**Задание:**

1. Между DC-RTR-2, DC-RTR-1, MSK-RTR, YEKT-RTR настройте удалённый защищённый туннель:
   * В качестве инструмента защиты используйте любое решение;
   * Важно организовать следующую логику работы соединений:
     + В случае, если соединение пропадает с Московского офиса до ЦОД - канал связи автоматически перестраивается на соединение с Екатеринбургом и перенаправляет трафик через резервного провайдера;
     + В случае, если соединение пропадает из офиса в Екатеринбурге - канал связи аналогично переключается на Московский офис и проходит по другому каналу связи;
     + Необходимо обеспечить бесперебойную работу сервера печати для всех филиалов, используйте этот сервис в качестве проверки доступности;
     + Основное соединение, ровно, как и резервные обязательно должны быть зашифрованы.

**IP addresation:**

**DC-RTR-1:**

ens20:

ens21:

ens22:

enp2s29: 200.100.100.20/24 -> AZAZON

gre1: 10.0.1.1/30 -> MSK-RTR

gre2: 10.0.2.1/30 -> YEKT-RTR

**DC-RTR-2:**

ens20:

ens21:

ens22:

enp2s29: 100.200.100.20/24 -> MOOGLE

gre1: 10.0.3.1/30 -> MSK-RTR

gre2: 10.0.4.1/30 -> YEKT-RTR

**MSK-RTR:**

ens20: 181.121.90.2/30 -> MOOGLE

ens21:

ens22:

enp2s29:

gre1: 10.0.1.2/30 -> DC-RTR-1

gre2: 10.0.3.2/30 -> DC-RTR-2

**YEKT-RTR:**

ens20: 88.8.8.27/27 -> AZAZON

ens21:

ens22:

enp2s29:

gre1: 10.0.2.2/30 -> DC-RTR-1

gre2: 10.0.4.2/30 -> DC-RTR-2

**MOOGLE:**

ens20: 100.200.100.10/24 -> DC-RTR-2

ens21: 31.18.10.10/24 -> REMOTE-TERMINAL

ens22: 188.121.90.1/30 -> MSK-RTR

**AZAZON:**

ens20: 200.100.100.10/24 – DC-RTR-1

ens21: 88.8.8.1/27 -> YEKT-RTR

ens22: 179.13.219.123/24 -> CLOUD-VM1

**Настройка bond и приоритета к разным роутерам на машине DC-STORAGE**

Устанавливаем пакет для bond:

apt install ifenslave

Редактируем network:

# Интерфейс bond0, объединяющий eth0 и eth1

auto bond0

iface bond0 inet static

address 10.15.10.150 # IP-адрес для бонда

netmask 255.255.255.0 # Маска подсети

gateway 10.15.10.3 # Основной шлюз

bond-mode 802.3ad # Режим бондинга (LACP)

bond-miimon 100 # Мониторинг линков каждые 100 миллисекунд

bond-updelay 200 # Задержка включения интерфейса

bond-downdelay 200 # Задержка отключения интерфейса

slaves ens20 ens21 # Интерфейсы, включенные в бонд

# Добавление маршрутов

up ip route add default via 192.168.2.1 metric 200 # Резервный шлюз с более низким приоритетом

# Настройка интерфейсов eth0 и eth1 (они будут использоваться в бонде, не назначаем IP-адреса)

iface eth0 inet manual

iface eth1 inet manual

systemctl restart networking

**Решение:**

**Подготовка:**

На роутерах DC-RTR-1, DC-RTR-2, MSK-RTR, YEKT-RTR:

**apt update**

**apt install strongswan keepalived iproute2 -y**

strongSwan — это решение для создания VPN-соединений с использованием протокола IPsec.

keepalived — будет использоваться для реализации механизма failover между основным и резервным каналами.

**Настраиваем туннели**

**На роутере DC-RTR-1:**

# Туннель до MSK-RTR

**sudo ip tunnel add gre1 mode gre local <IP-DC-RTR-1> remote <IP-MSK-RTR> ttl 255**

**sudo ip addr add 10.0.1.1/30 dev gre1**

**sudo ip link set gre1 up**

# Туннель до YEKT-RTR

**sudo ip tunnel add gre2 mode gre local <IP-DC-RTR-1> remote <IP-YEKT-RTR> ttl 255**

**sudo ip addr add 10.0.2.1/30 dev gre2**

**sudo ip link set gre2 up**

**На роутере DC-RTR-2:**

# Туннель до MSK-RTR

**sudo ip tunnel add gre1 mode gre local <IP-DC-RTR-2> remote <IP-MSK-RTR> ttl 255**

**sudo ip addr add 10.0.3.1/30 dev gre1**

**sudo ip link set gre1 up**

# Туннель до YEKT-RTR

**sudo ip tunnel add gre2 mode gre local <IP-DC-RTR-2> remote <IP-YEKT-RTR> ttl 255**

**sudo ip addr add 10.0.4.1/30 dev gre2**

**sudo ip link set gre2 up**

**На роутере MSK-RTR:**

# Туннель до DC-RTR-1

**sudo ip tunnel add gre1 mode gre local <IP-MSK-RTR> remote <IP-DC-RTR-1> ttl 255**

**sudo ip addr add 10.0.1.2/30 dev gre1**

**sudo ip link set gre1 up**

# Туннель до DC-RTR-2

**sudo ip tunnel add gre2 mode gre local <IP-MSK-RTR> remote <IP-DC-RTR-2> ttl 255**

**sudo ip addr add 10.0.3.2/30 dev gre2**

**sudo ip link set gre2 up**

**На роутере YEKT-RTR:**

# Туннель до DC-RTR-1

**sudo ip tunnel add gre1 mode gre local <IP-YEKT-RTR> remote <IP-DC-RTR-1> ttl 255**

**sudo ip addr add 10.0.2.2/30 dev gre1**

**sudo ip link set gre1 up**

# Туннель до DC-RTR-2

**sudo ip tunnel add gre2 mode gre local <IP-YEKT-RTR> remote <IP-DC-RTR-2> ttl 255**

**sudo ip addr add 10.0.4.2/30 dev gre2**

**sudo ip link set gre2 up**

**Чтобы добавить скрипт в автозапуск:**

**@reboot root /bin/bash /path/to/script.sh**

**Настройка IPsec VPN:**

**На роутере DC-RTR-1:**

**/etc/ipsec.conf**

config setup

uniqueids=no

conn %default

keyexchange=ikev2

ike=aes256-sha2\_256-modp2048

esp=aes256-sha2\_256

dpdaction=restart

dpddelay=30s

dpdtimeout=120s

keyingtries=%forever

authby=secret

# Туннель с MSK-RTR

conn dc1-msk

left=200.100.100.20

right=181.121.90.2

auto=start

type=transport

# Туннель с YEKT-RTR

conn dc1-yekt

left=200.100.100.20

right=88.8.8.27

auto=start

type=transport

**/etc/ipsec.secrets**

200.100.100.20 181.121.90.2 : PSK "C00lCompanY"

200.100.100.20 88.8.8.27 : PSK "C00lCompanY"

systemctl restart ipsec

systemctl enable --now ipsec

**На роутере DC-RTR-2:**

config setup

uniqueids=no

conn %default

keyexchange=ikev2

ike=aes256-sha2\_256-modp2048

esp=aes256-sha2\_256

dpdaction=restart

dpddelay=30s

dpdtimeout=120s

keyingtries=%forever

authby=secret

# Туннель с MSK-RTR

conn dc2-msk

left=100.200.100.20

right=181.121.90.2

auto=start

type=transport

# Туннель с YEKT-RTR

conn dc2-yekt

left=100.200.100.20

right=88.8.8.27

auto=start

type=transport

100.200.100.20 181.121.90.2 : PSK "C00lCompanY"

100.200.100.20 88.8.8.27 : PSK "C00lCompanY"

**На роутере MSK-RTR:**

config setup

uniqueids=no

conn %default

keyexchange=ikev2

ike=aes256-sha2\_256-modp2048

esp=aes256-sha2\_256

dpdaction=restart

dpddelay=30s

dpdtimeout=120s

keyingtries=%forever

authby=secret

# Туннель с DC-RTR-1

conn msk-dc1

left=181.121.90.2

right=200.100.100.20

auto=start

type=transport

# Туннель с DC-RTR-2

conn msk-dc2

left=181.121.90.2

right=100.200.100.20

auto=start

type=transport

181.121.90.2 200.100.100.20 : PSK "C00lCompanY"

181.121.90.2 100.200.100.20 : PSK "C00lCompanY"

**На роутере YEKT-RTR:**

config setup

uniqueids=no

conn %default

keyexchange=ikev2

ike=aes256-sha2\_256-modp2048

esp=aes256-sha2\_256

dpdaction=restart

dpddelay=30s

dpdtimeout=120s

keyingtries=%forever

authby=secret

# Туннель с DC-RTR-1

conn yekt-dc1

left=88.8.8.27

right=200.100.100.20

auto=start

type=transport

# Туннель с DC-RTR-2

conn yekt-dc2

left=88.8.8.27

right=100.200.100.20

auto=start

type=transport

88.8.8.27 200.100.100.20 : PSK "C00lCompanY"

88.8.8.27 100.200.100.20 : PSK "C00lCompanY"

**Настройка EIGRP с аутентификацией MD5**

vim /etc/frr/daemons:

eigrpd=yes

systemctl restart frr

systemctl enable --now frr

Также на всех роутерах включаем forwarding

vim /etc/sysctl.conf

sysctl -p

**DC-RTR-1:**

vtysh

conf t

router eigrp 100

network 10.0.1.0/30

network 10.0.2.0/30

eigrp router-id 1.1.1.1

eigrp log-neighbor-changes

! Аутентификация MD5

key chain C00lCompanY

key 1

key-string C00lCompanY

interface gre1

ip authentication mode eigrp 100 md5

ip authentication key-chain eigrp 100 C00lCompanY

interface gre2

ip authentication mode eigrp 100 md5

ip authentication key-chain eigrp 100 C00lCompanY

exit

write

**DC-RTR-2:**

vtysh

conf t

router eigrp 100

network 10.0.3.0/30

network 10.0.4.0/30

eigrp router-id 2.2.2.2

eigrp log-neighbor-changes

key chain C00lCompanY

key 1

key-string C00lCompanY

interface gre1

ip authentication mode eigrp 100 md5

ip authentication key-chain eigrp 100 C00lCompanY

interface gre2

ip authentication mode eigrp 100 md5

ip authentication key-chain eigrp 100 C00lCompanY

exit

write

**MSK-RTR:**

vtysh

conf t

router eigrp 100

network 10.0.1.0/30

network 10.0.3.0/30

eigrp router-id 3.3.3.3

eigrp log-neighbor-changes

key chain C00lCompanY

key 1

key-string C00lCompanY

interface gre1

ip authentication mode eigrp 100 md5

ip authentication key-chain eigrp 100 C00lCompanY

interface gre2

ip authentication mode eigrp 100 md5

ip authentication key-chain eigrp 100 C00lCompanY

exit

write

**YEKT-RTR:**

vtysh

conf t

router eigrp 100

network 10.0.2.0/30

network 10.0.4.0/30

eigrp router-id 4.4.4.4

eigrp log-neighbor-changes

key chain C00lCompanY

key 1

key-string C00lCompanY

interface gre1

ip authentication mode eigrp 100 md5

ip authentication key-chain eigrp 100 C00lCompanY

interface gre2

ip authentication mode eigrp 100 md5

ip authentication key-chain eigrp 100 C00lCompanY

exit

write

show ip eigrp neighbors

**Конфигурация Keepalived на каждом маршрутизаторе:**

**DC-RTR-1**:

vim /etc/keepalived/keepalived.conf

vrrp\_script chk\_gre1 {

script "ping -c 2 10.0.1.2" # Проверка туннеля до MSK-RTR

interval 2

weight -10

}

vrrp\_script chk\_gre2 {

script "ping -c 2 10.0.2.2" # Проверка туннеля до YEKT-RTR

interval 2

weight -10

}

vrrp\_instance GRE1 {

state MASTER

interface gre1

virtual\_router\_id 51

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass C00lCompanY

}

track\_script {

chk\_gre1

}

}

vrrp\_instance GRE2 {

state MASTER

interface gre2

virtual\_router\_id 52

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass C00lCompanY

}

track\_script {

chk\_gre2

}

}

systemctl restart keepalived

systemctl enable --now keepalived

**DC-RTR-2**:

vrrp\_script chk\_gre1 {

script "ping -c 2 10.0.3.2" # Проверка туннеля до MSK-RTR

interval 2

weight -10

}

vrrp\_script chk\_gre2 {

script "ping -c 2 10.0.4.2" # Проверка туннеля до YEKT-RTR

interval 2

weight -10

}

vrrp\_instance GRE1 {

state MASTER

interface gre1

virtual\_router\_id 53

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass C00lCompanY

}

track\_script {

chk\_gre1

}

}

vrrp\_instance GRE2 {

state MASTER

interface gre2

virtual\_router\_id 54

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass C00lCompanY

}

track\_script {

chk\_gre2

}

}

**MSK-RTR:**

vrrp\_script chk\_gre1 {

script "ping -c 2 10.0.1.1" # Проверка туннеля до DC-RTR-1

interval 2

weight -10

}

vrrp\_script chk\_gre2 {

script "ping -c 2 10.0.3.1" # Проверка туннеля до DC-RTR-2

interval 2

weight -10

}

vrrp\_instance GRE1 {

state MASTER

interface gre1

virtual\_router\_id 55

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass C00lCompanY

}

track\_script {

chk\_gre1

}

}

vrrp\_instance GRE2 {

state MASTER

interface gre2

virtual\_router\_id 56

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass C00lCompanY

}

track\_script {

chk\_gre2

}

}

**YEKT\_RTR:**

vrrp\_script chk\_gre1 {

script "ping -c 2 10.0.2.1" # Проверка туннеля до DC-RTR-1

interval 2

weight -10

}

vrrp\_script chk\_gre2 {

script "ping -c 2 10.0.4.1" # Проверка туннеля до DC-RTR-2

interval 2

weight -10

}

vrrp\_instance GRE1 {

state MASTER

interface gre1

virtual\_router\_id 57

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass C00lCompanY

}

track\_script {

chk\_gre1

}

}

vrrp\_instance GRE2 {

state MASTER

interface gre2

virtual\_router\_id 58

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass C00lCompanY

}

track\_script {

chk\_gre2

}

}

На базе сервера CLOUD-VM1 необходимо реализовать OpenConnect со следующей логикой работы:

* + Машина CLOUD-VM1 является публичным ресурсом и компания не скрывает её адрес, а вот адреса роутеров в ЦОДе компания не планирует опубликовывать, именно поэтому CLOUD-VM1 является брокером подключений;
  + Машина CLOUD-VM1 должна проверять какой из двух роутеров доступен для подключения и подключаться только к тому, кто работает на текущий момент времени. Если роутер недоступен больше 1 минуты - необходимо автоматически подключиться на резервный канал связи;
  + Основным роутером является DC-RTR-1, резервным DC-RTR-2;
  + При восстановлении соединения, трафик снова должен проходить через основной канал связи;
  + Для пользователей VPN все переключения происходят незаметно