

Správa rozsiahlych infraštruktúr

Obsahom tohto cvičenia je ukážka správy rozsiahlej infraštruktúry, ktorá bude pozostávať z nasledujúcich častí:

1. Sieťové, multi vendor prostredie obsahujúce smerovače:
 - a. Cisco CSR1kv
 - b. Juniper
 - c. Mikrotik
2. Firewall OPNsense
3. Server

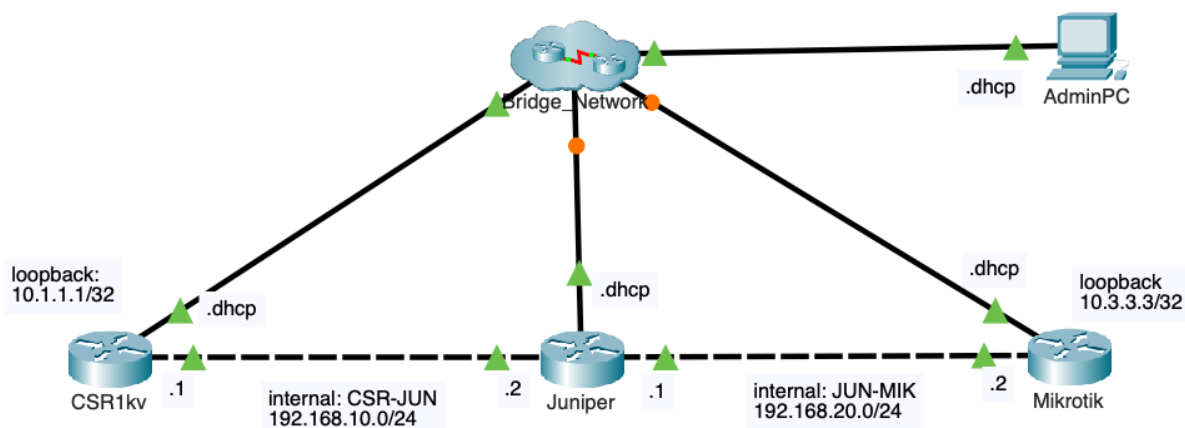
Konfigurácia multi vendor prostredia smerovačov

V počítačových sieťach sa je možné často stretnúť s prostredím, ktoré obsahuje smerovače od rôznych výrobcov. V rámci tejto kapitoly bude ukázané, ako je možné vykonať základnú konfiguráciu smerovača Cisco, Juniper a Mikrotik. Konfigurácia bude zahŕňať pomenovanie zariadení, nastavenie IP adries na rozhrania a konfigurácia protokolu OSPF na zabezpečenie smerovania. Ukážka konfigurácie bude obsahovať prácu s CLI, webovým rozhraním, a modernými prístupmi RESTCONF/NETCONF/YANG.

Topológia

Topológia sieťových zariadení bude vytvorená vo virtualizačnej platforme VirtualBox, v ktorom budú nasadené virtuálne obrazy sieťových zariadení:

- Cisco CSR1kv
- JunOS Olive
- MikroTik RouterOS CHR [<https://mikrotik.com/download> - CHR - OVA - 7.19.2]

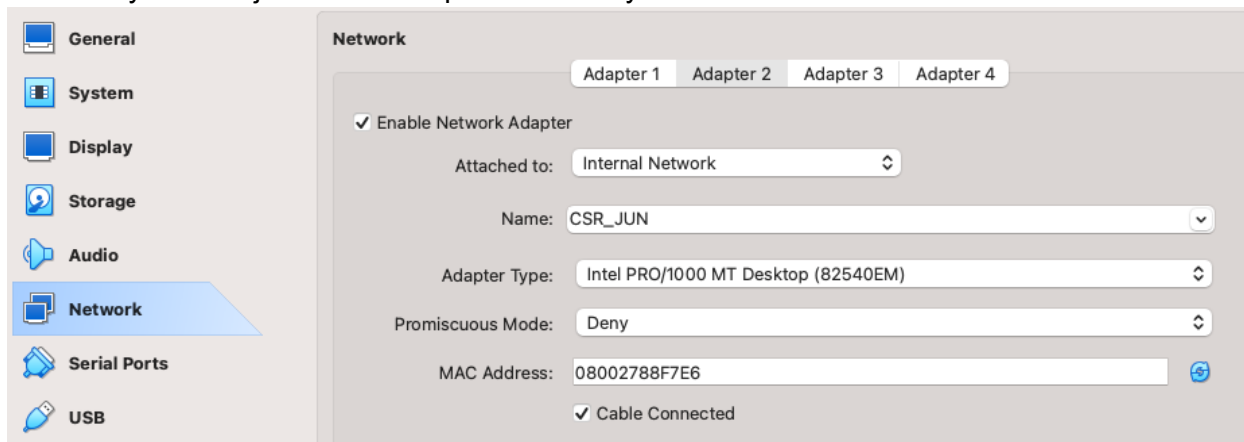


Networking v rozhraní VBox

Počítačová sieť v rozhraní VBox je možné vytvoriť pridaním viacerých rozhraní daným VM, pričom ako typy rozhraní budú použité:

1. Bridged Adapter (cez toto rozhranie bude možné priamo pristupovať z AdminPC na smerovače)
2. Internal Network (bude slúžiť na internú komunikáciu medzi smerovačmi)
 - a. CSR_JUN (prepája Cisco zariadenie s Juniper zariadením)
 - b. JUN_MIK (prepája Juniper zariadenie s Mikrotik zariadením)

Ukážka vytvorenia jedného z Juniper rozhraní vyzerať nasledovne:



Konfigurácia sieťových zariadení cez CLI

Nasledujúce časti znázorňujú konfiguráciu sieťových zariadení využitím ich CLI. CLI zariadenia je možné otvoriť priamo v rozhraní VBox. Neskôr, po konfigurácii smerovania a služby SSH sa bude možné na zariadenia pripojiť aj vzdialene.

Konfigurácia Cisco zariadenia

1. Konfigurácia názvu zariadenia

```
hostname csr1kv
```

2. Konfigurácia rozhraní

```
interface gi1
 ip address dhcp
 no shutdown
interface gi2
 ip add 192.168.10.1 255.255.255.0
 no shutdown
interface loopback 1
 ip add 10.1.1.1 255.255.255.255
```

3. Konfigurácia protokolu SSH

```
ip domain-name cnl.sk
crypto key generate rsa
  1024
username cisco secret cisco123!
enable secret cisco
line vty 0 4
  login local
```

4. Konfigurácia protokolu OSPF

```
router ospf 1
  network 192.168.10.0 0.0.0.255 area 0
  network 10.1.1.1 0.0.0.0 area 0
```

5. Overenie konfigurácie

```
show ip interface brief
show ip route
```

```
from_python(config)#do sh ip int br
Interface                IP-Address      OK? Method Status        Protocol
GigabitEthernet1         192.168.2.10    YES DHCP    up            up
GigabitEthernet2         192.168.10.1    YES manual  up            up
Loopback1                 10.1.1.1        YES manual  up            up

Gateway of last resort is 192.168.2.1 to network 0.0.0.0

    10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
    0       10.3.3.3/32 [110/3] via 192.168.10.2, 00:02:36, GigabitEthernet2
    0       192.168.20.0/24 [110/2] via 192.168.10.2, 00:02:36, GigabitEthernet2
```

6. Celú bežiacu konfiguráciu je možné zobrazíť príkazom **show running-config**.

Konfigurácia Juniper zariadenia

Defaultné prihlasovacie údaje k Juniper zariadeniu sú [root: <bez hesla>]. Konfigurácia Juniper zariadenia funguje tak, že každú konfiguráciu je potrebné potvrdiť príkazom Commit. Zapísaná konfigurácia bez Commit-u je len v "stage" stave = vykoná sa až po jej uložení do bežiackej konfigurácie.

1. Nastavenie root autentifikácie (root heslo) - bez toho nie je možné ukladať konfiguráciu

```
cli
configure
set system root-authentication plain-text-password
<heslo>
<heslo>
```

2. Konfigurácia názvu zariadenia

```
set system host-name juniper_device
commit
```

3. Nastavenie sieťových nastavení na rozhrania

- Príkaz run show zobrazuje aktuálny stav, príkaz show vykonanú konfiguráciu

```
run show interfaces brief
set interfaces em0 unit 0 family inet address 192.168.2.200/24
set interfaces em1 unit 0 family inet address 192.168.10.2/24
set interfaces em2 unit 0 family inet address 192.168.20.1/24
commit
show interfaces
```



```
root@juniper_device# show interfaces
em0 {
  unit 0 {
    family inet {
      address 192.168.2.200/24;
    }
  }
}
em1 {
  unit 0 {
    family inet {
      address 192.168.10.2/24;
    }
  }
}
em2 {
  unit 0 {
    family inet {
      address 192.168.20.1/24;
    }
  }
}
```

4. Konfigurácia protokolu SSH

- SSH kľúč sa generuje automaticky po spustení služby SSH
- Povolené je štandardne SSH aj Telnet

```
set system services ssh protocol-version v2

set system login user juniper uid 2000 class super-user
set system login user juniper authentication plain-text-password
<password>
<password>

delete system services telnet
commit
```

```

[rastislavpetija@Rastislavs-iMac ~ % ssh juniper@192.168.2.200
The authenticity of host '192.168.2.200 (192.168.2.200)' can't be established.
ECDSA key fingerprint is SHA256:kCVfq144deqeqlENge7mIJ37V1HDccsaV1yhY0ohxw.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.2.200' (ECDSA) to the list of known hosts.
[juniper@192.168.2.200's password:
--- JUNOS 12.1R1.9 built 2012-03-24 12:52:33 UTC
juniper@juniper_device> configure
Entering configuration mode
Users currently editing the configuration:
  root terminal v0 (pid 1592) on since 2025-06-26 12:19:14 UTC, idle 00:01:30
  [edit]

[edit]
juniper@juniper_device# █

```

5. Konfigurácia protokolu OSPF

```

set protocols ospf area 0.0.0.0 interface em1
set protocols ospf area 0.0.0.0 interface em2
commit

```

6. Overenie úspešnosti nadviazania OSPF susedstva a naučených sietí

```

run show route protocol ospf

```

```

juniper@juniper_device# run show route protocol ospf

```

```

inet.0: 9 destinations, 9 routes (9 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

```

```

10.1.1.1/32      *[OSPF/10] 00:03:45, metric 2
                  > to 192.168.10.1 via em1.0
10.3.3.3/32      *[OSPF/10] 00:35:34, metric 2
                  > to 192.168.20.2 via em2.0
224.0.0.5/32     *[OSPF/10] 00:35:45, metric 1
                  MultiRecv

```

7. Celú bežiacu konfiguráciu je možné zobrazíť príkazom **show**.

Konfigurácia MikroTik zariadenia

Defaultné prihlasovacie údaje k MikroTik zariadeniu sú [admin: <bez hesla>, alebo pri prvom prihlásení je vyžadované nastavenie hesla]

```
MMM      MMM      KKK      TTTTTTTTTT      KKK
MMMM     MMMM     KKK      TTTTTTTTTT      KKK
MMM MMMM MMM III KKK KKK RRRRRR 000000 TTT III KKK KKK
MMM MM  MMM III KKKKK RRR RRR 000 000 TTT III KKKKK
MMM     MMM III KKK KKK RRRRRR 000 000 TTT III KKK KKK
MMM     MMM III KKK KKK RRR RRR 000000 TTT III KKK KKK

MikroTik RouterOS 7.19.2 (c) 1999-2025      https://www.mikrotik.com/

Press F1 for help

2025-06-26 08:16:34 system,error,critical router was rebooted without proper shutdown
[admin@MikroTik] >
```

1. Nastavenie názvu zariadenia

```
/system/identity/set name=mikrotik_device
```

```
[admin@mikrotik] > /system/identity/set name=mikrotik_device
[admin@mikrotik_device] >
```

2. Zobrazenie existujúcich rozhraní

```
/interface/print
```

```
[admin@mikrotik_device] > /interface/print
Flags: R - RUNNING
Columns: NAME, TYPE, ACTUAL-MTU, MAC-ADDRESS
#  NAME  TYPE  ACTUAL-MTU  MAC-ADDRESS
0  R ether1 ether  1500  08:00:27:06:03:56
1  R ether2 ether  1500  08:00:27:8C:F7:E2
2  R lo    loopback 65536  00:00:00:00:00:00
```

3. Konfigurácia sieťových nastavení a ich následné zobrazenie

```
/ip/dhcp-client/add interface=ether1 disabled=no comment="Bridged Interface - DHCP"

/ip/address/add address=192.168.20.2/24 interface=ether2
comment="JUN_MIK"
/interface/ethernet/set ether2 disabled=no

/interface/bridge/add name=Loopback
/ip/address/add address=10.3.3.3/32 interface=Loopback
comment="Router Loopback"

/ip/address/print
```

```
[admin@mikrotik_device] > /ip/address/print
Flags: D - DYNAMIC
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS NETWORK INTERFACE
0 D 192.168.2.9/24 192.168.2.0 ether1
::: JUN_MIK
1 192.168.20.2/24 192.168.20.0 ether2
::: Router Loopback
2 10.3.3.3/32 10.3.3.3 Loopback
```

3. Nastavenie protokolu SSH

- SSH kľúč sa generuje automaticky po spustení služby SSH
- Používateľ v skupine full má plné práva
- Štandardne je na zariadenie povolený prístup cez SSH, Telnet a WinBox. Telnet je možné zakázať
- Nastavenie domény nie je potrebné

```
/ip/service/set ssh disabled=no
```

```
/user/add name=mikrotik password="mikrotik123!" group=full
```

```
/ip/service/set telnet disabled=yes
```

```
|rastislavpetija@Rastislavs-iMac ~ % ssh mikrotik@192.168.2.9
The authenticity of host '192.168.2.9 (192.168.2.9)' can't be established.
RSA key fingerprint is SHA256:0s+s5Z6lg8abhLo/7pFIabt2VRgEp+BVsTkzUDQs35o.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.2.9' (RSA) to the list of known hosts.
|mikrotik@192.168.2.9's password:
```

```
MMM      MMM      KKK      TTTTTTTTTT      KKK
MMMM     MMMM     KKK      TTTTTTTTTT      KKK
MMM MMMM MMM III KKK KKK RRRRRR 000000 TTT III KKK KKK
MMM MM  MMM III KKKKK RRR RRR 000 000 TTT III KKKKK
MMM     MMM III KKK KKK RRRRRR 000 000 TTT III KKK KKK
MMM     MMM III KKK KKK RRR RRR 000000 TTT III KKK KKK
```

MikroTik RouterOS 7.19.2 (c) 1999-2025 <https://www.mikrotik.com/>

Press F1 for help

2025-06-26 08:16:34 system,error,critical router was rebooted without proper shutdown

```
[mikrotik@mikrotik_device] > 
```

4. Konfigurácia protokolu OSPF

```
/routing/ospf/instance/add name=OSPF version=2 router-id=10.3.3.3
/routing/ospf/area/add name=area0 area-id=0.0.0.0 instance=OSPF
/routing/ospf/interface-template/add networks=192.168.20.0/24
area=area0
/routing/ospf/interface-template/add networks=10.3.3.3/32 area=area0

/ip/route/print
/routing/ospf/neighbor/print
/routing/ospf/interface/print
```

```
[admin@mikrotik_device] > /ip/route/print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, o - OSPF, d - DHCP
Columns: DST-ADDRESS, GATEWAY, ROUTING-TABLE, DISTANCE
  DST-ADDRESS    GATEWAY          ROUTING-TABLE  DISTANCE
DAd 0.0.0.0/0     192.168.2.1      main           1
DAo 10.1.1.1/32   192.168.20.1%ether2 main           110
DAc 10.3.3.3/32   Loopback         main           0
DAc 192.168.20/24 ether1           main           0
DAo 192.168.10.0/24 192.168.20.1%ether2 main           110
DAc 192.168.20.0/24 ether2           main           0
```

```
[admin@mikrotik_device] > routing/ospf/neighbor/print
Flags: V - virtual; D - dynamic
 0 D instance=OSPF area=area0 address=192.168.20.1 priority=128
   router-id=192.168.2.200 dr=192.168.20.2 bdr=192.168.20.1 state="Full"
   state-changes=6 adjacency=32m41s timeout=38s
```

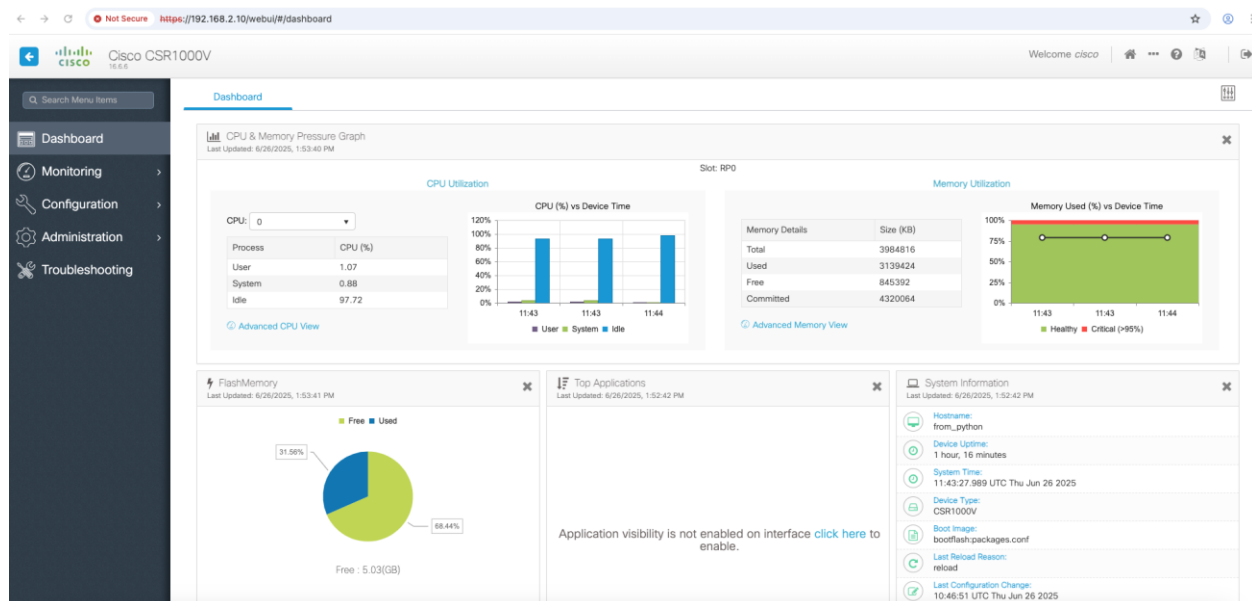
```
[admin@mikrotik_device] > routing/ospf/interface/print
Flags: D - dynamic
 0 D address=192.168.20.2%ether2 area=area0 state=dr network-type=broadcast
   bdr=192.168.20.1 cost=1 priority=128 use-bfd=no retransmit-interval=5s
   transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.3.3.3%Loopback area=area0 state=dr network-type=broadcast
   cost=1 priority=128 use-bfd=no retransmit-interval=5s transmit-delay=1s
   hello-interval=10s dead-interval=40s
```

Konfigurácia sieťových zariadení grafickým rozhraním

Cisco Web Interface

Smerovač CSR1kv poskytuje webové rozhranie pre pripojenie sa k nemu, ktoré je dostupné na: https://<ip_adresa_smerovaca>/webui. Prihlásiť sa je možné využitím používateľa vytvoreného na smerovači <username ... secret ...>.

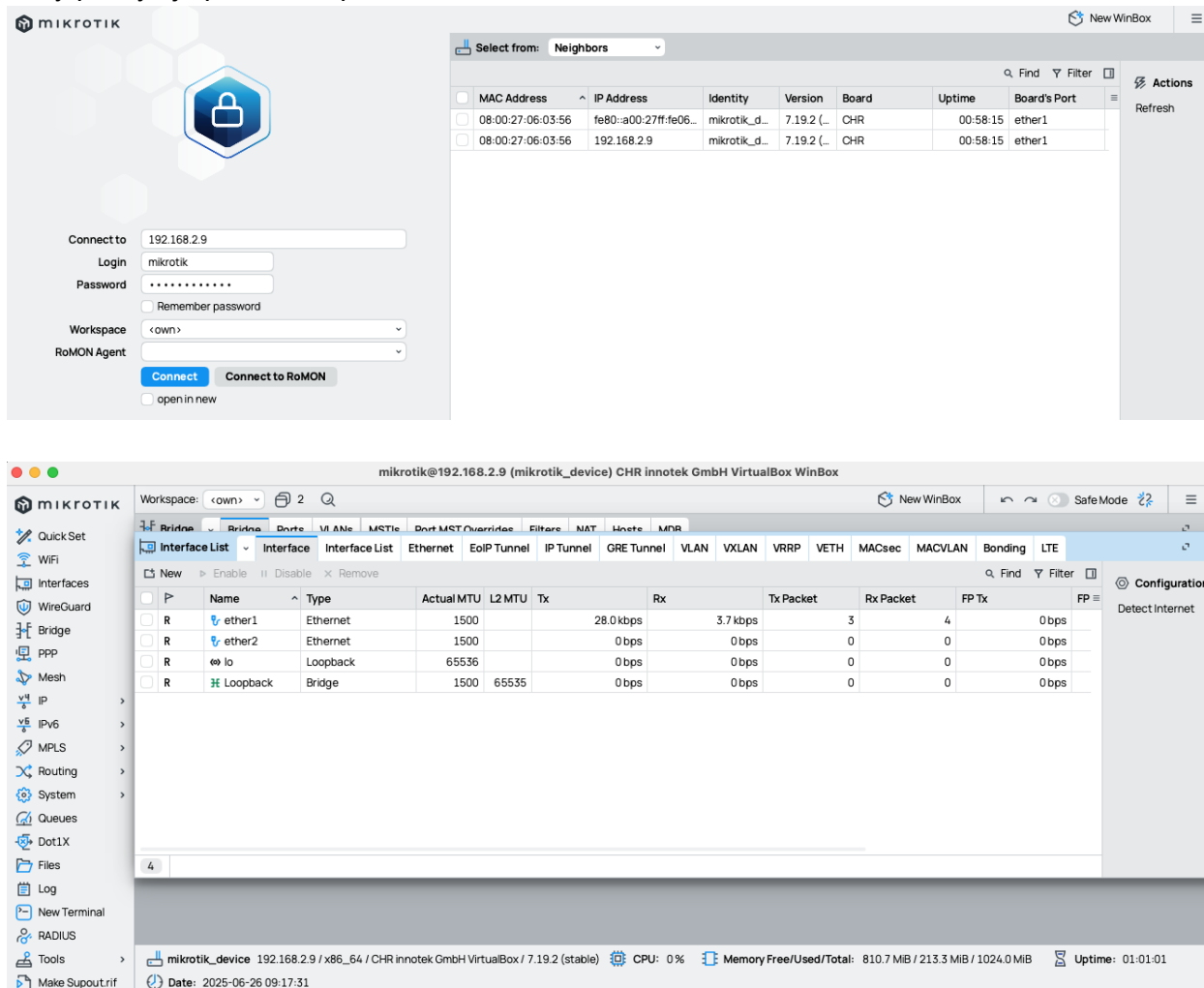


Juniper j-web

Juniper zariadenie poskytuje webové rozhranie nazývané j-web. Avšak vzhľadom k tomu, že zariadenie, ktoré máme k dispozícii nemá túto funkcionality, tak praktická ukážka práce s grafickým rozhraním j-web ukázaná nebude.

Mikrotik - WinBox

Mikrotik zariadenia sa odporúča konfigurovať cez nástroj WinBox [<https://mikrotik.com/download>], ktorý poskytuje prehľadné používateľské rozhranie:



Konfigurácia sieťových zariadení programovým prístupom

Cisco - NETCONF, RESTCONF

K zariadeniu Cisco sa je možné pripojiť protokolom NETCONF vykonaním nasledujúcich krokov:

1. Konfigurácia služby na cisco zariadení

```
netconf-yang
```

2. Pripojenie sa z terminálu príkazom: **[ssh cisco@192.168.2.10 -p 830]**
3. Odoslanie terminálom XML požiadavku reprezentujúcu požadovanú akciu. Nasledujúci výpis postupne vykoná otvorenie spojenia, načítanie informácií o rozhraniach, názve zariadenia a na zmenu názvu zariadenia.

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
  </capabilities>
</hello>
]]>]]>
```

```
<rpc message-id="103"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter>
      <interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-
interfaces"/>
    </filter>
  </get>
</rpc>
]]>]]>
```

```
<rpc message-id="103"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter>
      <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-
native">
        <hostname>
        </hostname>
      </native>
    </filter>
  </get>
</rpc>
]]>]]>
```

```
<rpc message-id="103"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
```

```
        <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-  
native">  
            <hostname>TEST_Hostname_NETCONF</hostname>  
        </native>  
    </config>  
</edit-config>  
</rpc>  
]]>]]>
```

4. Protokolom Netconf sa je možné na zariadenie pripojiť tiež využitím programovacieho jazyka.

a. Zdrojový kód v jazyku Python pre načítanie názvu zariadenia vyzerá nasledovne:

```
from ncclient import manager  
import xmltodict  
  
#definovanie udajov na pripojenie na NETCONF  
HOST = "192.168.2.10"  
USER = "cisco"  
PASS = "cisco123!"  
PORT = 830  
  
#filter pre prácu s názvom zariadenia  
netconf_filter = """  
<filter>  
    <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">  
        <hostname>  
        </hostname>  
    </native>  
</filter>  
"""  
  
#vytvorenie NETCONF spojenia a odoslanie GET_Config poziadavky  
m = manager.connect(host=HOST, port=PORT, username=USER,  
password=PASS, hostkey_verify=False, device_params={"name": "csr"})  
reply = m.get_config(source="running", filter = netconf_filter)  
  
print(reply)  
print("-----")  
  
#parsovanie XML  
result = xmltodict.parse(reply.xml)  
print(result)  
print("-----")  
print(result["rpc-reply"]["data"]["native"]["hostname"])  
  
#Uzatvorenie spojenia  
m.close_session()
```

b. Zdrojový kód v jazyku Python pre zmenu názvu zariadenia vyzerá nasledovne:

```

from ncclient import manager
import xmltodict

#definovanie udajov na pripojenie na NETCONF
HOST = "192.168.2.10"
USER = "cisco"
PASS = "cisco123!"
PORT = 830

#konfiguracny objekt
config_data = """
<config>
  <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
    <hostname>Router_python_Netconf</hostname>
  </native>
</config>
"""

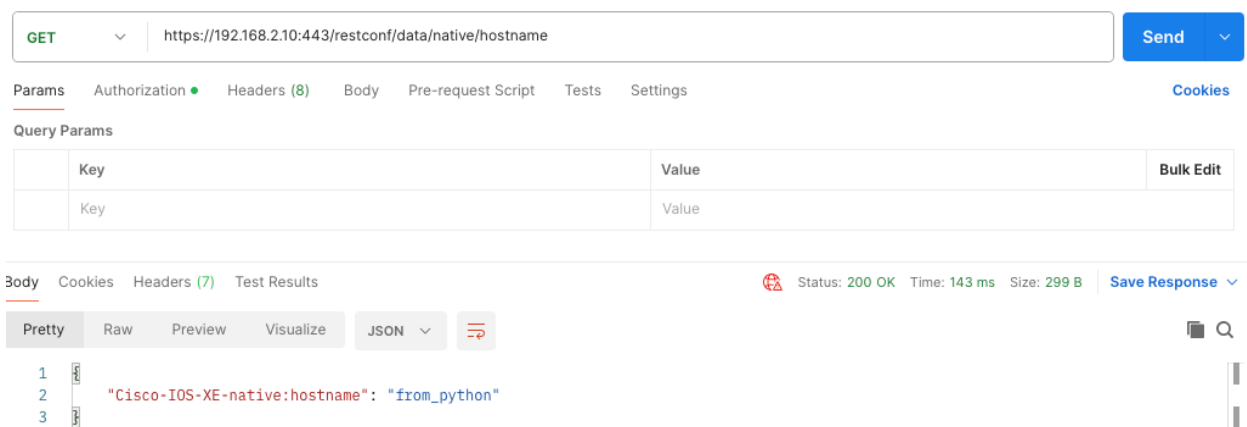
#vytvorenie NETCONF spojenia a odoslanie EDIT_Config požiadavky
m = manager.connect(host=HOST, port=PORT, username=USER,
password=PASS, hostkey_verify=False, device_params={"name": "csr"})
reply = m.edit_config(target="running", config = config_data)

#Uzatvorenie spojenia
m.close_session()

```

K zariadeniu Cisco sa je možné pripojiť protokolom RESTCONF vykonaním nasledujúcich krokov:

1. Stačí odoslať HTTP GET/PUT požiadavku s špecifikovaním požadovaných parametrov. Pre začiatok sa odporúča pracovať s nástrojom Postman, ktorého použitie znázorňuje nasledujúci obrázok:



2. Ukážka zdrojového kódu v jazyku Python:
 - a. Načítanie názvu zariadenia

```

#import knižníc pre prácu s JSON a HTTP protokolom a akceptovanie
self-signed certifikátov

```

```

import json
import requests
requests.packages.urllib3.disable_warnings()

if __name__ == '__main__':
    #sekcia, ktorá vytvorí http správu na komunikáciu s RESTCONF API
    url = "https://192.168.2.10:443/restconf/data/native"
    payload = {}
    headers = {
        'Accept': 'application/yang-data+json',
        'Content-Type': 'application/yang-data+json',
        'Authorization': 'Basic Y2lzY286Y2lzY28xMjMh'
    }

    #sekcia, ktorá pošle GET žiadosť a prevedie prijaté dáta do
    dict(JSON object) štruktúry
    get_result = requests.request("GET", url, headers=headers,
    data=payload, verify=False)
    json_data = get_result.json()

    #výpis prijatých dát + výpis konkrétnej hodnoty z JSON štruktúry
    print(json.dumps(json_data, indent = 2))
    print("\nNazov zariadenia: " + json_data["Cisco-IOS-XE-
    native:native"]["hostname"])

```

b. Zmena názvu zariadenia

```

#import knižnic pre prácu s JSON a HTTP protokolom a akceptovanie
self-signed certifikátov
import json
import requests
requests.packages.urllib3.disable_warnings()

if __name__ == '__main__':
    #sekcia, ktorá vytvorí http správu na komunikáciu s RESTCONF API
    url = "https://192.168.2.10:443/restconf/data/native/hostname"
    payload = {"Cisco-IOS-XE-native:hostname": "new_hostname123"}
    headers = {
        'Accept': 'application/yang-data+json',
        'Content-Type': 'application/yang-data+json',
        'Authorization': 'Basic Y2lzY286Y2lzY28xMjMh'
    }

    #sekcia, ktorá pošle PUT žiadosť na zmenu konfigurácie
    put_result = requests.request("PUT", url, headers=headers,
    data=json.dumps(payload), verify=False)

    #výpis správy o úspešnosti/neúspešnosti vykonania PUT operácie
    if(put_result.status_code == 204):
        print("Zmena konfigurácie úspešná :)")

```

```
else: print("Zmena konfigurácie neúspešná :'(")
```

Juniper - NETCONF

K zariadeniu Juniper sa je možné pripojiť protokolom NETCONF vykonaním nasledujúcich krokov:

1. Konfigurácia služby na juniper zariadení

```
set system services netconf ssh port 830
commit
```

2. Pripojenie sa z terminálu príkazom: **[ssh juniper@192.168.2.200 -t netconf]**
3. Odoslanie terminálom XML požiadavku reprezentujúcu požadovanú akciu. Nasledujúci príkaz slúži na načítanie informácií o rozhraniach.

```
<rpc>
  <get-config>
    <source>
      <running/>
    </source>

    <filter type="subtree">
      <configuration>
        <interfaces>
        </interfaces>
      </configuration>
    </filter>
  </get-config>
</rpc>
]]>]]>
```

```
]]>]]><rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns:junos="http://xml.juniper.net/junos/12.1R1/junos">
<data>
<configuration xmlns="http://xml.juniper.net/xnm/1.1/xnm" junos:commit-seconds="1750945602" junos:commit-localtime="2025-06-26 13:46:42 UTC" junos:
commit-user="root">
  <interfaces>
    <interface>
      <name>em0</name>
      <unit>
        <name>0</name>
        <family>
          <inet>
            <address>
              <name>192.168.2.200/24</name>
            </address>
          </inet>
        </family>
      </unit>
    </interface>
```

Dokumentácia k ďalším akciám je dostupná na:

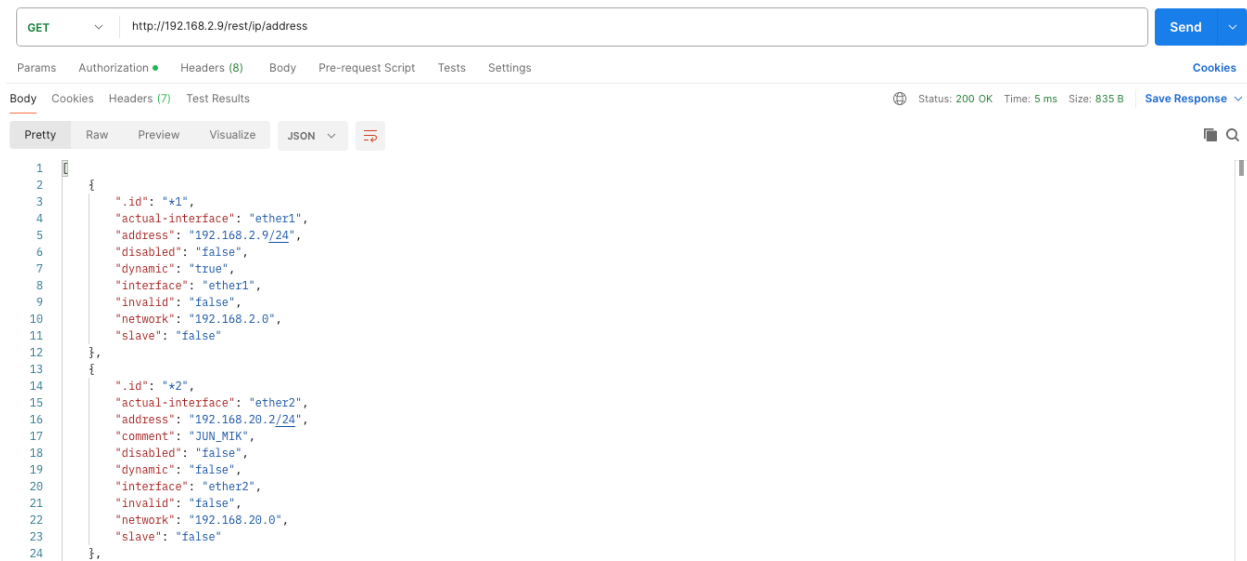
<https://www.juniper.net/documentation/us/en/software/junos/netconf/topics/topic-map/netconf-overview.html>

Mikrotik - RESTCONF

Dokumentáciu k RESTCONF prístupu na Mikrotik zariadenia je možné nájsť na:

<https://help.mikrotik.com/docs/spaces/ROS/pages/47579162/REST+API>

Ukážka HTTP požiadavky pre načítanie ip adries z Mikrotik zariadenia je znázornená na nasledujúcom obrázku a CURL výpise:



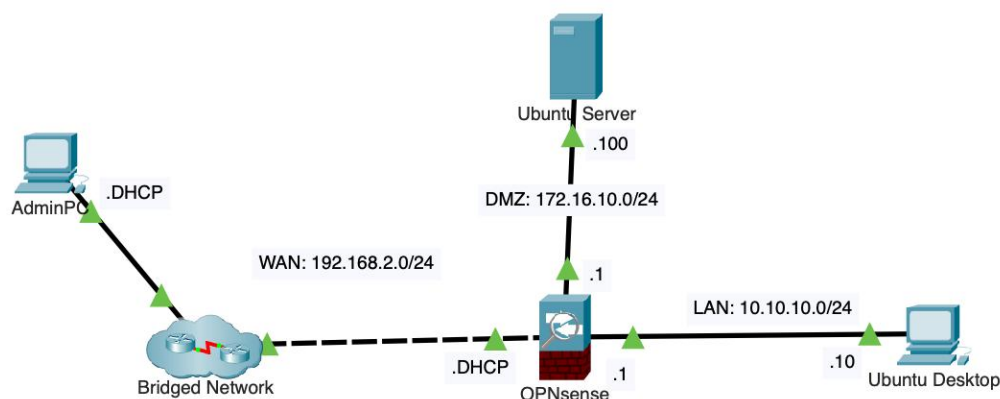
```
curl --location 'http://192.168.2.9/rest/ip/address' \
--header 'Authorization: Basic bWlrcm90aWs6bWlrcm90aWsxMjMh'
```

OPNsense firewall

V rámci tejto časti dokumentu je ukázaná práca z Firewall technológiou, konkrétne OPNsense.

Topológia

Ukážka práce s firewall-om bude demonštrovaná na nasledujúcej topológii.



Topológia pozostáva z nasledujúcich hlavných častí:

1. OPNsense Firewall:
 - a. Internal Network = LAN [Prvé rozhranie FW - em0]
 - b. Bridged Adapter = WAN [Druhé rozhranie FW - em1]
 - c. Internal Network = DMZ [Tretie rozhranie FW - bude neskôr vytvorené]
2. Ubuntu Server: Internal Network = DMZ
3. Ubuntu Desktop: Internal Network = LAN
4. AdminPC: Host zariadenie = bude mať IP adresu z rozsahu WAN rozhrania Firewall-u

V reálnom prostredí by na WAN prepojení boli používané verejné IP adresy. V našom virtuálnom prostredí je WAN sieťou, ktorá je pripojená k Bridge adaptéru.

Inštalácia OPNsense

Obraz je dostupný na: <https://opnsense.org/download/>

1. Základné nastavenie VM (po importe ISO súboru) je treba nastaviť nasledovné:

Name:	OPNsense
Type:	BSD
Subtype:	FreeBSD
Version:	FreeBSD (64-bit)

2. Prvé prihlásenie je možné využitím prihlasovacích údajov: [root:opnsense]
3. Po spustení firewall-u je mu potrebné priradiť sieťové nastavenia na jednotlivé rozhrania:

0) Logout	7) Ping host
1) Assign interfaces	8) Shell
2) Set interface IP address	9) pfTop
3) Reset the root password	10) Firewall log
4) Reset to factory defaults	11) Reload all services
5) Power off system	12) Update from console
6) Reboot system	13) Restore a backup

- WAN rozhranie tým, že je pripojené v Bridged network, získa sieťové nastavenia cez DHCP.
- LAN rozhraniu je potrebné sieťové nastavenia priradiť staticky
2 -> 1 -> N -> 10.10.10.1 -> 24 + nastavenie DNS a DHCP.

```

Enter an option: 2

Available interfaces:

1 - LAN (em0 - static, track6)
2 - WAN (em1 - dhcp, dhcp6)

Enter the number of the interface to configure: 1

Configure IPv4 address LAN interface via DHCP? [y/N] N

Enter the new LAN IPv4 address. Press <ENTER> for none:
> 10.10.10.1

Subnet masks are entered as bit counts (like CIDR notation).
e.g. 255.255.255.0 = 24
      255.255.0.0  = 16
      255.0.0.0    = 8

Enter the new LAN IPv4 subnet bit count (1 to 32):
> 24

```

```

For a WAN, enter the new LAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
> 192.168.2.1

Do you want to use the gateway as the IPv4 name server, too? [Y/n] n
Enter the IPv4 name server or press <ENTER> for none:
> 8.8.8.8

Configure IPv6 address LAN interface via WAN tracking? [Y/n] n
Configure IPv6 address LAN interface via DHCP6? [y/N] N

Enter the new LAN IPv6 address. Press <ENTER> for none:
>

Do you want to enable the DHCP server on LAN? [y/N] y

Enter the start address of the IPv4 client address range: 10.10.10.10
Enter the end address of the IPv4 client address range: 10.10.10.100

Do you want to change the web GUI protocol from HTTPS to HTTP? [y/N] N
Do you want to generate a new self-signed web GUI certificate? [y/N] N
Restore web GUI access defaults? [y/N] y

```

4. V tejto chvíli sa je možné pripojiť na webové rozhranie firewall-u. Pozor, webové rozhranie je dostupné len z LAN siete, v našom prípade na <https://10.10.10.1>

```
You can now access the web GUI by opening
the following URL in your web browser:

https://10.10.10.1

*** OPNsense.localdomain: OPNsense 25.1 (amd64) ***

LAN (em0)      -> v4: 10.10.10.1/24
WAN (em1)      -> v4/DHCP4: 192.168.2.11/24
```

5. Teraz je potrebné spustiť VM Ubuntu Desktop a z nej sa pripojiť na webové rozhranie.

The screenshot shows the OPNsense web interface. The top part displays the login page with the OPNsense logo and the text "Securing networks made easy". The login form has fields for "Username:" (root) and "Password:" (masked with asterisks), and a "Login" button. Below the login page, the dashboard is visible. The dashboard includes a sidebar with navigation links: Lobby, Dashboard, License, Password, Logout, Reporting, System, Interfaces, Firewall, VPN, Services, Power, and Help. The main content area is titled "Lobby: Dashboard" and contains a message: "You are currently running in live media mode. A reboot will reset the configuration. SSH remote login is enabled for the users 'root' and 'installer' using the same password." The dashboard also features several widgets: "System Information" (Name: OPNsense.localdomain, Versions: OPNsense 25.1-amd64, FreeBSD 14.2-RELEASE, OpenSSL 3.0.15, Updates: Click to check for updates, Uptime: 00:20:31), "Memory" (13.29% used), "Disk" (100% used), "Interface Statistics" (a donut chart showing LAN and WAN traffic), "Firewall" (a donut chart showing firewall rules), "Gateways" (LAN_GW (active) 192.168.2.1, WAN_DHCP6 (active) undefined, WAN_DHCP 192.168.2.1), "Services" (System Configuration Daemon, Cron, DHCPv4 Server), and "Traffic Graph" (Traffic In, showing a line graph of traffic over time).

Networking

Pridanie DMZ rozhrania

1. Do Firewall-u je potrebné pridať ešte DMZ rozhranie:

Interfaces

[LAN]

[WAN]

Assignments

Devices

Neighbors

Overview

Settings

Virtual IPs

Wireless

Diagnostics

Firewall

VPN

Services

You are currently running in live media mode. A reboot will reset the configuration. SSH remote login is enabled with the same password.

Interface	Identifier	Device
[LAN]	lan	em0 (08:00:27:03:0b:fd)
[WAN]	wan	em1 (08:00:27:56:63:19)

Save

+ Assign a new interface

Device

em2 (08:00:27:c2:85:3b)

Description

DMZ

Add

2. Novo vytvorené DMZ rozhranie je treba zapnúť:

Interfaces

[DMZ]

[LAN]

[WAN]

Assignments

You are currently running in live media mode. A reboot will reset the configuration. SSH remote login is enabled with the same password.

Basic configuration

Enable

Enable Interface

3. Ďalej je potrebné nakonfigurovať IPv4 adresu

IPv4 Configuration Type

Static IPv4

Static IPv4 configuration

IPv4 address

172.16.10.1

24

IPv4 gateway rules

Disabled

4. Po vykonaní akejkoľvek zmeny je potrebné vykonať **Save** a **Apply changes**.

Výsledný stav konfigurácie rozhraní vyzerá nasledovne:

You are currently running in live media mode. A reboot will reset the configuration. SSH remote login is enabled for the users "root" and "installer" the same password.

Status	Interface	Device	VLAN	Link Type	IPv4	IPv6	Gateway	Routes
🟢	LAN (lan)	em0		static	10.10.10.1/24	fe80::a00:27ff:fe03:bfd/64	192.168.2.1	10.10.10.0/24 192.168.2.1 Expand
🟢	WAN (wan)	em1		dhcp	192.168.2.11/24	fe80::a00:27ff:fe56:6319/64	192.168.2.1	default 192.168.2.0/24 Expand
🟢	DMZ (opt1)	em2		static	172.16.10.1/24			172.16.10.0/24

Zariadenia z LAN siete dokážu komunikovať do internetu. Firewall vykonáva PAT. Zároveň zariadenie z LAN siete dokáže pristupovať k zdrojom v DMZ, avšak v opačnom smere to neplatí.

5. Nastavenie DNS servera je možné vykonať v konfigurácii Services

Services

- Captive Portal
- DHCRelay
- Dnsmasq DNS

DNS servers 8.8.8.8

6. Teraz je možné overiť funkčnosť komunikácie z vnútra siete smerom von:

```
rastislav@rastislavdesktop:~/Desktop$ ifconfig | grep 10.10.10
    inet 10.10.10.10 netmask 255.255.255.0 broadcast 10.10.10.255
rastislav@rastislavdesktop:~/Desktop$ ping google.com
PING google.com (142.251.36.142) 56(84) bytes of data:
64 bytes from prg03s12-in-f14.1e100.net (142.251.36.142): icmp_seq=1 ttl=247 time=23.2 ms
64 bytes from prg03s12-in-f14.1e100.net (142.251.36.142): icmp_seq=2 ttl=247 time=15.2 ms
64 bytes from prg03s12-in-f14.1e100.net (142.251.36.142): icmp_seq=3 ttl=247 time=21.1 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 15.155/19.815/23.197/3.405 ms
```

PortForwarding a NAT 1:1

- **1:1 NAT:** umožňuje prístup k špecifickému serveru = k všetkým jeho službám
- **PortForwarding:** umožňuje prístup k špecifickej službe v rámci špecifického servera

V DMZ spustíme VM Ubuntu server, na ktorej bude spustená webová služba. Využitím PortForwarding mechanizmu zabezpečíme jej dostupnosť aj z WAN prostredia - pod IP adresou WAN rozhrania a portom 13080.

1. Najskôr je potrebné staticky priradiť sieťové nastavenia serveru v DMZ:

sudo nano /etc/netplan/network.yaml

```
network:
  version: 2
  ethernets:
    enp0s3:
      dhcp4: no
```

```

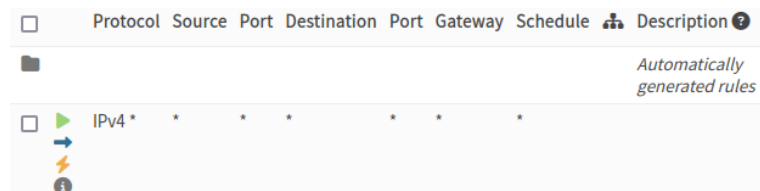
addresses:
- 172.16.10.100/24
gateway4: 172.16.10.1
nameservers:
addresses:
- 8.8.8.8

```

sudo netplan apply

- Po uvedenej konfigurácii dokáže server komunikovať len na úrovni Localhost. Komunikácia mimo DMZ je zatiaľ zakázaná.
- Pre účel inštalácie webového servera (potrebná komunikácia smerom do internetu), je možné vložiť pravidlo povoľujúcu celú komunikáciu vstupujúcu do DMZ rozhrania:

Firewall - Rules - DMZ



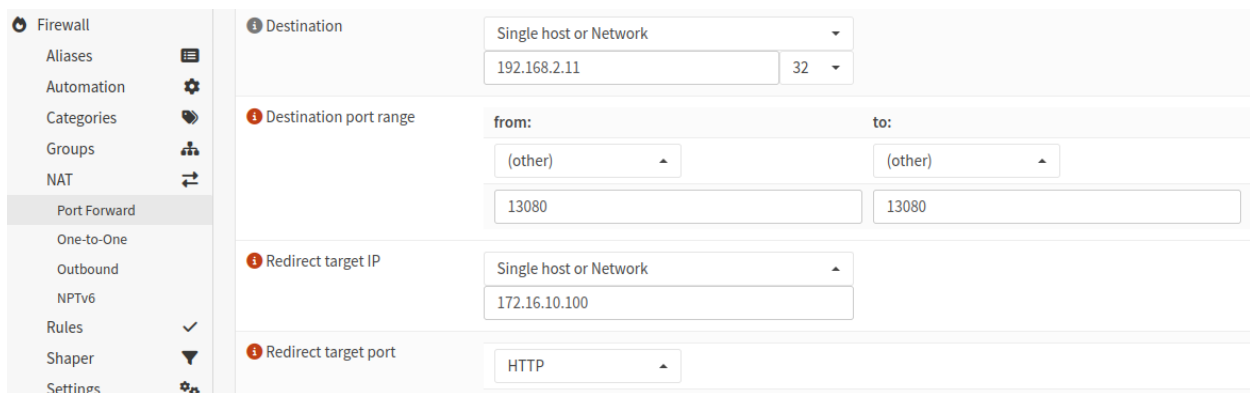
Inštalácia NGINX servera:

```

sudo apt update
sudo apt install nginx

```

- V časti **Firewall - NAT- Port Forward - Add** vykonajme nasledujúcu konfiguráciu:
 - Interface: WAN
 - TCP/IP Version: IPv4
 - Protocol: TCP
 - Source: any any (IP a zdrojový port)
 - Destination: 192.168.2.11 + 13080
 - Redirect target IP: 10.10.10.100 (adresa servera v DMZ)
 - Redirect target port: 80 (port služby, ktorú chceme sprístupniť)
 - Filter rule association: Pass



Port Forward										
One-to-One										
Outbound										
NPTv6										
Rules										
Shaper										
Settings										
Log Files										

Source		Destination		NAT							
Interface	Proto	Address	Ports	Address	Ports	IP	Ports	Description			
LAN	TCP	*	*	LAN address	22, 80, 443	*	*	Anti-Lockout Rule			
WAN	TCP	*	*	192.168.2.11	13080	172.16.10.100	80 (HTTP)				
Enabled rule				No redirect				Linked rule			
Disabled rule				Disabled no redirect				Disabled linked rule			
Alias (click to view/edit)											

Z vonkajšej siete je v tejto chvíli dostupná webová služba:

Not Secure 192.168.2.11:13080

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Podobne je možné povoliť prístup na službu SSH

1. Firewall - NAT- 1:1 NAT - Add
 - a. Interface: WAN
 - b. Address Family: IPv4
 - c. External subnet IP: WAN address alebo 192.168.2.11
 - d. Internal IP: Address - 10.10.10.100
 - e. Destination: 22

Source		Destination		NAT							
Interface	Proto	Address	Ports	Address	Ports	IP	Ports	Description			
LAN	TCP	*	*	LAN address	22, 80, 443	*	*	Anti-Lockout Rule			
WAN	TCP	*	*	192.168.2.11	13080	172.16.10.100	80 (HTTP)				
WAN	TCP	*	*	192.168.2.11	22 (SSH)	172.16.10.100	22 (SSH)				

Overenie dostupnosti služby SSH znázorňuje nasledujúci obrázok:

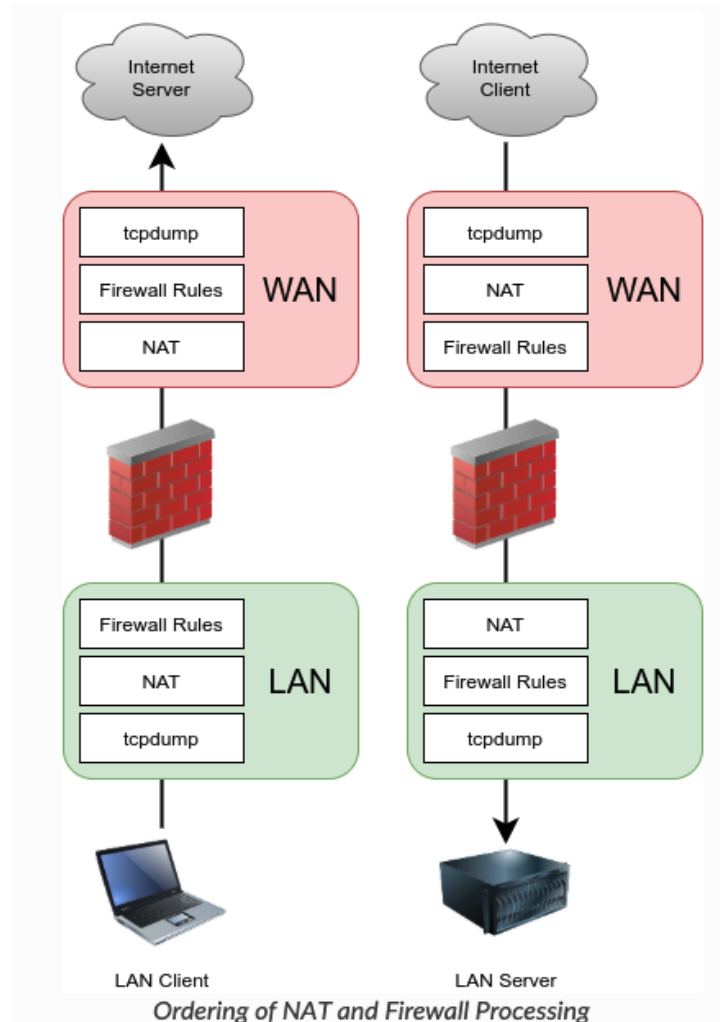
```
rastislavpetija@Rastislavs-iMac ~ % ssh rastislav@192.168.2.11
The authenticity of host '192.168.2.11 (192.168.2.11)' can't be established.
ED25519 key fingerprint is SHA256:ZdWbv6/IGXb9ozcfrjDdy8AxH50TN88tGN7viD5T700.
This host key is known by the following other names/addresses:
~/.ssh/known_hosts:137: 192.168.2.17
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.2.11' (ED25519) to the list of known hosts.
rastislav@192.168.2.11's password:
Last login: Wed Jun 25 17:15:09 2025 from 192.168.2.6
```

Vyhodnocovanie NAT a Firewall pravidiel

Definíciu vyhodnocovania NAT a Firewall pravidiel je možné nájsť v oficiálnej dokumentácii.

Nasledujúci obrázok platí pre firewall pfSense

[<https://docs.netgate.com/pfsense/en/latest/nat/process-order.html>]



Filtrovanie sieťovej prevádzky

Základné správanie

- WAN - implicit deny
- LAN - implicit pass
- Vykonávané podľa poradia záznamov
- NAT je vykonávané ako prvé
- Pass, Block (tiché zahodenie), Reject

Úloha:

- Vytvorte pravidlo, ktoré umožní test dostupnosti na IP adresu 8.8.8.8. Zvyšok komunikácie z DMZ siete nech je zakázaný (toto pravidlo netreba konfigurovať - implicit deny).

The screenshot shows the Mikrotik WinBox interface. At the top, a table lists firewall rules with columns: Protocol, Source, Port, Destination, Port, Gateway, Schedule, and Description. Below this, a rule is configured for IPv4 ICMP traffic to 8.8.8.8. Below the rule configuration, a terminal window shows the execution of ping commands and their statistics.

```

rastislav@rastislavserver:/etc/netplan$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=247 time=18.4 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=247 time=49.7 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=247 time=15.0 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 14.974/27.700/49.733/15.642 ms
rastislav@rastislavserver:/etc/netplan$
rastislav@rastislavserver:/etc/netplan$
rastislav@rastislavserver:/etc/netplan$ ping 8.8.4.4
PING 8.8.4.4 (8.8.4.4) 56(84) bytes of data.
^C
--- 8.8.4.4 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7162ms

```

Logging

Logovať je možné pakety spĺňajúce špecifické filtrovacie pravidlá (Firewall RULES). Vytvorené záznamy je následne možné vidieť v časti Firewall - Log Files - Live view.

The screenshot shows the Mikrotik WinBox interface for Firewall Log Files - Live view. At the top, there is a 'Log' button and a checkbox 'Log packets that are handled by this rule' which is checked. Below this, there is a table with columns: Interface, Time, Source, Destination, Proto, and Label. The table shows several log entries for DMZ and WAN interfaces.

Interface	Time	Source	Destination	Proto	Label
DMZ	→ 2025-06-26T19:11:30	172.16.10.100:46225	8.8.8.8:53	udp	Default deny / state violation rule
DMZ	→ 2025-06-26T19:11:30	172.16.10.100:60454	8.8.8.8:53	udp	Default deny / state violation rule
DMZ	→ 2025-06-26T19:11:30	172.16.10.100:56898	8.8.8.8:53	udp	Default deny / state violation rule
WAN	← 2025-06-26T19:11:26	192.168.2.11	8.8.8.8	icmp	let out anything from firewall host itself (force gw)
DMZ	→ 2025-06-26T19:11:26	172.16.10.100	8.8.8.8	icmp	

VPN

Dôležitou súčasťou zabezpečenia siete je služba VPN, ktorá umožňuje prenášať komunikáciu cez nedôveryhodné siete bezpečným spôsobom.

Remote-Access VPN

- Remote-Access VPN umožní pripojenie vzdialených používateľov do vnútornej siete nachádzajúcej sa za Firewall-om. V rámci tejto ukážky bude znázornený postup, ako umožniť vzdialenému používateľovi (AdminPC) sa pripojiť do LAN siete a komunikovať s Ubuntu-Desktop VM.
- VPN je tiež možné používať za účelom komunikácie s internetom, kde celá komunikácia je smerovaná cez VPN, ktorá daný prenos šifruje. Na základe firemných politík je tiež možné prenášanú komunikáciu kontrolovať.

Prerekvizity:

- WAN IP = 192.168.2.8
- LAN IP GW = 10.10.10.1
- LAN VM = 10.10.10.10
- VPN Tunnel Net: 10.0.200.0/24
- VPN Tunnel GW = 10.0.200.1
- Povoľiť na WAN prepoji komunikáciu z privátnych adries (v našej topológii je WAN z rozsahu 192.168.2.0/24). Táto konfigurácia je možná v časti [**Interfaces - WAN**]

Generic configuration

 Block private networks ☐

 Block bogon networks ☐

1. Vytvorenie Certifikačnej Authority [**System - Trust - Authorities**]

- Create an internal Certificate Authority

Trust

Authorities

Certificates

Revocation

Search

7

☐	Description	Issuer	Usages	Valid from	Valid to
☐	Internal CA	self-signed	0	Jun 26, 2025 7:30 PM	Sep 29, 2027 7:30 PM

2. Vytvorenie certifikátu VPN servera [System - Trust - Certificates]

Method

Create an internal Certificate

Description

VPN Server Certificate

Key

Type

Server Certificate

Private key location

Save on this firewall

Key type

RSA-2048

Digest Algorithm

SHA256

Issuer

Internal CA

Lifetime (days)

397

Common Name

vpn_server_certificate

Trust

Authorities

Certificates

Revocation

Settings

Wizard

Log Files

Certificates

Search

Authority

User client certificate

☐	In use	Description	Issuer	Purpose	Valid from
☐	✓	Web GUI TLS certificate	self-signed	id-kp-serverAuth	Jun 26, 2025 6:36 PM
☐	×	VPN Server Certificate	Internal CA	id-kp-serverAuth	Jun 27, 2025 9:14 AM

3. Vytvorenie používateľa a klientskeho certifikátu [System - Access - Users]

Defined By

user

uid

2000

Disabled

☐

Deny authentication, only applicable for local users

Username

vpn_user_cnl

Password

Scrambled Password

☐

Generate a scrambled password to prevent local database logins for this user.

Full name

vpn_user_cnl

[System - Trust - Certificates]

Method Create an internal Certificate

Description vpn_user_cnl

Key

Type Client Certificate

Private key location Save on this firewall

Key type RSA-2048

Digest Algorithm SHA256

Issuer Internal CA

Lifetime (days) 397

Common Name vpn_user_cnl

Na základe Common Name sa prepojí User a klientský certifikát (musia byť zhodné).

Trust	
Authorities	
Certificates	
Revocation	
Settings	
Wizard	
Log Files	

☐	In use	Description	Issuer	Purpose	Valid from
☒		Web GUI TLS certificate	self-signed	id-kp-serverAuth	Jun 26, 2025 6:36 PM
☐		VPN Server Certificate	Internal CA	id-kp-serverAuth	Jun 27, 2025 9:14 AM
☐		vpn_user_cnl	Internal CA	id-kp-clientAuth	Jun 27, 2025 9:26 AM

4. Vytvorenie konfigurácie VPN servera [VPN - OpenVPN - Server [legacy]]

Tu je potrebné nastaviť:

- Parametre servera (port a IP adresu, na ktorej bude prijímať VPN požiadavky)
- Certifikát servera
- VPN Tunnel informácie + redirect GW (všetka prevádzka pôjde cez tunel)
- DNS server

General information

Disabled ☐

Description vpn_server_legacy

Server Mode Remote Access (SSL/TLS + User Auth)

Backend for authentication Local Database

Enforce local group (none)

Protocol UDP

Device Mode tun

Interface any

Local port 1194

Peer Certificate Revocation List	No Certificate Revocation Lists (CRLs) defined. Create one under System: Certificates .
Server Certificate	VPN Server Certificate (Internal CA) *In Use ▲
Encryption algorithm (deprecated)	None ▼
Auth Digest Algorithm	SHA1 (160-bit) ▲
Certificate Depth	One (Client+Server) ▼
Strict User/CN Matching	☐

Tunnel Settings

IPv4 Tunnel Network	10.0.200.0/24
IPv6 Tunnel Network	
Redirect Gateway	☒
IPv4 Remote Network	
IPv6 Remote Network	
Concurrent connections	
Compression	No Preference ▲
Type-of-Service	☐
Inter-client communication	☒

Client Settings

Dynamic IP	☒
Topology	☐
DNS Default Domain	☐
DNS Domain search list	☐
DNS Servers	☒ Server #1: 8.8.8.8

5. Konfigurácia Firewall pravidiel [Firewall - Rules - OpenVPN]

a. OpenVPN

☐	Protocol	Source	Port	Destination	Port	Gateway	Schedule	Description ?
Automatically generated rules								
☐	IPv4	10.0.200.0/24	*	*	*	*	*	

b. WAN

☐	Protocol	Source	Port	Destination	Port	Gateway	Schedule	Description ?
Automatically generated rules								
☐	IPv4 UDP	*	*	WAN address	1194 (OpenVPN)	*	*	

6. Export VPN klientského certifikátu [VPN - OpenVPN - Client Export]

Remote Access Server	vpn_server_legacy UDP:1194
Export type	File Only
Hostname	192.168.2.8
Port	1194
Use random local port	☒
Validate server subject	☒

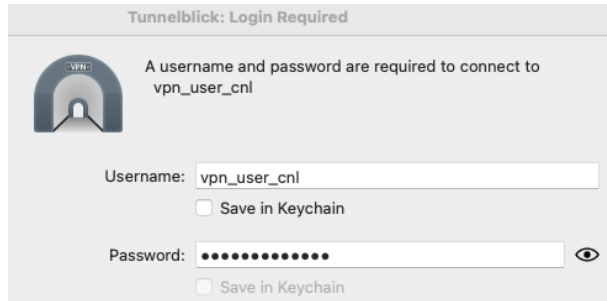
vpn_user_cnl

vpn_user_cnl

Konfiguračný súbor má nasledujúci obsah

```
rastislavpetija@Rastislavs-iMac Desktop % cat vpn_user_cnl.ovpn
dev tun
persist-tun
persist-key
auth SHA1
client
resolv-retry infinite
remote 192.168.2.8 1194 udp
lport 0
verify-x509-name "C=NL, CN=vpn_server_certificate" subject
remote-cert-tls server
auth-user-pass
<ca>
-----BEGIN CERTIFICATE-----
MIIDMzCCAhugAwIBAgIBADANBgkqhkiG9w0BAQsFADANMQswCQYDVQQGEwJOTDAe
```

7. Overenie pripojenia z klientskeho zariadenia [AdminPC]



Výstup príkazu ifconfig obsahuje:

```
utun7: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1500
    inet 10.0.200.6 --> 10.0.200.5 netmask 0xffffffff
```

- Komunikácia na počítač v súkromnej LAN sieti je vďaka VPN možná:


```
rastislavpetija@Rastislavs-iMac Desktop % ping 10.10.10.10
PING 10.10.10.10 (10.10.10.10): 56 data bytes
64 bytes from 10.10.10.10: icmp_seq=0 ttl=63 time=2.290 ms
^C
--- 10.10.10.10 ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 2.290/2.290/2.290/0.000 ms
rastislavpetija@Rastislavs-iMac Desktop %
rastislavpetija@Rastislavs-iMac Desktop %
rastislavpetija@Rastislavs-iMac Desktop % traceroute 10.10.10.10
traceroute to 10.10.10.10 (10.10.10.10), 64 hops max, 52 byte packets
 1  10.0.200.1 (10.0.200.1)  1.663 ms  1.401 ms  1.146 ms
 2  10.10.10.10 (10.10.10.10)  2.381 ms  1.959 ms  1.778 ms
```
- Zároveň komunikácia na externé zariadenia - napr. Google možná je a prechádza cez VPN.


```
rastislavpetija@Rastislavs-iMac Desktop % ping google.com
PING google.com (142.251.36.142): 56 data bytes
64 bytes from 142.251.36.142: icmp_seq=0 ttl=247 time=15.648 ms
^C
--- google.com ping statistics ---
1 packets transmitted, 1 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 15.648/15.648/15.648/0.000 ms
rastislavpetija@Rastislavs-iMac Desktop %
rastislavpetija@Rastislavs-iMac Desktop %
rastislavpetija@Rastislavs-iMac Desktop % traceroute google.com
traceroute to google.com (142.251.36.142), 64 hops max, 52 byte packets
 1  10.0.200.1 (10.0.200.1)  1.820 ms  1.244 ms  1.179 ms
```

8. Overenie nástrojom Wireshark

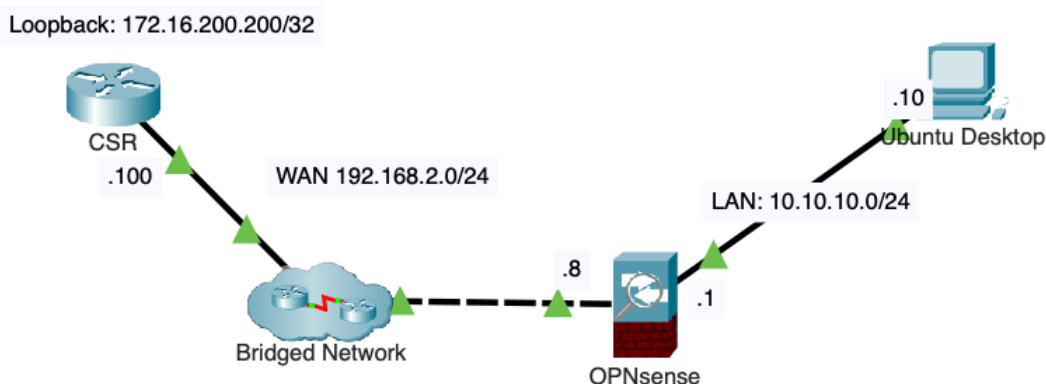
```
Internet Protocol Version 4, Src: 192.168.2.8, Dst: 192.168.2.7
User Datagram Protocol, Src Port: 1194, Dst Port: 65459
OpenVPN Protocol
  > Type: 0x48 [opcode/key_id]
    Peer ID: 0
  > Data (1046 bytes)
    Data [truncated]: 0000050242e0e43d475ecbfa4565d27c8941e6e7353fee8dd9f87593b164b0c332b635b37119f60790322d75add08e6749cdd5e
```

Site-to-Site VPN

- Site-to-Site VPN umožňuje navzájom prepájať produkčné prostredia (Centrála - Pobočka). Koncový používateľ o existencii VPN nevie. VPN tunel sa vytvára medzi hraničným smerovačom a VPN koncentrátorom.

Topológia

Pre ukážku práce s VPN typu Site-to-Site je použitá nasledujúca topológia.



Úloha:

1. Vytvoriť tunel medzi smerovačom CSR a OPNsense zariadením.
2. Umožniť šifrovať komunikáciu medzi Loopback rozhraní (reprezentuje LAN sieť za smerovačom) a zariadením Ubuntu Desktop.

Konfigurácia smerovača

Na smerovači je potrebné vykonať nasledujúcu konfiguráciu:

```
interface Loopback1
ip address 172.16.200.200 255.255.255.255
```

!Konfigurácia IKE fáza 1 - vytvorenie bezpečnostného tunela na výmenu kľúčov a autentifikáciu

```
crypto isakmp policy 1
encr aes
hash sha256
authentication pre-share
group 14
```

!definovanie suseda a bezpečnostného kľúča

```
crypto isakmp key cisco address 192.168.2.8
```

!Konfigurácia IKE fáza 2

!Vytvorenie transformsetu - konfigurácia IPSec

```
crypto ipsec transform-set X esp-aes esp-sha-hmac
```

!Vytvorenie ACL, ktorý identifikuje komunikáciu, ktorú chceme šifrovať

```
access-list 100 permit ip host 172.16.200.200 host 10.10.10.10
```

!Vytvorenie crypto_Mapy - definovanie suseda, aplikovanie transform-setu, DH a prevádzky, ktorá sa bude šifrovať

```
crypto map MAP 10 ipsec-isakmp
 set peer 192.168.2.8
 set transform-set X
 set pfs group14
 match address 100
```

!priradenie crypto_mapy na rozhranie

```
interface GigabitEthernet1
 description VBox
 ip address 192.168.2.100 255.255.255.0
 crypto map MAP
```

!priradenie statickej cesty na dosiahnutie cieľa

```
ip route 10.10.10.10 255.255.255.255 192.168.2.8
```

Kontrola konfigurácie

```
show crypto isakmp policy
show crypto isakmp sa
show crypto map
show crypto ipsec sa
```

Konfigurácia IP-Sec Point-to-Point VPN na OPNsense firewall-e

1. Konfigurácia IKE fáza 1 [VPN - IPsec - Tunnel Settings [legacy]]

General information	
i Disabled	☐ Disable this phase1 entry
i Connection method	default
i Key Exchange version	auto
i Internet Protocol	IPv4
i Interface	WAN
i Remote gateway	192.168.2.100
i Dynamic gateway	☐ Allow any remote gateway to connect
i Description	Site-to-Site VPN Tunnel - F1

Phase 1 proposal (Authentication)

i Authentication method: Mutual PSK

i Negotiation mode: Main

i My identifier: My IP address

i Peer identifier: Peer IP address

i Pre-Shared Key: cisco

Phase 1 proposal (Algorithms)

i Encryption algorithm: AES
128

i Hash algorithm: SHA1, SHA256

i DH key group: 14 (2048 bits)

Phase 1

7

Enabled	Type	Remote Gateway	Mode	Phase 1 Proposal	Commands
☒	IPv4 auto	192.168.2.100		AES (128 bits) + SHA1,SHA256 + DH Group 14	Mut: Site + ✎ 📄 🗑

« < 1 > »

Showing 1 to 1 of 1 entries

2. Konfigurácia IKE fáza 2

General information

i Disabled: ☐

i Mode: Tunnel IPv4

i Description: Site-to-Site VPN Tunnel - F2

Local Network

i Type: Address

i Address: 10.10.10.10 32 ▲

Remote Network

i Type: Address

i Address: 172.16.200.200 32 ▲

Phase 2 proposal (SA/Key Exchange)

Protocol ESP

Encryption algorithms AES128

Hash algorithms SHA1, SHA256

PFS key group 14 (2048 bits)

Lifetime 3600 seconds

Phase 1

Enabled	Type	Remote Gateway	Mode	Phase 1 Proposal	Commands
☒	IPv4 auto	192.168.2.100		AES (128 bits) + SHA1, SHA256 + DH Group 14	Mutl Site +

Showing 1 to 1 of 1 entries

Phase 2

Enabled	Reqid	Type	Local Subnet	Remote Subnet	Phase 2 Proposal
☒	1	ESP IPv4 tunnel	10.10.10.0/24	172.16.200.0/24	AES128 + SHA1, SHA256 + DH Group 14

Showing 1 to 1 of 1 entries

☒ Enable IPsec

3. Pridanie Firewall pravidla na povolenie komunikácie smerujúcej do LAN [Firewall - Rules - IPsec]

☐	Protocol	Source	Port	Destination	Port	Gateway	Schedule	Description
Automatically generated rules								
☐	IPv4 *	*	*	10.10.10.0/24	*	*	*	IPsec to LAN

4. Nastavenie Status-u na Connected [VPN - IPsec - Status Overview]

Phase 1

Search

7

Status	Connection	Version	Local ID	Local IP	Remote ID	Remote IP	Time	Bytes in	Bytes out
	Site-to-Site VP...	IKEv1	192.168.2.8	192.168.2.8	192.168.2.100	192.168.2.100	2029	966.63 KB	1.59 MB

1

Showing 1 to 1 of 1 entries

Phase 2

Search

7

Local subnets	spl-in	spl-out	Remote subnets	State	Time	Bytes in	Bytes out
10.10.10.10/32	cff5c68e	03d85018	172.16.200.200/32	INSTALLED	2029	966.63 KB	1.59 MB

1

Showing 1 to 1 of 1 entries

5. Test dostupnosti (CSR - VM a VM - CSR)

```

CSR1kv#ping
Protocol [ip]:
Target IP address: 10.10.10.10
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Ingress ping [n]:
Source address or interface: 172.16.200.200
DSCP Value [0]:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0x0000ABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.10, timeout is 2 seconds:
Packet sent with a source address of 172.16.200.200
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/9/34 ms

```

```

rastislav@rastislavdesktop:~/Desktop$ traceroute 172.16.200.200
traceroute to 172.16.200.200 (172.16.200.200), 30 hops max, 60 byte packets
 1 _gateway (10.10.10.1)  53.016 ms  3.451 ms  5.259 ms
 2 192.168.2.100 (192.168.2.100)  12.907 ms  *  *

```

V nástroji Wireshark je tiež možné vidieť, že medzi koncovými bodmi VPN tunelu sú prenášané ESP pakety.

```

> Internet Protocol Version 4, Src: 192.168.2.8, Dst: 192.168.2.100
> Encapsulating Security Payload

```

6. Overenie funkčnosti konfigurácie na smerovači

```

show crypto isakmp policy
show crypto isakmp sa
show crypto map
show crypto ipsec sa

```

```

CSR1kv#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst          src          state          conn-id status
192.168.2.100 192.168.2.8  QM_IDLE       1014 ACTIVE

```

```
CSR1k0#show crypto ipsec sa
```

```
interface: GigabitEthernet1
```

```
  Crypto map tag: MAP, local addr 192.168.2.100
```

```
protected vrf: (none)
```

```
local  ident (addr/mask/prot/port): (172.16.200.200/255.255.255.255/0/0)
```

```
remote ident (addr/mask/prot/port): (10.10.10.10/255.255.255.255/0/0)
```

```
current_peer 192.168.2.8 port 500
```

```
  PERMIT, flags={origin_is_acl,}
```

```
  #pkts encaps: 9907, #pkts encrypt: 9907, #pkts digest: 9907
```

```
  #pkts decaps: 9946, #pkts decrypt: 9946, #pkts verify: 9946
```

```
  #pkts compressed: 0, #pkts decompressed: 0
```

```
  #pkts not compressed: 0, #pkts compr. failed: 0
```

```
  #pkts not decompressed: 0, #pkts decompress failed: 0
```

```
  #send errors 0, #recv errors 0
```

```
local crypto endpt.: 192.168.2.100, remote crypto endpt.: 192.168.2.8
```

```
plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet1
```

```
current outbound spi: 0xCFF5C68E(3488990862)
```

```
PFS (Y/N): Y, DH group: group14
```