СЪЗДАВАНЕ НА ВИРТУАЛНА ЧАСТНА МРЕЖА ЧРЕЗ WIREGUARD

Делян Генков
Технически университет - Габрово

CREATING A VIRTUAL PRIVATE NETWORK USING WIREGUARD

Delyan Genkov
Technical University - Gabrovo

Abstract

The virtual private networks (VPN) are an essential part of our modern world. During the COVID crisis we all had to work from home and in the same time to access our corporate resources from anywhere. VPN's are a convenient way to establish a secure remote access to any corporate network. There are many protocols and software suites that can be used for setup a VPN access. Wireguard is a relatively new way for establishing a VPN network. The present paper aims to share the author's experience with this VPN suite.

Keywords: virtual private network, VPN, Wireguard.

ВЪВЕДЕНИЕ

Виртуалните частни мрежи (Virtual Private Network, VPN) са широко използван начин за достъп до отдалечени мрежови ресурси по сигурен начин чрез криптиране на предаваните данни. Има много протоколни набори, чрез които може да се реализира такава мрежа, един от широко използваните стандарти се нарича IPSec, създаден е от Internet Engineering Task Force (IETF) през 1995 година и неговата текуща версия е описана в документа RFC-2401. [1]. Той може да се намери вграден в софтуера на доста маршрутизатори, с цел реализация на сигурна отдалечена връзка.

Други често използвани пакети са Point-to-Point Tunneling Protocol (PPTP) [2] и Layer Two Tunneling Protocol (L2TP) [3], които са създадени през 1999 година и са вградени в операционната система Microsoft Windows.

Проектът OpenVPN [4], създаден през 2001 година също е популярен и се намира вграден в софтуера на някои маршрутизатори, главно от домашен и среден клас, като се отличава с лесните настройки и начин на използване.

През 2020 г. в ядрото на операционната система Linux беше включен един нов проект за създаване и използване на виртуални частни мрежи – Wireguard [5]. Той бързо спечели привърженици със своите възможности и в момента е компилиран и работи в много платформи, включително Windows, macOS, BSD, iOS, Android. WireGuard е изключително проста, но бърза и модерна VPN, която използва най-съвременна криптография. Той има за цел да бъде по-бърз, по-опростен, по-икономичен и по-полезен от IPsec, като същевременно избягва огромното главоболие по сложните настройки. Той има за цел да бъде значително по-производителен от OpenVPN. WireGuard е проектиран като VPN с общо предназначение за работа от вградени системи до супер компютри и е подходящ за много различни ситуации.[5]

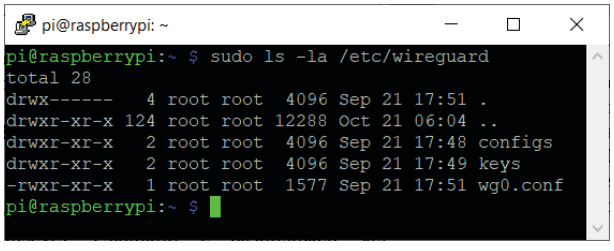
Сред предимствата на тази платформа са нейната висока производителност, минималната възможност за атаки, възможността за поддържане на връзка при преминаване от мрежа в мрежа (роуминг) и високото ниво на сигурност, осигурено от подбраните криптографски протоколи.

В разработката, която използва описаната мрежова инфраструктура беше необходимо да се избере протоколен набор за изграждане на виртуална частна мрежа между машини, разположени на различни места в страната и свързани чрез различни технологии за достъп до Интернет и през мрежи на различни Интернет доставчици. В част от тези свързаности не е възможно да се предостави публичен IP адрес на свързаността към Интернет. Тези машини трябва да установяват автоматично връзка към централен сървър, като получават фиксирани IP адреси във виртуалната частна мрежа, за да може да се идентифицира коректно машината. Компютърът на администратора също трябва да може да установява връзка към виртуалната частна мрежа, откъдето да достъпва машините, за да може да наблюдава тяхното състояние и да ги управлява. След направена поредица от тестове на различни платформи беше избран Wireguard за изграждане на виртуалната частна мрежа. Въпреки срещнатите трудности по време на изграждането, изборът се оказа успешен и мрежата изпълнява своите изисквания. В настоящия доклад са представени начините на конфигуриране и резултатите от използването на изградената мрежа, както и срещнатите предизвикателства и начините за тяхното решаване.

ИЗЛОЖЕНИЕ

За разработката са използвани Raspberry Pi компютри за сървър и клиенти с операционна система Raspbian версия 11 (bullseye). Сървърът е разположен зад маршрутизатор с публичен IP адрес и конфигурирано препращане на порт (port forwarding) за порта на сървъра – UDP 51820.

Инсталацията на сървърната част е в папка /etc/wireguard. Съдържанието и е показано на фигура 1.



***Фиг. 1**. Съдържание на сървърната папка*

Главният конфигурационен файл е wg0.conf. В папката configs се пазят генерираните конфигурационни файлове за всеки клиент, а в папката keys се пазят генерираните частен и публичен ключ на сървъра.

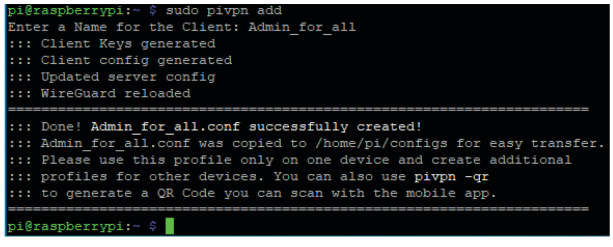
В конфигурационния файл основната секция е [Interface], показана на фигура 2.

```
[Interface]
PrivateKey = …
Address = 10.8.0.1/24
PostUp = iptables -A FORWARD -i wg0 -j ACCEPT; iptables -A FOR ACCEPT; iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
PostDown = iptables -D FORWARD -i wg0 -j ACCEPT; iptables - D -j ACCEPT; iptables -t nat -D POSTROUTING -o eth0 -j MASQUERADE
MTU = 1420
ListenPort = 51820
```

***Фиг. 2**. Обща конфигурация на сървъра*

В секцията PrivateKey се изписва генерирания частен ключ на сървъра, който е премахнат от показаната конфигурация от съображения за сигурност. В секцията Address се указва мрежата, от която ще се раздават адреси на клиентските устройства. Командите в секциите PostUp и PostDown се използват за добавяне и премахване на изключение за необходимите пакети в защитната стена на операционната система при установяване и прекъсване на връзката. Секцията MTU е необходима за намаляване на подразбиращия се максимален размер на пакета от 1500 байта на 1420 байта, за да могат увеличените със заглавната част на VPN протокола пакети да преминават през установения тунел без фрагментация. Накрая в секцията ListenPort се указва UDP порта, който се използва за връзката.

За генериране на конфигурация за нов клиент се използва командата sudo pivpn add, както е показано на фигура 3.



***Фиг. 3**. Генериране на нов клиент*

След изпълнението на тази команда трябва да въведем име на потребителя в случая Admin_for_all

Командата създава потребителски конфигурационен файл с име Admin_for_all.conf името му е такова заради, ролята му която ще изпълнява. Този потребител ще може да вижда всички останали потребители свързани към сървъра, като за целта са му нужни определени настройки за да изпълни тези изисквания. За да видим и редактираме този файл трябва да навигираме до директория /home/pi/configs. Това подсещане се изписва след като създадем определения потребител. Тази реализация на VPN сървър поддържа и опция за генериране на QR код като с негова помощ и мобилно приложение, което сканира QR кода ще има достъп до VPN сървъра.

За всеки конфигуриран клиент в конфигурационния файл на сървъра трябва да има по една секция [Peer], която да съдържа показаната на фигура 4 конфигурация.

```
[Peer]
PublicKey = …
PresharedKey = …
AllowedIPs = 10.8.0.7/32
```

Фиг. 4*. Секция за клиентска конфигурация*

В секцията PublicKey се записва генерирания публичен ключ на отдалеченото устройство, а в секцията PresharedKey – споделения ключ, който е еднакъв за двете устройства. Двата ключа са премахнати от показаната конфигурация от съображения за сигурност. В секцията AllowedIPs се записва IP адресът, който ще получи клиента с префикс /24.

За да се конфигурира клиентското устройство, трябва да се копира съдържанието на генерирания файл за съответния клиент и да се запише в папката /etc/wireguard като файл с име wg0.conf. необходимото съдържание е показано на фигура 5. Към автоматично генерираното съдържание трябва да се добавят и някои нови неща, за да може само необходимите пакети да преминават през VPN тунела и да се осъществява автоматично връзката.

```
[Interface]
PrivateKey = …
Address = 10.8.0.2/24
DNS = 8.8.8.8, 8.8.4.4

[Peer]
PublicKey = …
PresharedKey = …
Endpoint = 1.2.3.4:51820
AllowedIPs = 10.8.0.0/16
PersistentKeepalive=10
```

Фиг. 5*. Конфигурация на Raspberry PI клиент*

В секцията PrivateKey се записва генерираният частен ключ за тази станция, както е записан в автоматично генерирания файл. В секцията Address се записва IP адресът, с който ще работи тази станция. За нашата задача е важно станциите да имат статични IP адреси, за да се знае коя точно машина се достъпва в даден момент от време. В секцията DNS се записват адреси на един или повече DNS сървъри, които ще използва клиента. В секцията PublicKey се записва генерираният публичен ключ на сървъра, а в секцията PresharedKey – споделеният ключ за тази станция. В конфигурационния фрагмент ключовете са премахнати от съображения за сигурност. В секцията Endpoint се записва публичния IP адрес на сървъра, заедно с номера на порта, на който е конфигуриран Wireguard сървъра (показан на фигура 2 в секцията ListenPort).

Секцията AllowedIPs по подразбиране се генерира със стойност 0.0.0.0/0, ::0 при автоматичния процес на генериране, но тази стойност указва целия трафик на отдалечената машина за протоколите IPv4 и IPv6 да се препраща към тунела, което няма да позволи локалната машина да достъпва ресурси в Интернет. Затова е добре тя да се смени, както е показано на фигура 6 – да позволява само трафикът за мрежата на адресите на VPN мрежата (в случая 10.8.0.0) да бъде препращана през тунела, а останалия трафик да бъде маршрутизиран през локалната Интернет свързаност. В нашата мрежа не се използва протокол IPv6, затова неговата конфигурация се премахва от автоматично генерирания файл. Също така е необходимо ръчно да се добави в конфигурационния файл секцията PersistentKeepalive=10, което ще позволи връзката на отдалечения

клиент да се установи автоматично при наличие на Интернет свързаност към сървъра.

От конзолата на сървъра може да се види информация за свързаните клиенти с командата: sudo wg show, както е показано на фигура 6.

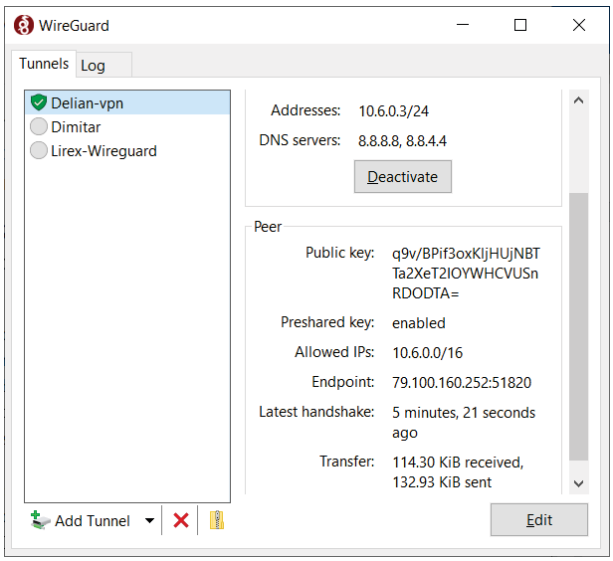
```
root@raspberrypi:/home/pi# wg show
interface: wg0
  public key: …
  private key: (hidden)
  listening port: 51820

peer:
cO8uzrHuvkUFqANlTE4DYcc3Jc5rx7ZkfYUXG
+qS2hE=
  preshared key: (hidden)
  endpoint: 5.6.7.8:41230
  allowed ips: 10.6.0.2/32
  latest handshake: 1 second ago
  transfer: 20.94 MiB received,
11.28 MiB sent
```

***Фиг. 6**. Проверка на свързаните клиенти*

Показаната информация съдържа данни за публичния ключ на сървъра, който не е таен, но е премахнат от конфигурационния файл за по-добра прегледност, частния ключ, който е таен не се визуализира от командата. Показва се още и номерът на порта, на който е конфигуриран сървъра. За всеки свързан клиент отдолу се визуализира следната информация – публичния ключ на клиента, неговия публичен IP адрес и номерът на порта, частният му IP адрес във VPN мрежата и информация за предадените и приети данни към и от този клиент. От съображения за сигурност командата не показва споделения ключ. В примера е показана информация само за един клиент.

На компютъра на администраторите на системата се инсталира Wireguard VPN Client – в случая се използва операционна система Microsoft Windows и е показана работата и конфигурацията на този клиент, но е аналогично за останалите поддържани операционни системи. За този клиент също се генерира конфигурационен файл на сървъра, както е показано на фигура 3 и конфигурационния файл се импортира в клиента, като се прави отново редакция на IP адресите, които ще се препращат през тунела, в противен случай клиентът няма да има Интернет свързаност. Екранът на клиентския софтуер е показан на фигура 7.



***Фиг. 7**. Клиент за Windows*

В този клиент са добавени три различни VPN тунела, които се използват за различни цели. В момента е активен тунелът, който се използва в конкретната разработка. Вижда се, че той е активен, както и може да се разбере информация за VPN адреса на компютъра, DNS сървърите, които се използват, публичния ключ на компютъра, позволените IP адреси, които преминават през тунела, както и информация за предадените и приети данни. С бутона Edit може да се редактира конфигурацията на тунела, а с Activate/ Deactivate да се активира и деактивира VPN свързаността. След активиране на тунела всички свързани машини вече са достъпни и могат да се наблюдават и управляват отдалечено през всеки разрешен протокол – в нашия случай ssh и http.

ЗАКЛЮЧЕНИЕ

Конфигурирана и тествана е виртуална частна мрежа, която свързва отдалечени машини с различен Интернет достъп към централен сървър, откъдето могат да бъдат наблюдавани и управлявани устройствата, свързани към мрежата. Избран е подходящ протоколен набор за изграждането и са показани необходимите настройки за конфигурирането на сървъра и клиентските машини. В момента в мрежата са свързани три машини и три администраторски компютъра. В бъдеще се очаква да бъдат включени около 300 машини в мрежата. Получените резултати показват, че мрежата изпълнява своите функции и е адекватна на

поставените предварителни изисквания. Wireguard се оказва лесна за употреба технология с малко натоварване върху компютърните системи, но в същото време бърза, надеждна и сигурна.

БЛАГОДАРНОСТИ

Настоящият документ е изготвен с финансовата помощ на договор № 2203Е за провеждане на научни изследвания по проект на тема: „Разработка и валидиране на решения за ефективно дистанционно обучение чрез използване на иновативни ИКТ технологии“ към Технически университет – Габрово.

REFERENCE

[1] S. Kent, R. Atkinson, Security Architecture for the Internet Protocol, RFC 2401, https://datatracker.ietf.org/doc/html/rfc2401, date of usage 20.10.2022.

[2] K. Hamzeh, G. Pall, etc., Point-to-Point Tunneling Protocol (PPTP), RFC 2637, https://www.rfc-editor.org/rfc/rfc2637, date of usage 20.10.2022.

[3] W. Townsley, A. Valencia, etc., Layer Two Tunneling Protocol "L2TP", RFC 2661, https://datatracker.ietf.org/doc/html/rfc2661, date of usage 20.10.2022.

[4] OpenVPN, Secure access and network connectivity reimagined, https://openvpn.net/, Date of usage 20.20.2022

[5] Jason A. Donenfeld, Wireguard, fast, modern, secure VPN Tunnel, https://www.wireguard.com/, date of usage 20.10.2022.