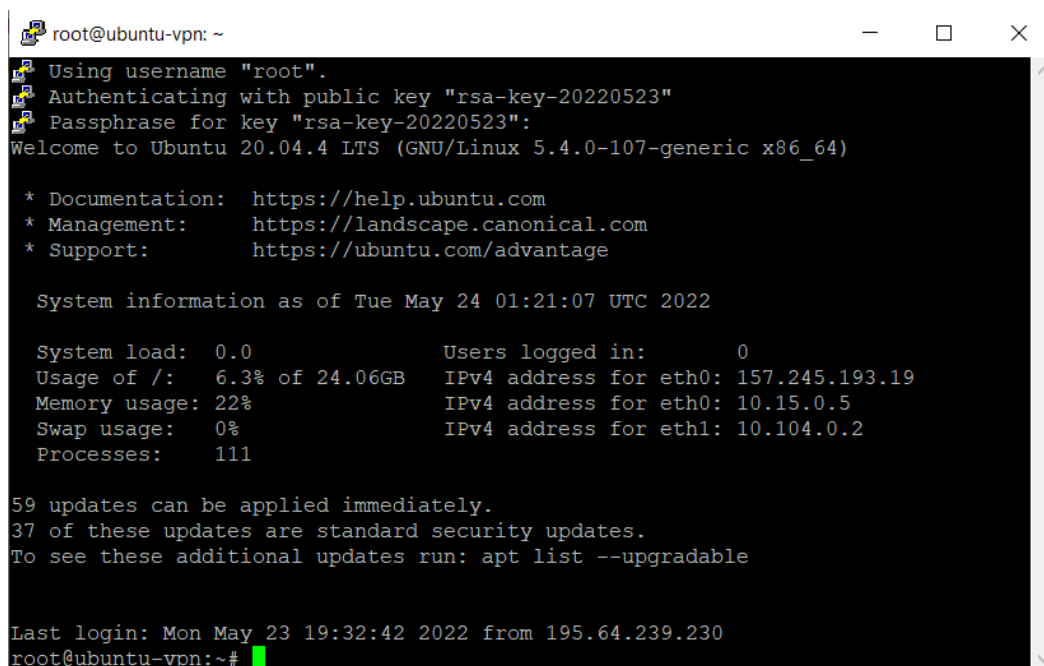


Рис. 6. Підключення до сервера за допомогою PuTTY

3) Сервер конфігурації.

Крок 1 – Встановлення StrongSwan

По-перше, ми встановимо StrongSwan, демон IPsec з відкритим вихідним кодом, який ми налаштуємо як наш VPN-сервер. Ми також встановимо компонент інфраструктури відкритих ключів (PKI), щоб створити центр сертифікації (CA) для надання облікових даних для нашої інфраструктури.

Почніть з оновлення локального кешу пакунків:

```
$ sudo apt update
```

Потім встановіть програмне забезпечення, ввівши команду:

```
$ sudo apt install strongswan strongswan-pki libcharon-extra-
plugins libcharon-extauth-plugins libstrongswan-extra-plugins
```

Додатковий пакунок `libcharon-extauth-plugins` використовується для забезпечення того, щоб різні клієнти могли розпізнаватися на вашому сервері за допомогою спільного імені користувача та парольної фрази. Пакунок `libstrongswan-extra-plugins` включено таким чином, що Strongswan підтримує набори шифрів з еліптичною кривою, які використовують криптографічний набір Curve25519 [7].

Крок 2 – Створення центру сертифікації

Сервер IKEv2 вимагає сертифіката, щоб ідентифікувати себе для клієнтів. Щоб допомогти створити необхідний сертифікат, пакет `strongswan-pki` постачається з утилітою під назвою `pki` для створення центру сертифікації та сертифікатів сервера.

Для початку давайте створимо кілька каталогів для зберігання всіх активів, над якими ми будемо працювати. Структура каталогів збігається з деякими каталогами в `/etc/ipsec.d`, куди ми в кінцевому підсумку перемістимо всі створені нами елементи:

Тепер, коли все встановлено, перейдемо до створення наших сертифікатів.

```
$ mkdir -p ~/pki/{cacerts,certs,private}
```

Потім заблокуємо дозволи, щоб інші користувачі не могли бачити наші особисті файли.

```
$ chmod 700 ~/pki
```

Тепер, коли у нас є структура каталогів для зберігання всього, ми можемо згенерувати кореневий ключ. Це буде 4096-бітний ключ RSA, який буде використано для підписання нашого кореневого центру сертифікації.

Виконаємо такі команди, щоб згенерувати ключ:

```
$ pki --gen --type rsa --size 4096 --outform pem >
~/pki/private/ca-key.pem
```

Після цього ми можемо перейти до створення нашого центру кореневого сертифіката, використовуючи ключ, який ми щойно згенерували для підпису кореневого сертифіката:

```
$ pki --self --ca --lifetime 3650 --in ~/pki/private/ca-
key.pem \
$ --type rsa --dn "CN=VPN root CA" --outform pem >
~/pki/cacerts/ca-cert.pem
```

Прапорець `--lifetime 3650` використовується для забезпечення того, щоб кореневий сертифікат центру сертифікації був дійсним протягом 10 років. Кореневий сертифікат для органу влади, як правило, не змінюється, оскільки його потрібно було б розповсюджувати на всі сервери та клієнти, які на нього покладаються, тому 10 років є безпечним значенням терміну дії за замовчуванням.

Тепер, коли ми запустили наш кореневий центр сертифікації, ми можемо створити сертифікат, який використовуватиме VPN-сервер.

Крок 3 – Генерація сертифіката для VPN-сервера

Тепер ми створимо сертифікат і ключ для VPN-сервера. Цей сертифікат дозволить клієнту перевірити справжність сервера за допомогою щойно згенерованого нами сертифіката ЦС.

Спочатку створіть приватний ключ для VPN-сервера за допомогою наступної команди:

```
pki --gen --type rsa --size 4096 --outform pem >
~/pki/private/server-key.pem
```

Тепер створіть і підпишіть сертифікат VPN-сервера за допомогою ключа центру сертифікації, який ви створили на попередньому кроці. Виконайте наступну команду, але змініть поле «Загальне ім'я» (CN) і «Альтернативне ім'я суб'єкта» (SAN) на DNS-ім'я або IP-адресу вашого VPN-сервера:

```
$ pki --pub --in ~/pki/private/server-key.pem --type rsa \
| pki --issue --lifetime 1825 \
```

```

--cacert ~/pki/cacerts/ca-cert.pem \
--cakey ~/pki/private/ca-key.pem \
--dn "CN=157.245.193.19" --san @157.245.193.19 --san
157.245.193.19 \
--flag serverAuth --flag ikeIntermediate --outform pem \
> ~/pki/certs/server-cert.pem

```

Параметр `--flag serverAuth` використовується для позначення того, що сертифікат буде використано явним чином для розпізнавання сервера до того, як буде встановлено зашифрований тунель. Параметр `--flag ikeIntermediate` використовується для підтримки старіших клієнтів macOS.

Тепер, коли ми згенерували всі файли TLS/SSL, необхідні StrongSwan, ми можемо перемістити файли на місце в каталозі `/etc/ipsec.d`, ввівши команду:

```
$ sudo cp -r ~/pki/* /etc/ipsec.d/
```

На цьому кроці ми створили пару сертифікатів, яка буде використовуватися для захисту зв'язку між клієнтом і сервером. Ми також підписали сертифікати ключем ЦС, тому клієнт зможе перевірити справжність VPN-сервера за допомогою сертифіката ЦС. Коли всі ці сертифікати готові, ми перейдемо до налаштування програмного забезпечення.

Крок 4 — Налаштування StrongSwan

У StrongSwan є конфігураційний файл за замовчуванням з деякими прикладами, але більшу частину конфігурації нам доведеться робити самостійно. Давайте створимо резервну копію файлу для довідки, перш ніж почати з нуля:

```
$ sudo mv /etc/ipsec.conf{,.original}
```

Створіть і відкрийте новий порожній файл конфігурації за допомогою текстового редактора. Тут ми будемо використовувати `nano`:

```
sudo nano /etc/ipsec.conf
```

По-перше, ми скажемо StrongSwan реєструвати статуси демонів для налагодження та дозволяти дублювання з'єднань.

Налаштування конфігурації

```
charondebug="ike 1, knl 1, cfg 0"
uniqueids=no
```

Потім ми створимо розділ конфігурації для нашого VPN. Ми також накажемо StrongSwan створювати тунелі IKEv2 VPN і автоматично завантажувати цей розділ конфігурації під час запуску. Додайте до файлу такі рядки:

```

. . .
conn ikev2-vpn
    auto=add
    compress=no
    type=tunnel
    keyexchange=ikev2
    fragmentation=yes
    forceencaps=yes

```

Ми також налаштуємо виявлення мертвих вузлів, щоб очистити будь-які «завислі» з'єднання у випадку, якщо `clie dpdaction=clear`

```
dpddelay=300 s
```

```
rekey=nont
```

несподівано розриває зв'язок. Додайте такі рядки:

Далі ми налаштуємо параметри IPsec в «лівій» частині сервера. Кожен із наведених нижче параметрів гарантує, що сервер налаштовано на прийом з'єднань від клієнтів і правильну автентифікацію. Ми додамо кожен з цих опцій до файлу `/etc/ipsec.conf` після того, як ознайомимося з тим, що це таке і чому вони використовуються:

`left=%any` Значення `%any` гарантує, що сервер використовуватиме мережевий інтерфейс, у якому він отримує вхідні з'єднання, для подальшого зв'язку з клієнтами.

Наприклад, якщо ви підключаєте клієнта через приватну мережу, сервер використовуватиме приватну IP-адресу, за якою він отримує трафік для решти з'єднання.

`leftid=@server_domain_or_IP` Цей параметр керує назвою, яку сервер надає клієнтам. У поєднанні з наступним параметром `leftcert`, параметр `leftid` гарантує, що налаштоване ім'я сервера та визначне ім'я (DN), яке міститься у загальнодоступному сертифікаті, збігається.

`leftcert=server-cert.pem` Цей параметр є шляхом до загальнодоступного сертифіката для сервера, який ви налаштували на кроці 3. Без нього сервер не зможе пройти аутентифікацію у клієнтів або завершити переговори щодо налаштування IKEv2.

`leftsendcert=always` Значення `always` гарантує, що будь-який клієнт, який підключається до сервера, завжди отримуватиме копію публічного сертифіката сервера як частину початкового налаштування з'єднання.

`leftsubnet=0.0.0.0/0` Остання опція «лівої» сторони, яку ви додасте, повідомляє клієнтам про підмережі, які доступні за сервером. У цьому випадку `0.0.0.0/0` використовується для представлення всього набору IPv4-адрес, що означає, що сервер за замовчуванням буде вказувати клієнтам надсилати весь свій трафік через VPN.

Тепер, коли ви ознайомилися з кожним з відповідних варіантів «лівої» сторони, додайте їх усі до файлу таким чином:

```
left=%any
leftid=157.245.193.19
leftcert=server-cert.pem
leftsendcert= always
leftsubnet=0.0.0.0/0
```

Далі ми можемо налаштувати параметри IPsec у «правій» частині клієнта. Кожен із наведених нижче параметрів повідомляє серверу, як приймати з'єднання від клієнтів, як клієнти повинні автентифікуватися на сервері, а також діапазони приватних IP-адрес і DNS-сервери, які використовуватимуть клієнти. Давайте додамо кожен з цих опцій до файлу `/etc/ipsec.conf`, як тільки ми ознайомимось з тим, що це таке і для чого вони використовуються:

`right=%any` Параметр `%any` у правій частині з'єднання наказує серверу приймати вхідні з'єднання від будь-якого віддаленого клієнта.

`rightid=%any` Цей параметр гарантує, що сервер не відхилить з'єднання від клієнтів, які надають ідентифікаційні дані до встановлення зашифрованого тунелю.

`rightauth=eap-mschapv2` За допомогою цього параметра можна налаштувати метод розпізнавання, який клієнти використовуватимуть для розпізнавання на сервері. `eap-mschapv2` використовується тут для широкої сумісності для підтримки таких клієнтів, як пристрої Windows, macOS та Android.

`rightsourcexp=10.10.10.0/24` Цей параметр наказує серверу призначити приватні IP-адреси клієнтам із вказаного пулу IP-адрес `10.10.10.0/24`.

`rightdns=8.8.8.8,8.8.4.4` Ці IP-адреси є загальнодоступними DNS-резольверами Google. Їх можна змінити, щоб використовувати інші публічні розв'язувачі, резольвери VPN-сервера або будь-який інший розв'язувач, до якого можуть отримати доступ клієнти.

`rightsendcert=never` Цей параметр вказує серверу, що клієнтам не потрібно надсилати сертифікат для автентифікації.

Тепер, коли ви ознайомилися з необхідними опціями «правильної» сторони для VPN, додайте наступні рядки до `/etc/ipsec.conf`:

```
. . .
right=%any
rightid=%any
rightauth=eap-mschapv2
```



```
rightsourcetype=10.10.10.0/24
rightdns=8.8.8.8,8.8.4.4
rightsendcert= never
```

Тепер ми скажемо StrongSwan запитувати у клієнта облікові дані користувача при підключенні:

```
eap_identity=%identity
```

Нарешті, додайте наступні рядки для підтримки клієнтів Linux, Windows, macOS, iOS та Android. Ці рядки визначають різні алгоритми обміну ключами, хешування, аутентифікації та шифрування (зазвичай звані Cipher Suites), які StrongSwan дозволить використовувати різним клієнтам:

```
ike=chacha20poly1305-sha512-curve25519-
prfsha512,aes256gcm16-sha384-prfsha384-ecp384,aes256-sha1-
modp1024,aes128-sha1-modp1024,3des-sha1-modp1024!
ESP=Chacha20Poly1305-SHA512,AES256GCM16-EC384,AES256-
SHA256,AES256-SHA1,3DES-SHA1!
```

Кожен підтримуваний набір шифрів відокремлюється від інших комами. Наприклад, chacha20poly1305-sha512-curve25519-prfsha512 – це один комплект, а aes256gcm16-sha384-prfsha384-ecp384 – інший. Набори шифрів, перелічені тут, вибираються для забезпечення найширшого діапазону сумісності з клієнтами Windows, macOS, iOS, Android і Linux [7].

Тепер, коли ми налаштували параметри VPN, давайте перейдемо до створення облікового запису, щоб наші користувачі могли підключатися до сервера.

Крок 5 – Налаштування VPN-аутентифікації

Наш VPN-сервер тепер налаштований на прийом клієнтських з'єднань, але ми ще не налаштували жодних облікових даних. Нам потрібно буде налаштувати кілька речей у спеціальному файлі налаштувань під назвою ipsec.secrets:

Нам потрібно сказати StrongSwan, де знайти приватний ключ для нашого сертифіката сервера, щоб сервер міг автентифікуватися перед клієнтами.

Нам також потрібно налаштувати список користувачів, яким буде дозволено підключатися до VPN.

Відкриємо файл секретів для редагування:

```
$ sudo nano /etc/ipsec.secrets
```

Для початку ми розповімо StrongSwan, де знайти наш приватний ключ і як його розібрати.

```
: RSA "server-key.pem"
```

Потім ми визначимо облікові дані користувача. Ви можете скласти будь-яку комбінацію імені користувача або пароля, яка вам подобається:

```
our_username : EAP "our_password"
```

наприклад:

```
Читець : НВД "12345678"
```

Збережіть і закрийте файл. Тепер, коли ми закінчили роботу з параметрами VPN, ми перезапустимо службу VPN, щоб застосувати нашу конфігурацію:

```
$ sudo systemctl restart strongswan-starter
```

Крок 6 – Налаштування брандмауера та переадресації IP-адрес ядра

Після завершення налаштування StrongSwan нам потрібно налаштувати брандмауер, щоб пропускати та перенаправляти VPN-трафік.

```
$ sudo ufw дозволити OpenSSH
```

Потім увімкніть брандмауер, ввівши команду:

```
$ sudo ufw enable
```

Потім додайте правило, яке дозволяє UDP-трафік стандартним портам IPsec, 500 і 4500:

```
$ sudo ufw дозволити 500,4500/udp
```

Далі ми відкриємо один з конфігураційних файлів UFW, щоб додати кілька низькорівневих політик для маршрутизації та пересилання пакетів IPsec. Однак, перш ніж ми зможемо це зробити, нам потрібно з'ясувати, який мережевий інтерфейс на нашому сервері використовується для доступу до Інтернету. Знайдіть цей інтерфейс за допомогою запиту пристрою, пов'язаного з маршрутом за замовчуванням:

```
$ ip route show за замовчуванням
```

Результат команди показує інтерфейс з іменем eth0

Коли у нас є наш загальнодоступний мережевий інтерфейс, відкриваємо файл /etc/ufw/before.rules у текстовому редакторі. Правила в цьому файлі додаються до брандмауера перед іншими звичайними правилами введення та виведення. Вони використовуються для налаштування трансляції мережевих адрес (NAT), щоб сервер міг правильно маршрутизувати з'єднання з клієнтами та Інтернетом.

```
$ sudo nano /etc/ufw/before.rules
```

У верхній частині файлу (перед рядком *filter) додайте наступний блок налаштувань. Змініть кожен екземпляр eth0 у наведеній вище конфігурації, щоб він відповідав назві інтерфейсу, яку ви знайшли, з ip route. Рядки *nat створюють правила, щоб брандмауер міг правильно маршрутизувати та маніпулювати трафіком між VPN-клієнтами та Інтернетом. Рядок *mangle регулює максимальний розмір сегмента пакета, щоб запобігти потенційним проблемам з певними VPN-клієнтами:

```
*nat
-A POSTROUTING -s 10.10.10.0/24 -o eth0 -m policy --pol ipsec
--dir out -j ПРИЙНЯТИ
-A POSTROUTING -s 10.10.10.0/24 -o eth0 -j МАСКАРАД
COMMIT
*mangle
-A FORWARD --match policy --pol ipsec --dir in -s
10.10.10.0/24 -o eth0 -p tcp -m tcp --tcp-flags SYN,RST SYN -m
tcpmss --mss 1361:1536 -j TCPMSS --set-mss 1360
COMMIT
*filter
:ufw-before-input - [0:0]
:ufw-before-output - [0:0]
:ufw-before-forward - [0:0]
:ufw-not-local - [0:0]
.
.
.
```

Далі, після рядків *filter та chain definition, додаємо ще один блок конфігурації:

```
-A ufw-before-forward --match policy --pol ipsec --dir in --
proto esp -s 10.10.10.0/24 -j АСЦЕРТ
-A ufw-before-forward --match policy --pol ipsec --dir out --
proto esp -d 10.10.10.0/24 -j АСЦЕРТ
```

Ці рядки повідомляють брандмауеру перенаправити трафік ESP (Encapsulating Security Payload), щоб клієнти VPN могли підключитися. ESP забезпечує додатковий захист наших VPN-пакетів, оскільки вони перетинають ненадійні мережі.

Перед перезапуском брандмауера ми змінимо деякі параметри мережевого ядра, щоб дозволити маршрутизацію з одного інтерфейсу на інший. Файл, який керує цими параметрами, називається /etc/ufw/sysctl.conf. Нам потрібно буде налаштувати кілька речей у файлі.

Спочатку потрібно ввімкнути переадресацію пакетів IPv4, щоб трафік міг переміщатися між VPN і загальнодоступними мережевими інтерфейсами на сервері. Далі ми вимкнемо виявлення Path MTU, щоб запобігти проблемам фрагментації пакетів. Нарешті, ми не прийматимемо перенаправлення ICMP і не надсилатимемо перенаправлення ICMP, щоб запобігти атакам типу "людина посередині" [9–11].

Відкрийте файл конфігурації параметрів ядра UFW за допомогою текстового редактора:
`$ sudo nano /etc/ufw/sysctl.conf`

Тепер додайте наступне налаштування `net/ipv4/ip_forward=1` в кінці файлу, щоб дозволити пересилання пакетів між інтерфейсами:

```
NET/IPv4/ip_forward=1
```

Наступний блок відправки і отримання пакетів ридиректу ICMP шляхом додавання наступних рядків в кінець файлу:

```
net/ipv4/conf/all/accept_redirects=0
```

```
net/ipv4/conf/all/send_redirects=0
```

Нарешті, вимкніть виявлення шляху MTU, додавши цей рядок в кінець файлу:

```
NET/IPv4/ip_no_pmtu_disc=1
```

Збережіть файл. Тепер ми можемо ввімкнути всі наші зміни, вимкнувши та знову ввімкнувши брандмауер, оскільки UFW застосовує ці налаштування щоразу, коли він перезапускається:

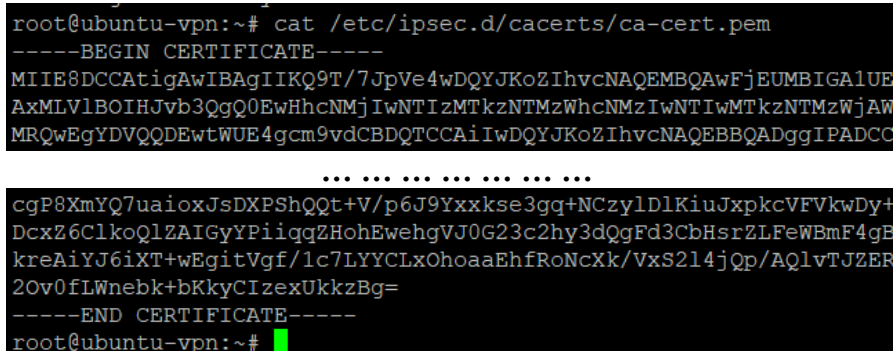
```
$ sudo ufw disable
```

```
$ sudo ufw enable
```

Крок 7 – Тестування VPN-з'єднання на клієнті

Тепер, коли ми все налаштували, настав час спробувати. По-перше, нам потрібно скопіювати створений нами сертифікат ЦС і встановити його на наш клієнтський пристрій, який буде підключатися до VPN. Найпростіший спосіб зробити це - увійти на наш сервер і вивести вміст файлу сертифіката (рис. 7):

```
$ cat /etc/ipsec.d/cacerts/ca-cert.pem
```



```
root@ubuntu-vpn:~# cat /etc/ipsec.d/cacerts/ca-cert.pem
-----BEGIN CERTIFICATE-----
MIIE8DCCAtigAwIBAgIIKQ9T/7JpVe4wDQYJKoZIhvcNAQEMBQAwFjEUMBIGAlUE
AxMLV1BOIHJvb3QgQ0EwHhcNMjIwNTIzMTkzNTMzWWhcNMzIwNTIwMTkzNTMzWjAW
MRQwEgYDVQQDEwtWUE4gcm9vdCBDQ0TCCEAiIwDQYJKoZIhvcNAQEBBQADggIPADCC
... ..
cgP8XmYQ7uaioxJsDXPShQQt+V/p6J9Yxxkse3gq+NCzylDlKiuJxpkcVFVkwDy+
DcxZ6ClkoQlZAIGyYPiigqZHohEwehgVJ0G23c2hy3dQgFd3CbHsrZLFeWBmF4gB
kreAiYJ6iXT+wEgitVgf/1c7LYYCLxOhoaaEhfRoNcXk/Vxs2l4jQp/AQ1vTJZER
2Ov0fLWnebk+bKkyCIzexUkzBg=
-----END CERTIFICATE-----
root@ubuntu-vpn:~#
```

Рис. 7. Приклад сертифіката

Скопіюйте цей вихід на наш комп'ютер, включаючи рядки `-----BEGIN CERTIFICATE-----` і `-----END CERTIFICATE-----`, і збережіть його у файл з впізнаваним ім'ям, наприклад `ca-cert.pem`. Переконайтеся, що файл, який ми створюємо, має розширення `.pem`.

Підключення з Windows

Існує безліч способів імпорту кореневого сертифіката та налаштування Windows для підключення до VPN. У першому способі використовуються графічні інструменти для кожного кроку. Другий метод використовує команди PowerShell, які можуть бути написані сценаріями та змінені відповідно до вашої конфігурації VPN.

Налаштування Windows за допомогою графічних інструментів.

Спочатку імпортуйте кореневий сертифікат, виконавши такі дії:

1) Натисніть сполучення клавіш `WINDOWS+R`, щоб відкрити діалогове вікно «Виконати», і введіть `mms.exe` щоб запустити консоль керування Windows.

2) У меню «Файл» перейдіть до розділу «Додати або видалити оснастку», виберіть «Сертифікати» зі списку доступних оснасток і натисніть «Додати».

3) Ми хочемо, щоб VPN працював з будь-яким користувачем, тому виберіть «Обліковий запис комп'ютера» та натисніть «Далі».

4) Ми налаштовуємо речі на локальному комп'ютері, тому виберіть «Локальний комп'ютер», а потім натисніть «Готово».

5) Під кореневим вузлом консолі розгорніть запис Сертифікати (локальний комп'ютер), розгорніть Довірені кореневі центри сертифікації, а потім виберіть запис Сертифікати.

6) У меню Дія виберіть пункт Усі завдання та натисніть кнопку Імпортувати, щоб відобразити майстер імпорту сертифікатів. Натисніть Далі, щоб перейти до вступу.

7) На екрані «Файл для імпорту» натисніть кнопку «Огляд», переконайтеся, що ми змінюємо тип файлу з «Сертифікат X.509 (.cer; .crt)» на "Усі файли (.)" і виберіть файл sa-cert.pem, який ми зберегли. Потім натисніть кнопку Далі.

8) Переконайтеся, що для сховища сертифікатів встановлено значення Довірені кореневі центри сертифікації, і натисніть кнопку Далі.

Натисніть кнопку Готово, щоб імпортувати сертифікат.

Потім налаштуйте VPN, виконавши такі дії:

1) Запустіть «Панель керування», а потім перейдіть до Центру управління мережами та загальним доступом.

2) Натисніть Налаштувати нове підключення або мережу, а потім виберіть Підключитися до робочого місця.

3) Виберіть Використовувати підключення до Інтернету (VPN).

4) Введіть дані VPN-сервера. Введіть доменне ім'я або IP-адресу сервера в поле «Адреса Інтернету», а потім заповніть «Ім'я призначення» вказаним для вашого VPN-з'єднання. Потім натисніть «Готово».

У підсумку маємо наступну схему мережі (рис. 8).

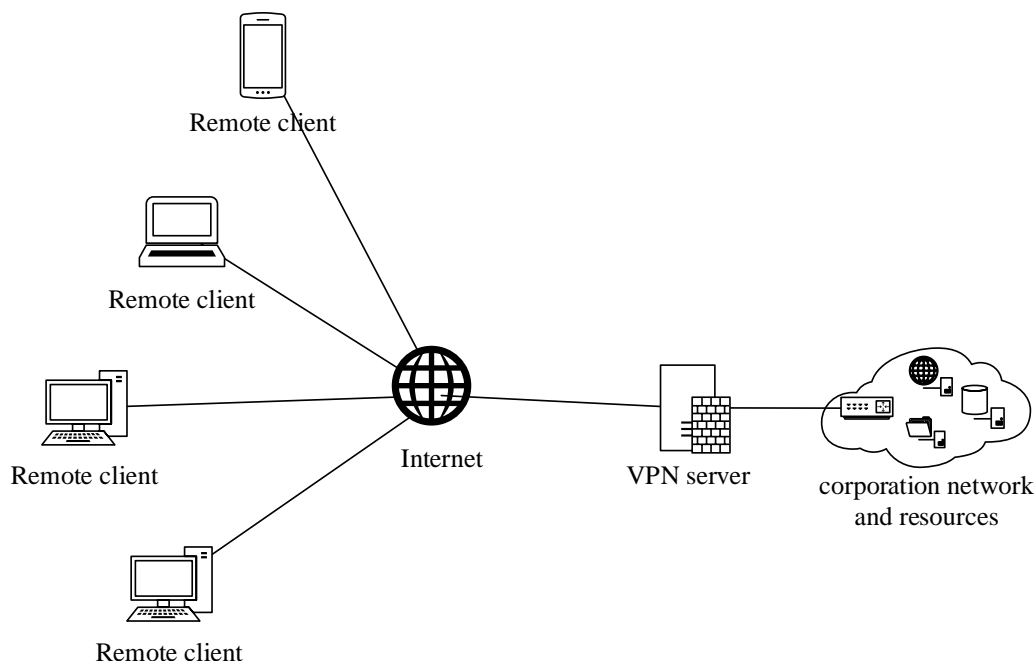


Рис. 8. Схема мережі

Висновки

Виконавши налаштування сервера на базі Ubuntu 20.04 з використанням рішення StrongSwan – демона IPsec, який підтримує як IKEv1, так і IKEv2, спочатку був створений 4096-бітний ключ RSA для підпису кореневого ЦС під час його генерації. Після цього було створено закритий ключ VPN-сервера, підписаний сертифікат VPN-сервера за допомогою

ключа центру сертифікації, а також здійснено налаштування IPsec, внівши зміни до конфігураційного файлу.

Після цих кроків було проведено налаштування аутентифікації, що означало настройку ідентифікатора (логіна) і пароля. Останнім етапом була настройка брандмауера, відкриття доступу для OpenSSH для віддаленого управління сервером, дозвіл стандартних портів IPsec та налаштування переадресації.

Цей тип зв'язку забезпечує надійну безпеку, оскільки сервер запитує у клієнта сертифікат і дані автентифікації. Ще однією перевагою є використання типів шифрування, які сумісні з усіма платформами. Проте, якщо конкретна платформа не підтримує найбільш безпечний набір, обирається відповідний, що може призвести до трохи менш безпечного шифрування.

Створення такого сервера дозволяє підключати віддалених користувачів, яких стає все більше в сучасному світі.

Перелік посилань

1. Могильний Г.А., Семенов М.А., Кіреєв І.Ю. Впровадження системи віддаленого доступу до інформаційних ресурсів комп'ютерних лабораторій. Вісник Східноукраїнського національного університету імені Володимира Даля, № 2 (272), 2022. – С. 7–14. <https://doi.org/10.33216/1998-7927-2022-272-2-7-14>
2. Бойко, Ю., & Білявць, Б. (2022). SAML: Дефініція та принцип роботи через VPN тунель у захищених інформаційних мережах. *Measuring and Computing Devices in Technological Processes*, (4), 41–48. <https://doi.org/10.31891/2219-9365-2022-72-4-4>
3. Тушук, І. (2023). Вибір технології віддаленого доступу для ефективної організації захисту мережних з'єднань. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 3(19), 34–45. <https://doi.org/10.28925/2663-4023.2023.19.3445>
4. Yeboah-Boateng, E. and Kwabena-Adade, G. (2020) Remote Access Communications Security: Analysis of User Authentication Roles in Organizations. *Journal of Information Security*, 11, 161-175. doi: 10.4236/jis.2020.113011.
5. Snader, J. *VPNs Illustrated: Tunnels, VPNs, and IPsec* // Addison-Wesley, 2006. Access mode: <https://www.amazon.com/VPNs-Illustrated-Tunnels-IPsec/dp/032124544X>
6. IKEv2 Cipher Suites. [Electronic resource]. Access mode: <https://docs.strongswan.org/docs/5.9/config/IKEv2CipherSuites.html>
7. ChaCha20, Poly1305, and Their Use in the Internet Key Exchange Protocol (IKE) and IPsec. [Electronic resource]. Access mode: <https://www.rfc-editor.org/rfc/rfc7634.html>
8. Internet Key Exchange Protocol Version 2 (IKEv2). [Electronic resource]. Access mode: <https://www.rfc-editor.org/rfc/rfc7296.html>
9. Stallings, W. *Effective Cybersecurity. A Guide to Using Best Practices and Standards*, 2018. Access mode: <https://www.amazon.com/Effective-Cybersecurity-Guide-Practices-Standards/dp/0134772806>
10. Бірюков, А. Інформаційна безпека: захист та напад, 3-є видання, 2023. Режим доступу: <https://liderbooks.com.ua/ua/p2008314790-informatsionnaya-bezopasnost-zaschita.html>
11. Andrew, S. Tanenbaum, David J. *Wetherall Computer Networks*, 5th Edition, 2013. Access mode: https://faculty.ksu.edu.sa/sites/default/files/computer_networks_-_a_tanenbaum_-_5th_edition.pdf

Надійшла: 29.10.2023

Рецензент: д.т.н., професор Ахромович В.М.