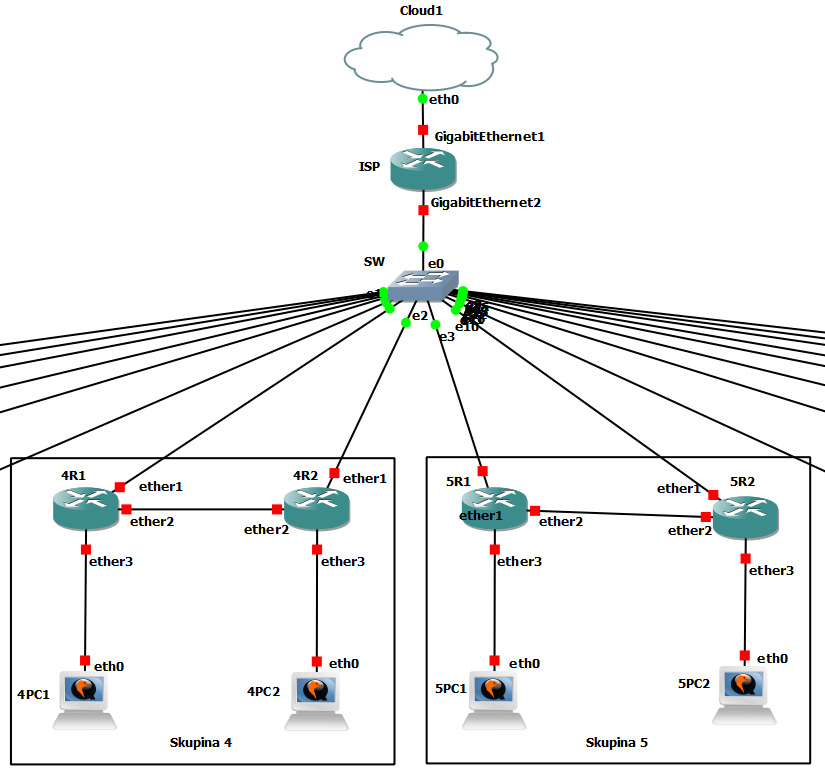
**PS2** / Cvičenie 11 / **Sieťová automatizácia**

**Topológia**

Pracuje sa v **dvojiciach** (pri nepárnom počte študentov pracuje jeden študent sám).  
R1 a R2 sú v každej skupine **Mikrotik** smerovače, ISP je **Cisco** smerovač.

Učiteľ naklonuje **1x** topológiu: kontsek-2021-PS2-CV11-Automatizacia-vzor   
(v GNS3, všetci pracujú v jednej topoĺogii)

Učiteľ si môže všetky vytvorené žiadosti do Postmanu naimportovať zo súborov   
 „PS2\_cv11\_LAB\_Automatizacia\_2021\_04\_28\_MK-cisco.postman\_collection“ a   
 „PS2\_cv11\_LAB\_Automatizacia\_2021\_04\_28\_MK-mikrotik.postman\_collection“ (návod v úlohe 1.h).   
Pozor ale na IP a port v jednotlivých žiadostiach.



**Príklad odporúčanej IP adresácie (vzor pre skupinu 1)**

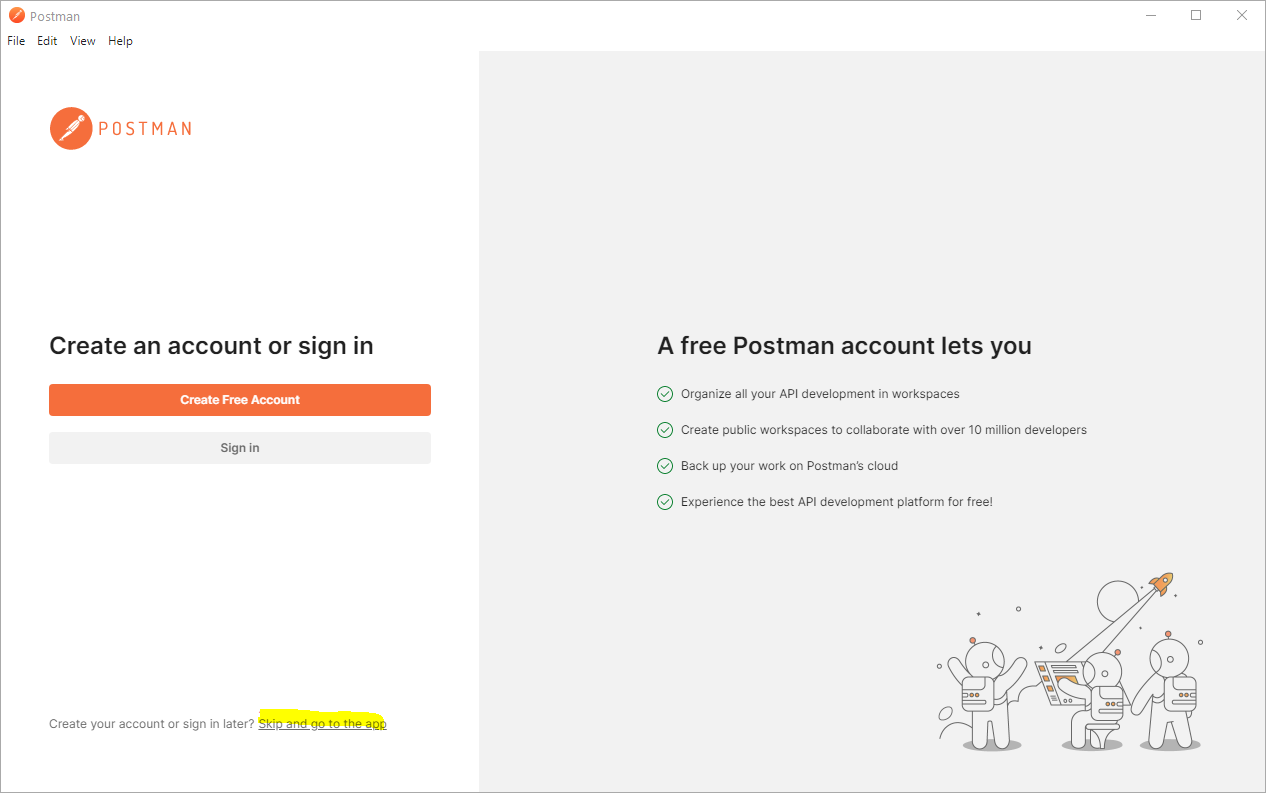
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Device | Interface | IP Address | Subnet Mask | Default Gateway |
| 1R1 | ether1 | 192.168.1.11 | 255.255.255.0 | 192.168.1.1 |
| ether2 | 10.1.12.1 | 255.255.255.0 | N/A |
| ether3 | 10.1.1.1 | 255.255.255.0 | N/A |
| lo0 | 10.1.111.1 | 255.255.255.0 | N/A |
| 1R2 | ether1 | 192.168.1.12 | 255.255.255.0 | 192.168.1.1 |
| ether2 | 10.1.12.2 | 255.255.255.0 | N/A |
| ether3 | 10.1.2.1 | 255.255.255.0 | N/A |
| lo0 | 10.1.222.2 | 255.255.255.0 | N/A |
| 1PC1 | eth0 | od DHCP servera | | |
| 1PC2 | eth0 | od DHCP servera | | |

**Obsah**

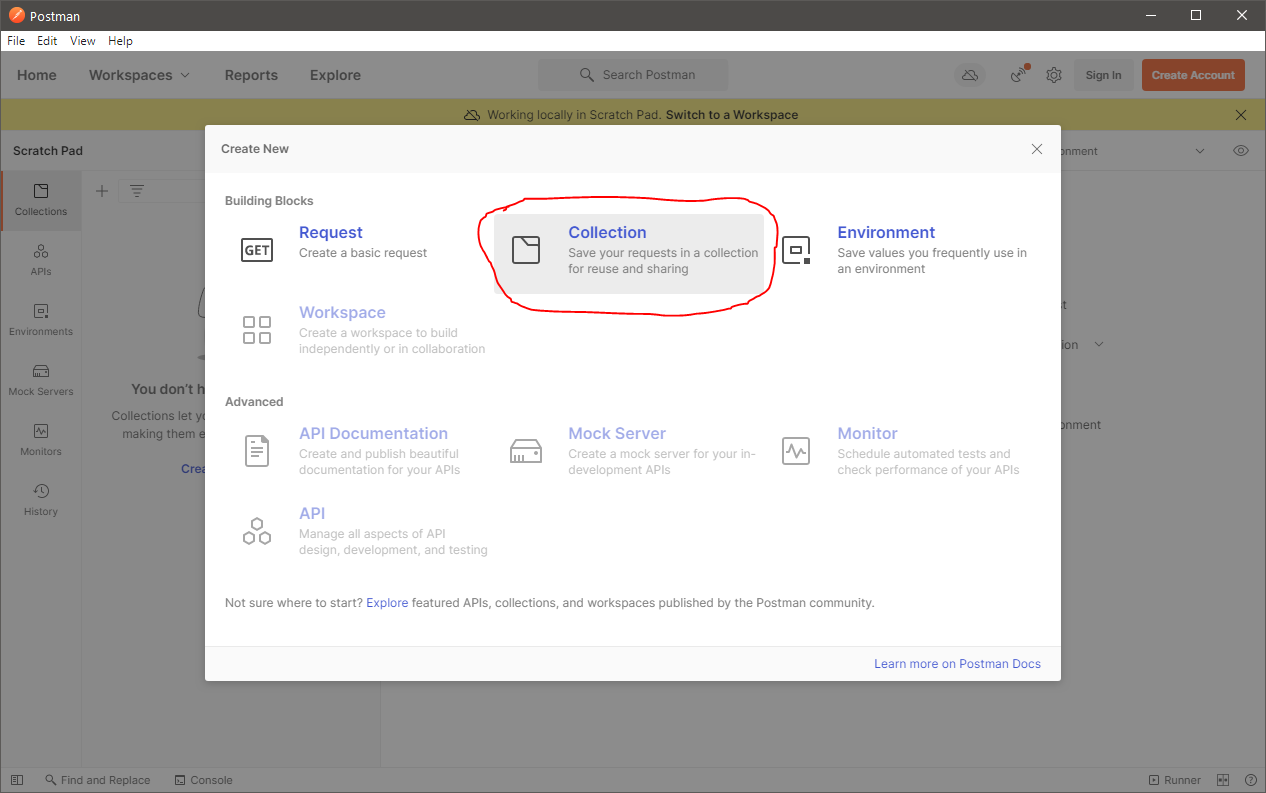
1. Konfigurácia ISP a pridanie mien do topológie
2. Inštalácia nástroja Postman
3. Konfigurácia Cisco smerovača cez RESTCONF
4. Základná konfigurácia Mikrotik smerovača
5. Zapnutie REST API na Mikrotiku
6. Konfigurácia sieťových rozhraní a info o smerovači cez Mikrotik REST API
7. Konfigurácia DHCPv4 cez Mikrotik REST API
8. Konfigurácia NAT cez Mikrotik REST API
9. Konfigurácia OSPFv2 cez Mikrotik REST API
10. Konfigurácia Firewall cez Mikrotik REST API – dobrovoľná úloha (nie je v prednáške)

**Postup:**

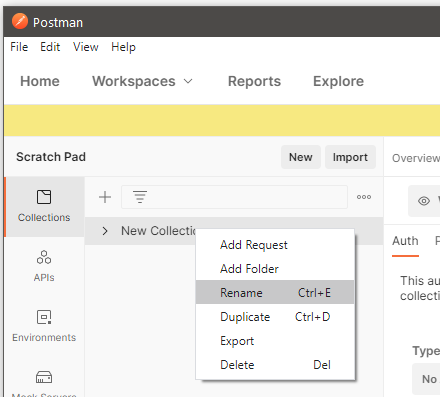
1. **Konfigurácia ISP a pridanie mien do topológie**
   1. Najrýchlejšia skupina nahrá na ISP konfiguráciu, ktorá je na konci tohto zadania a dá vedieť ostatným, že je hotové.
   2. Každý si pridá pod svoju časť topológie mená ľudí, ktorí robia spolu v skupine
2. **Inštalácia nástroja Postman**
   1. Stiahnite si nástroj Postman z adresy <https://www.postman.com/downloads/>
      1. Ak sa vám zobrazila prázdna stránka, vyskúšajte stránku otvoriť v inom prehliadači.
      2. Alebo použite tento link pre 64-bit verziu Postmanu na Windows: <https://dl.pstmn.io/download/latest/win64>
   2. Nainštalujte nástroj Postman otvorením inštalačného súboru a následne ho spustite.
   3. Po spustení nie je potrebné vytvárať bezplatný účet (jeho doplnkové funkcie nebudeme používať). Stačí kliknúť na voľbu „Skip and go to the app“.



* 1. V hlavnom okne si vytvorte 2 kolekcie s názvami Cisco a Mikrotik. Kolekciu vytvoríte pomocou tlačidla New -> Collection.

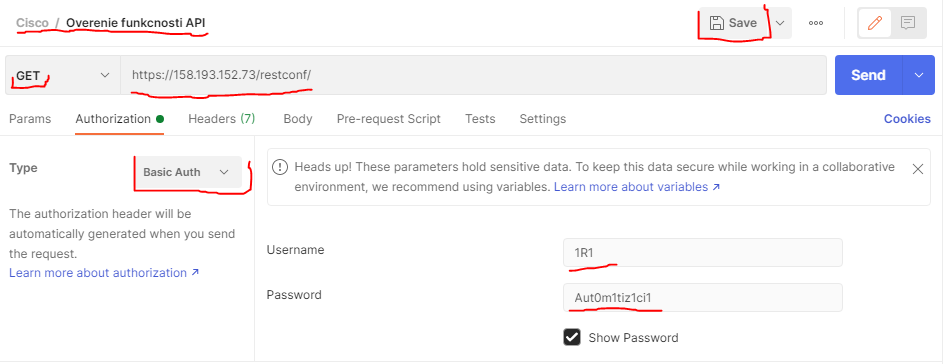


* 1. Vytvorenú kolekciu premenujete pravým kliknutím na názov kolekcie a výberom voľby Rename.

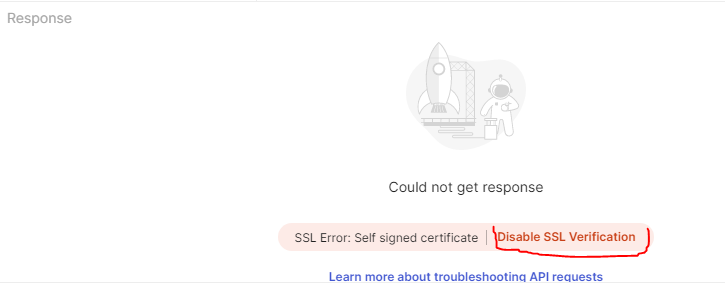


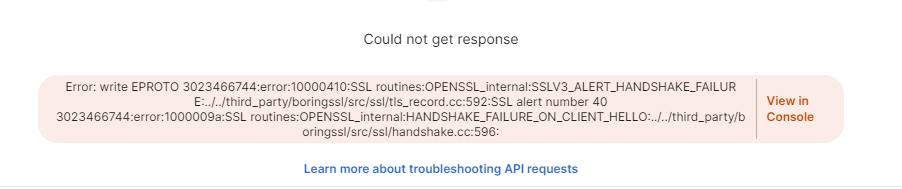
* 1. Do kolekcie je následne možné pridávať Requesty (volania REST API) voľbou New -> Request.
  2. Kolekciu je možné exportovať do textového súboru pravým kliknutím na kolekciu a výberom voľby Export.
  3. Výsledný súbor je následne možné naimportovať pomocou voľby Import.

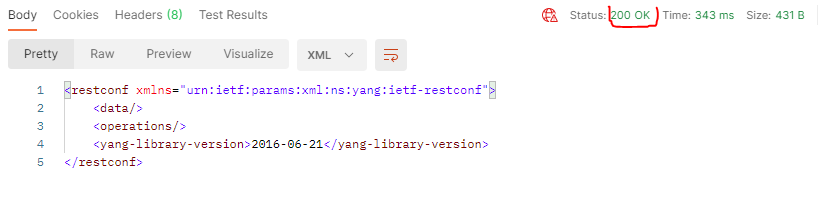
1. **Konfigurácia Cisco smerovača ISP cez RESTCONF**
   1. Vytvorte nový request v kolekcii Cisco (aj všetky ďalšie requesty v tejto úlohe budeme vytvárať v kolekcii Cisco), ktorý pomenujte „Overenie funkcnosti API“.
      1. Typ žiadosti: GET
      2. URL: https://*VEREJNA\_IP\_g1\_ISP*/restconf/
         * Verejnú IP si zistite na ISP (sh ip int br)
      3. Na karte Authorization vyberte typ Basic Auth a zadajte prihlasovacie údaje:
         * meno: hostname vášho smerovača (1R1...)
         * heslo: Aut0m1tiz1ci1
      4. Žiadosť uložte tlačidlom Save.



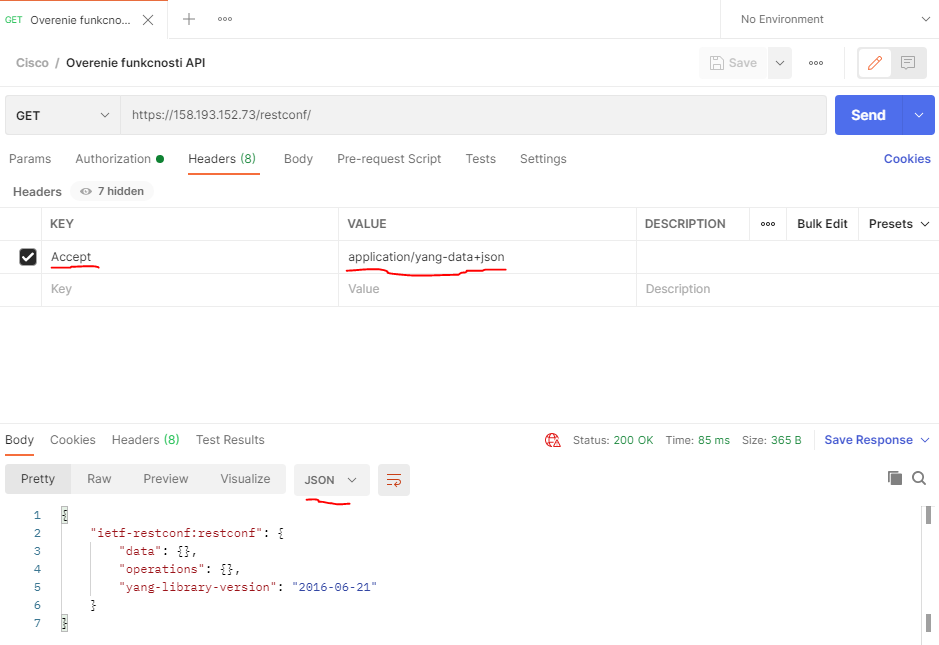
* 1. Žiadosť vykonajte tlačidlom Send. Upozorňujeme na možné chyby a ich riešenie:
     1. ak sa vám objaví chyba SSL error, vyberte voľbu Disable SSL Verification (používame self-signed certifikáty)



* + 1. V jednom prípade pri SSL chybe:  
         
       bolo potrebné doplniť tento príkaz do konfigurácie ISP:  
       ip http authentication local
    2. Ak nepomáha nič z vyššie popísaných riešení, uložte konfiguráciu na ISP a reštartujte ho.
  1. úspešná odpoveď by mala byť indikovaná http odpoveďou 200 OK a v časti response by ste mali vidieť telo vo formáte XML.



* 1. Inštruujte smerovač, aby vám odpoveď vrátil vo formáte JSON.
     1. žiadosť upravte tak, že do časti Headers pridajte kľúč „Accept“ s hodnotou „application/yang-data+json“.
     2. Po odoslaní žiadosti by mala byť odpoveď vo formáte JSON.



* 1. Vytvorte novú žiadosť, ktorá vypíše informácie o sieťových rozhraniach na smerovači:
     1. Názov: Vypis rozhrania
     2. Typ žiadosti: GET
     3. URL: https://VEREJNA\_IP\_g1\_ISP/restconf/data/ietf-interfaces:interfaces
     4. nezabudnite pridať autorizáciu rovnako, ako v predchádzajúcej žiadosti
        + TIP: Autorizáciu môžete nastaviť aj na celú kolekciu (v nastaveniach kolekcie) a potom v žiadosti zadať typ autorizácie Inherit auth from parent
     5. nastavte, aby odpoveď bola vo formáte JSON
     6. žiadosť odošlite a prezrite si, ako vyzerá reprezentácia informácii o sieťových rozhraniach vo formáte JSON.
  2. Vytvorte novú žiadosť, ktorá vypíše informácie o rozhraní Loopback0:
     1. Názov: Vypis lo0
     2. Typ žiadosti: GET
     3. URL: https://VEREJNA\_IP\_g1\_ISP/restconf/data/ietf-interfaces:interfaces/interface=Loopback0
     4. nezabudnite pridať autorizáciu rovnako, ako v predchádzajúcej žiadosti
     5. nastavte, aby odpoveď bola vo formáte JSON
     6. odpoveď preskúmajte, a JSON text si skopírujte na použitie v ďalšej úlohe.
  3. Vytvorte novú žiadosť, ktorá vytvorí nové rozhranie Loopback#R:
     + - # - číslo skupiny
       - R – číslo smerovača
     1. Názov: Vytvor lo#R
     2. Typ žiadosti: PUT
     3. URL: https://VEREJNA\_IP\_g1\_ISP/ restconf/data/ietf-interfaces:interfaces/interface=Loopback#R
     4. nezabudnite pridať autorizáciu rovnako, ako v predchádzajúcej žiadosti
     5. nastavte, aby odpoveď bola vo formáte JSON
     6. Inštruujte smerovač, že mu posielate údaje vo formáte JSON
        + pridajte do Headers kľúč „Content-Type“ s hodnotou „application/yang-data+json“
     7. pridajte telo žiadosti (karta Body), typ „raw“, tak že vložíte skopírovanú JSON odpoveď z minulej žiadosti a upravíte ju nasledovne:
        + názov rozhrania: Loopback#R
        + popis rozhrania: vymyslite si nejaký reťazec
        + ip adresa: 192.0.2.#R
     8. žiadosť odošlite, odpoveď by mala byť typu 201 Created.
     9. overte, že sa rozhranie s požadovanými parametrami správne vytvorilo tak, že zavoláte vopred vytvorenú žiadosť „Vypis rozhrania“.
  4. Vytvorte novú žiadosť, ktorá vymaže rozhranie Loopback#R:
     + - # - číslo skupiny
       - R – číslo smerovača
     1. Názov: Vymaz lo#R
     2. Typ žiadosti: DELETE
     3. URL: https://VEREJNA\_IP\_g1\_ISP/ restconf/data/ietf-interfaces:interfaces/interface=Loopback#R
     4. nezabudnite pridať autorizáciu rovnako, ako v predchádzajúcej žiadosti
     5. nastavte, aby odpoveď bola vo formáte JSON
     6. žiadosť odošlite, odpoveď by mala byť typu 204 No Content.
     7. overte, že sa rozhranie správne vymazalo tak, že zavoláte vopred vytvorenú žiadosť „Vypis rozhrania“.

1. **Základná konfigurácia Mikrotik smerovača**
   1. Prihláste sa na svoj smerovač pomocou konzoly GNS3.
      1. meno: admin
      2. prázdne heslo
   2. Odporúčanie: všetky konfiguračné zmeny si overujte pomocou print príkazov, napr:

ip address print

* 1. Vymažeme konfiguráciu DHCP klienta na rozhraní ether1

ip dhcp-client remove 0

* 1. Na rozhranie ether1 na smerovačoch R1 a R2 nakonfigurujte statickú IP v tvare: 192.168.1.#R/24 – kde # je číslo skupiny a R číslo smerovača.

ip address add address=192.168.1.11/24 interface=ether1

* 1. Pridajte statickú default route na next hop 192.168.1.1

ip route add dst-address=0.0.0.0/0 gateway=192.168.1.1

* 1. Nakonfigurujte adresu DNS servera na 9.9.9.9.

ip dns set servers=9.9.9.9

* 1. Otestujte konektivitu na internet zo smerovačov R1 a R2
     1. IP aj DNS
  2. Nastavte hostnames #R1, #R2, za # dajte číslo skupiny.

system identity set name=1R1

* 1. Zmeňte heslo pre používateľa admin (napríklad na: class)

user set admin password=class

* 1. Zakážte nepotrebné manažment služby v časti IP services (telnet, ftp, api, api-ssl)

ip service disable telnet,ftp,api,api-ssl

* 1. Vypíšte si celú doterajšiu konfiguráciu (príkaz /export) a skontrolujte, či obsahuje zadané konfiguračné príkazy
  2. Overte, či cez protokol CDP (alebo LLDP/MNDP) vidíte susedné smerovače. Mali by ste ich vidieť za rozhraniami podľa topológie a taktiež ich IP adresu nakonfigurovanú na rozhraní k vám (ip neighbor print detail).

1. **Zapnutie REST API na Mikrotiku**

Pre zapnutie REST API je potrebné povoliť https prístup na Mikrotik, čo vyžaduje získanie SSL certifikátov. My si ich vygenerujeme priamo na smerovači.

* 1. Vytvorte si certifikát certifikačnej autority:

/certificate add name=LocalCA common-name=LocalCA key-usage=key-cert-sign,crl-sign

* 1. Certifikát podpíšte.

/certificate sign LocalCA

* 1. Vytvorte certifikát pre smerovač.

/certificate add name=REST common-name=192.168.1.#R

* 1. Certifikát podpíšte certifikačnou autoritou.

/certificate sign REST ca=LocalCA

* 1. povoľte HTTPS a nastavte mu vygenerovaný certifikát.

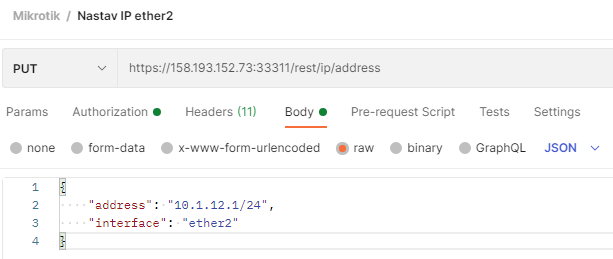
/ip service set www-ssl certificate=REST disabled=no

* 1. Overte, či je HTTPS zapnuté cez webový prehliadač.
     1. URL: https://VEREJNA\_IP\_g1\_ISP:333#R
     2. Mali by ste vidieť WebFig rozhranie. Skúste sa do neho prihlásiť.

1. **Konfigurácia sieťových rozhraní a info o smerovači cez Mikrotik REST API**

Plnú dokumentáciu k Mikrotik REST API nájdete tu: https://help.mikrotik.com/docs/display/ROS/REST+API

* 1. V kolekcii Mikrotik, vytvorte zložku (pravý klik na kolekciu – Add Folder) s číslom a názvom úlohy (5. Konfiguracia sietovych rozhrani). Aj všetky ďalšie žiadosti vytvárajte v kolekcii Mikrotik a organizuje do zložiek podľa číselných úloh.
  2. Vytvorte novú žiadosť, ktorá vypíše informácie o sieťových rozhraniach na smerovači:
     1. Názov: Vypis rozhrania
     2. Typ žiadosti: GET
     3. URL: https://VEREJNA\_IP\_g1\_ISP:333#R/rest/ip/address
     4. nezabudnite pridať autorizáciu rovnako, ako v prípade cisco smerovača
        + meno a heslo použite na prístup do mikrotiku
        + TIP: Autorizáciu môžete nastaviť aj na celú kolekciu (v nastaveniach kolekcie) a potom v žiadosti zadať typ autorizácie Inherit auth from parent
     5. žiadosť odošlite a prezrite si, ako vyzerá reprezentácia informácii o sieťových rozhraniach vo formáte JSON. Žiadosť si skopírujte do ďalšieho kroku.
  3. Vytvorte novú žiadosť, ktorá nastaví IP adresu 10.#.12.R/24 na rozhranie ether2:
     1. Názov: Nastav IP ether2
     2. Typ žiadosti: PUT
     3. URL: https://VEREJNA\_IP\_g1\_ISP:333#R/rest/ip/address
     4. nezabudnite pridať autorizáciu
     5. pri Mikrotik žiadostiach vždy pridávajte parametre do Headers:
        + kľúč „Content-Type“ hodnota „application/json“
        + kľúč „Accept“ hodnota „application/json“
     6. telo žiadosti (karta Body, typ raw) vytvorte na základe predchádzajúcej žiadosti. Parameter \*id neuvádzajte (vygeneruje ho smerovač – nezodpovedá indexu v CLI). Stačí uviesť iba parametre, ktoré sú požadované v CLI.
     7. Príklad tela pre 1R1:



* + 1. žiadosť odošlite a odpoveď by mala byť typu 201 Created
    2. v tele odpovede by ste mali vidieť plnú JSON reprezentáciu konfigurácie
    3. konfiguráciu overte zavolaním žiadosti „Vypis rozhrania“
  1. Vytvorte novú žiadosť s menom „Nastav IP ether3“, ktorá nastaví IP adresu 10.#.R.1/24 na rozhranie ether3
  2. Vytvorte novú žiadosť s menom „System info“, ktorá vypíše informácie o systéme (ekvivalent príkazu system resource print)
  3. Vytvorte novú žiadosť s menom „Vytvor lo0“, ktorá vytvorí rozhranie bridge s názvom lo0.
  4. Vytvorte novú žiadosť s menom „Nastav IP lo0“, ktorá nastaví IP adresu 10.#.0.1/24 na rozhranie lo0
     + - poznamenajte si ID nastavenej IP adresy (budete ho potrebovať na ďalšiu žiadosť)
  5. Vytvorte novú žiadosť s menom „Oprav IP lo0“, ktorá upraví IP adresu na rozhraní lo0 na 10.#.RRR.R/24.
     1. Názov: Oprav IP lo0
     2. Typ žiadosti: PATCH
     3. URL: https://VEREJNA\_IP\_g1\_ISP:333#R/rest/ip/address/\*ID
        + nahraďte ID s číslom ID z predchádzajúceho requestu
     4. do tela vám stačí špecifikovať parameter address
  6. Vytvorte novú žiadosť s menom „Vymaz IP lo0“, ktorá vymaze IP adresu na rozhraní.
     1. Názov: Vymaz IP lo0
     2. Typ žiadosti: DELETE
     3. URL: https://VEREJNA\_IP\_g1\_ISP:333#R/rest/ip/address/\*ID
        + nahraďte ID s číslom ID z predchádzajúceho requestu
     4. telo by malo byť prázdne
     5. Odpoveď by mala byť typu 204 No Content
  7. Overte konfiguráciu žiadosťou „Vypis rozhrania“

1. **Konfigurácia DHCPv4 cez Mikrotik REST API**

Všetky žiadosti z tejto úlohy ukladajte do zložky s názvom „6. DHCPv4“.

* 1. Nastavte DHCP server na vašom smerovači, aby prideľoval IP adresy pre priamo pripojený PC
     1. R1, R2 – vytvorte pool pre počítače na priamo pripojeniej sieti
        + názov žiadosti: Vytvor pool
     2. R1, R2 – vytvorte definíciu siete pre dhcp server a zadefinujte gateway a ľubovoľný verejný DNS server
        + názov žiadosti: Vytvor siet
     3. R1, R2 – vytvorte definíciu samotného DHCP servera na rozhraní vedúcemu ku klientovi (ether3) a priraďte mu pool
        + názov žiadosti: Vytvor dhcp-server
  2. Nakonfigurujte počítač na získanie adresy z DHCP a na PC overte získanie IP z poolu
  3. R1,R2 – overte pridelenie IP cez DHCP (zobrazte Lease)
     + - názov žiadosti: Vypis lease
  4. Overte IP konektivitu medzi routrami a medzi PC a GW.
  5. Overte, či cez protokol CDP vidíte počítač (ip neighbor print detail).
     + - názov žiadosti: Vypis susedov

1. **Konfigurácia NAT cez Mikrotik REST API**
   1. Na **R1** a **R2** nastavte **NAT** pre odchádzajúce pakety **do Internetu – PAT s preťažením rozhrania (ether1)** pre privátnu sieť, na ktorej sa nachádza počítač
      * + názov žiadosti: Vytvor pNAT
   2. Otestujte konektivitu na internet z počítača.
2. **Konfigurácia OSPFv2 cez Mikrotik REST API**
   1. Nakonfigurujte OSPFv2 medzi smerovačmi R1 a R2, nastavte router-id na 1.1.1.1(R1) a 2.2.2.2(R2), pričom do procesu pridajte sieť medzi smerovačmi ako aj siete vedúce k PC. Pozor!: na rozdiel informácii v prednáške, ktoré sú pre RouterOS v6, vo verzii 7.1, ktorú máte na cvičení, je potrebné OSPFv2 konfigurovať nasledovne:
      1. inštanciu ospf je treba vytvoriť. Default inštancia tam nie je.
         * názov žiadosti: Vytvor instanciu

routing ospf instance add name=default router-id=1.1.1.1 disabled=no

* + 1. vytvoriť OSPF oblasť 0:
       - názov žiadosti: Vytvor oblast

routing ospf area add instance=default name=backbone area-id=0.0.0.0

* + 1. pridať siete do OSPF procesu
       - názov žiadosti: Pridaj siet ku PC
       - názov žiadosti: Pridaj siet ku susedovi

routing ospf interface-template add network=10.1.1.0/24 area=backbone

routing ospf interface-template add network=10.1.12.0/24 area=backbone

* 1. Overte konektivitu medzi PC1 a PC2.
  2. Skontrolujte, či sú požadované smerovacie záznamy naučené cez OSPF v smerovacej tabuľke
     + - názov žiadosti: Vypis smerovaciu tabulku
  3. Preskúmajte OSFP tabuľku.
     + - názov žiadosti: Vypis OSPF tabulku
  4. Viac informácii o smerovacích protokoloch v RouterOS v7 nájdete tu: https://help.mikrotik.com/docs/display/ROS/ROSv7+Basic+Routing+Examples

1. **Konfigurácia Firewall-u cez Mikrotik REST API – dobrovoľná úloha (nie je v prednáške)**

Návody:

<https://help.mikrotik.com/docs/display/ROS/Building+Your+First+Firewall>

<https://help.mikrotik.com/docs/display/ROS/Basic+Concepts>

<https://help.mikrotik.com/docs/display/ROS/Filter>

Každý na svojom smerovači premyslite konfiguráciu Firewallu a vytvorte žiadosti tak, aby:

* 1. Smerom von zo siete, kde je PC boli povolené z aplikačných služieb iba:
     1. HTTPs kamkoľvek
     2. ICMP kamkoľvek
     3. SSH iba na internet a na lokálny smerovač (PC1 na R1)
     4. Nezabudni zvážiť, čo všetko ti v sieti okrem toho ešte beží, a je nutné, aby to ACL neblokoval, ale povoľoval:
        + Hints:
          - Klienti dostávajú IPv4 adresy dynamicky
          - Chcete využívať pri browsovaní aj doménové mená
          - Prípadne iné...?
  2. Smerom dnu do vnútornej siete: len už vytvorené spojenia (established)
  3. Ochrániť smerovač samotný: povoliť dnu len SSH, WinBox, WebFig a už vytvorené spojenia z vnútra

**Konfigurácia ISP:**

conf t

hostname ISP

int lo0

ip add 192.0.2.1 255.255.255.255

desc Novy loopback

int gi1

ip add dhcp

ip nat outside

no shut

int gi2

ip add 192.168.1.1 255.255.255.0

ip nat inside

no shut

access-list 1 permit 192.168.1.0 0.0.0.255

ip nat inside source list 1 interface gigabitEthernet 1 overload

ip nat inside source static tcp 192.168.1.11 8291 interface gi 1 11111

ip nat inside source static tcp 192.168.1.11 22 interface gi 1 22211

ip nat inside source static tcp 192.168.1.11 443 interface gi 1 33311

ip nat inside source static tcp 192.168.1.12 8291 interface gi 1 11112

ip nat inside source static tcp 192.168.1.12 22 interface gi 1 22212

ip nat inside source static tcp 192.168.1.12 443 interface gi 1 33312

ip nat inside source static tcp 192.168.1.21 8291 interface gi 1 11121

ip nat inside source static tcp 192.168.1.21 22 interface gi 1 22221

ip nat inside source static tcp 192.168.1.21 443 interface gi 1 33321

ip nat inside source static tcp 192.168.1.22 8291 interface gi 1 11122

ip nat inside source static tcp 192.168.1.22 22 interface gi 1 22222

ip nat inside source static tcp 192.168.1.22 443 interface gi 1 33322

ip nat inside source static tcp 192.168.1.31 8291 interface gi 1 11131

ip nat inside source static tcp 192.168.1.31 22 interface gi 1 22231

ip nat inside source static tcp 192.168.1.31 443 interface gi 1 33331

ip nat inside source static tcp 192.168.1.32 8291 interface gi 1 11132

ip nat inside source static tcp 192.168.1.32 22 interface gi 1 22232

ip nat inside source static tcp 192.168.1.32 443 interface gi 1 33332

ip nat inside source static tcp 192.168.1.41 8291 interface gi 1 11141

ip nat inside source static tcp 192.168.1.41 22 interface gi 1 22241

ip nat inside source static tcp 192.168.1.41 443 interface gi 1 33341

ip nat inside source static tcp 192.168.1.42 8291 interface gi 1 11142

ip nat inside source static tcp 192.168.1.42 22 interface gi 1 22242

ip nat inside source static tcp 192.168.1.42 443 interface gi 1 33342

ip nat inside source static tcp 192.168.1.51 8291 interface gi 1 11151

ip nat inside source static tcp 192.168.1.51 22 interface gi 1 22251

ip nat inside source static tcp 192.168.1.51 443 interface gi 1 33351

ip nat inside source static tcp 192.168.1.52 8291 interface gi 1 11152

ip nat inside source static tcp 192.168.1.52 22 interface gi 1 22252

ip nat inside source static tcp 192.168.1.52 443 interface gi 1 33352

ip nat inside source static tcp 192.168.1.61 8291 interface gi 1 11161

ip nat inside source static tcp 192.168.1.61 22 interface gi 1 22261

ip nat inside source static tcp 192.168.1.61 443 interface gi 1 33361

ip nat inside source static tcp 192.168.1.62 8291 interface gi 1 11162

ip nat inside source static tcp 192.168.1.62 22 interface gi 1 22262

ip nat inside source static tcp 192.168.1.62 443 interface gi 1 33362

ip nat inside source static tcp 192.168.1.71 8291 interface gi 1 11171

ip nat inside source static tcp 192.168.1.71 22 interface gi 1 22271

ip nat inside source static tcp 192.168.1.71 443 interface gi 1 33371

ip nat inside source static tcp 192.168.1.72 8291 interface gi 1 11172

ip nat inside source static tcp 192.168.1.72 22 interface gi 1 22272

ip nat inside source static tcp 192.168.1.72 443 interface gi 1 33372

ip nat inside source static tcp 192.168.1.81 8291 interface gi 1 11181

ip nat inside source static tcp 192.168.1.81 22 interface gi 1 22281

ip nat inside source static tcp 192.168.1.81 443 interface gi 1 33381

ip nat inside source static tcp 192.168.1.82 8291 interface gi 1 11182

ip nat inside source static tcp 192.168.1.82 22 interface gi 1 22282

ip nat inside source static tcp 192.168.1.82 443 interface gi 1 33382

ip nat inside source static tcp 192.168.1.91 8291 interface gi 1 11191

ip nat inside source static tcp 192.168.1.91 22 interface gi 1 22291

ip nat inside source static tcp 192.168.1.91 443 interface gi 1 33391

ip nat inside source static tcp 192.168.1.92 8291 interface gi 1 11192

ip nat inside source static tcp 192.168.1.92 22 interface gi 1 22292

ip nat inside source static tcp 192.168.1.92 443 interface gi 1 33392

lldp run

username 1R1 privilege 15 password 0 Aut0m1tiz1ci1

username 1R2 privilege 15 password 0 Aut0m1tiz1ci1

username 2R1 privilege 15 password 0 Aut0m1tiz1ci1

username 2R2 privilege 15 password 0 Aut0m1tiz1ci1

username 3R1 privilege 15 password 0 Aut0m1tiz1ci1

username 3R2 privilege 15 password 0 Aut0m1tiz1ci1

username 4R1 privilege 15 password 0 Aut0m1tiz1ci1

username 4R2 privilege 15 password 0 Aut0m1tiz1ci1

username 5R1 privilege 15 password 0 Aut0m1tiz1ci1

username 5R2 privilege 15 password 0 Aut0m1tiz1ci1

username 6R1 privilege 15 password 0 Aut0m1tiz1ci1

username 6R2 privilege 15 password 0 Aut0m1tiz1ci1

username 7R1 privilege 15 password 0 Aut0m1tiz1ci1

username 7R2 privilege 15 password 0 Aut0m1tiz1ci1

username 8R1 privilege 15 password 0 Aut0m1tiz1ci1

username 8R2 privilege 15 password 0 Aut0m1tiz1ci1

username 9R1 privilege 15 password 0 Aut0m1tiz1ci1

username 9R2 privilege 15 password 0 Aut0m1tiz1ci1

username 10R1 privilege 15 password 0 Aut0m1tiz1ci1

username 10R2 privilege 15 password 0 Aut0m1tiz1ci1

netconf-yang

restconf