Stavba našeho rodinného domku

… aneb stavebník, ten tvrdý chleba má

≡ Menu

Mikrotik poprvé: základní konfigurace

Původně jsem uvažoval, že článek o MikroTiku napíšu v jednom článku společně s tématem o návrhu domácí sítě, protože tyto věci spolu souvisí, nicméně, byl by to dlouhý článek a tak jsem se rozhodl že návrh sítě a konfiguraci Mikrotiku od sebe oddělím.

V neděli (17.9.2017) jsem potřeboval poprvé přistoupit na Loxone Miniserver, jenže to není tak jednoduché, když na stavbě máte jen rozvaděč, funkční zásuvky a jinak nic. Naštěstí jsem měl již zakoupený router Mikrotik (*RB2011UiAS-RM 5x Gbit LAN, 5x 100 Mbit LAN, microUSB, SFP, do racku*), který se dá použít zároveň jako switch.

Proč MikroTik, a proč zrovna tento? Odpověď je poměrně jednoduchá, protože Mikrotik (jeho Router OS) mi dává neuvěřitelnou svobodu a širokou škálu konfiguračních možností za cenu 2650 Kč i se zárukou 60 měsíců. Se zařízeními této značky mám více než 10 letou zkušenost a prošly mi jich pod rukama desítky. Pravda, na firmě používáme Cloud routery vyšších řad, ale také mi pod rukama prošly levné router boardy a musím říct, že to prostě funguje. Je pravdou, že občas vydají novou verzi Router OS kde sice dvě chyby opraví, ale další tři tam zanesou, v některých věcech jsou zabrždění (např. podpora IKEv2 jim fakt trvala), ale obecně nic lepšího za ty prach yna trhu prostě není.

Hlavní doménou Mikrotiku byly vždy venkovní bezdrátové spoje. S postupem času se produkty posunuly i do vnitřních prostor a také do servroven a datových center. Řešením, jak používat Mikrotik indoor v době kamenné, bylo využití desky (odsud routerboard) pro venkovní použití a vložit ji do plechových chassis, cena ovšem nebyla tak příznivá, jak by domácí uživatel očekával a „user friendly“ toto řešení také nebylo. Přibližně od roku 2011 je tomu naštěstí jinak a dnes Mikrotik nabízí širokou paletu zařízené od klasických routerboardů, přes domácí routery v hezké mílé plastové krabičce, Mini AP, robustní routery a switche. Naštěstí.

A proč zrovna tento model (*RB2011UiAS-RM*)?

- Potřebuji model s montáží do racku
- Nízká spotřeba
- Vyšší nároky na RAM 128 MB; V některých případech mi už 64MB nestačilo

- Možnost rozšíření o SFP, pro případ, že k nám někdy zavítá optický internet
- Plnohodnotný USB port, protože chci používat USB modem jako backup konektivitu a nechci používat ještě USB redukci
- Má PoE port
- Cena

Prvotní konfigurace

> *Původně jsem tuto část chtěl pojmout jako návod, jak nakonfigurovat Mikrotik, ale ono to tak úplně nepůjde, protože můj dům je živý organismus a i konfigurace Mikrotiku se bude měnit a růst v čase, takže sice dnes popíšu, co jsem nakonfiguroval, mnohé z vás to může inspirovat, ale tento článek rozhodně neříká* „jak správně nakonfigurovat Mikrotik„

Zapojení

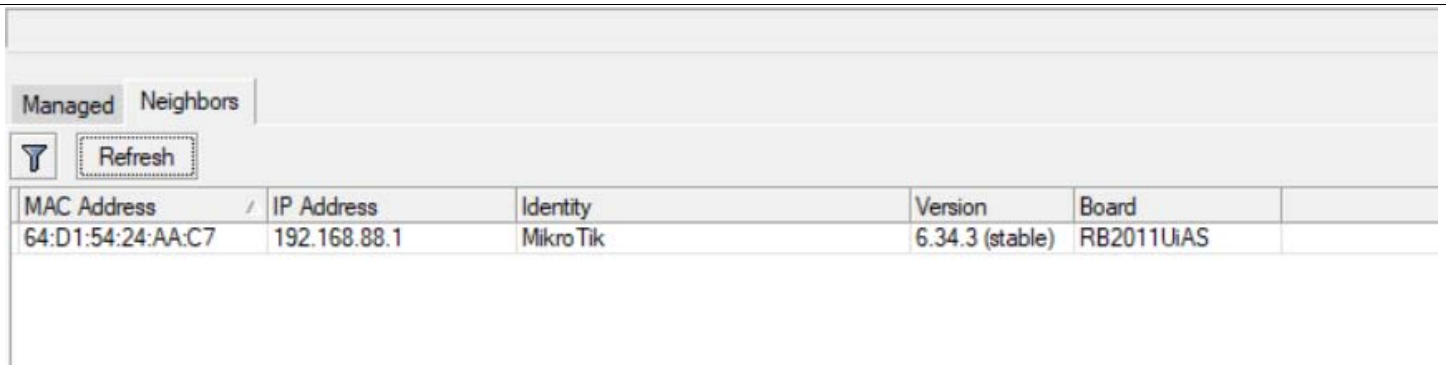
Po rozbalení je dobré Mikrotik zkompletovat, pokud máte rack mounted verzi, je potřeba namontovat i boční úchyty. Všechny router boardy zabírají v rackou 1U.

Po kompletaci je potřeba připojit napájení. Vyšší verze mají zásuvku pro kabel C13/C14 a některé mají dokonce dva zdroje. Můj router má pouze jeden zdroj a „bohužel“ je napájen z externího zdroje. Velice doporučuji zachytit napájecí drát do svorek, čímž se eliminuje riziko „vykopnutí“ napájení. Dobré zejména na stavbě

Winbox

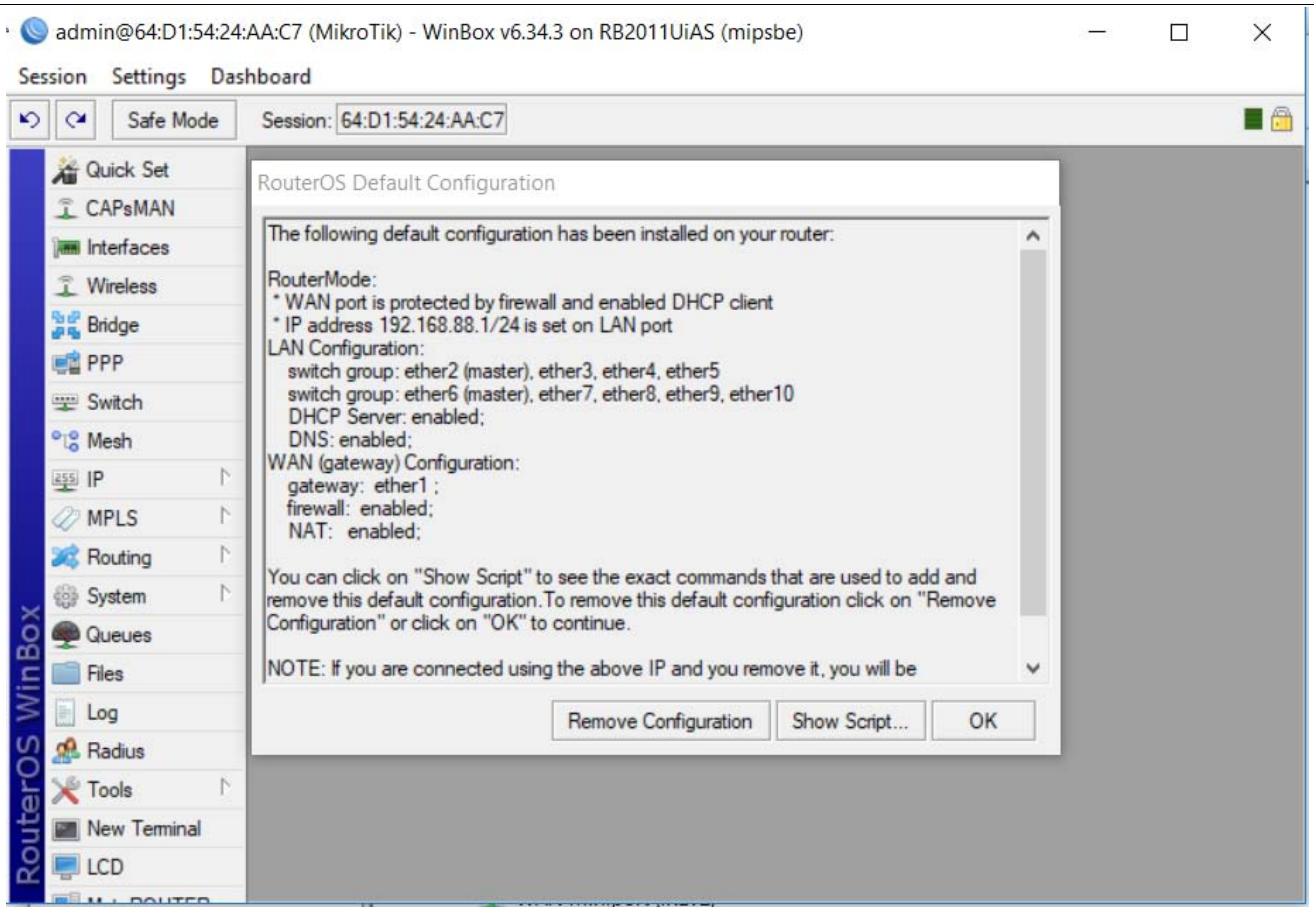
Pro to, abyste mohli konfigurovat Mikrotik, je potřeba si stáhnout nástroj, který se jmenuje Winbox (na webu úplně dole, toho času dostupná verze 3.11). Dále je potřeba připojit počítač do libovolného portu na Mikrotiku. Pozor – libovolný port mimo portu číslo 1, tento port **není určený pro** management. Každé zařízení má u sebe manuálek drobným písmem, kde je popsáno, do kterých portů je a není možné se připojit).

Máme tedy stažený Winbox a počítač připojený do Mikrotiku, například do portu 2. Nyní je potřeba winbox spustit a rozkliknout záložku „Neighbors“, případně si ještě pomoct tlačítkem „refresh“. Mikrotik by se měl zobrazit v seznamu nalezených zařízení. Nyní je potřeba kliknout na MAC adresu, vypllnit jméno admin, heslo ponechat prázdné a připojit se (**Connect**).



> *Na MAC adresu namísto IP adresy jsme kliknuli proto, protože jsme líní a nechce se nám nastavovat IP adresa v našem počítači na adresu z rozsahu 192.168.88.0/24 (Mikrotik má výchozí adresu `192.168.88.1`)*

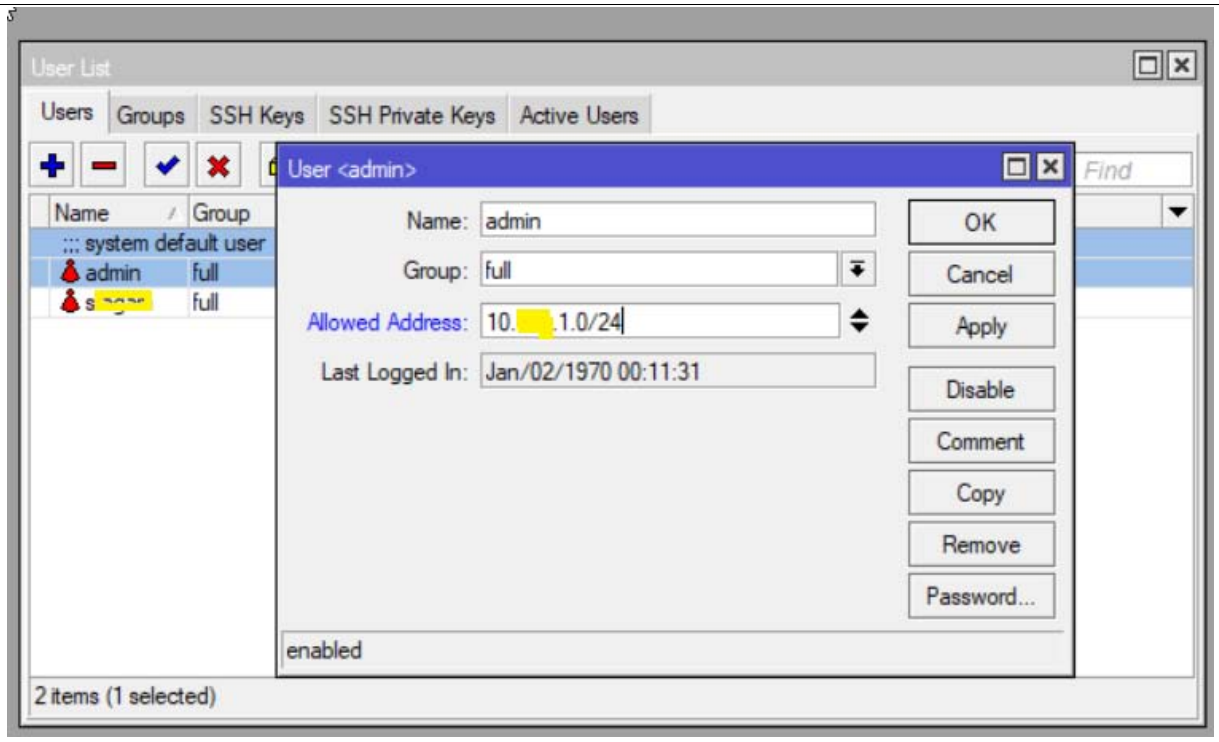
Po té, co se nám podařilo k mikrotiku připojit, zobrazí se nabídka pro založení výchozí konfigurace (RouterOS Default Configuration), kde se popisuje, jaká nastavení výchozí konfigurace obsahuje. Toto je možná dobré pro začátečníky, nicméně nám to komplikuje život tím, že se do konfigurace přidají nesmyslná pravidla, z nichž některé pak není možno smazat. Proto doporučuji použít tlačítko „**`Remove Configuration`**“ a to je vlastně další důvod, proč být připojen pomocí MAC Adresy, protože neřeším, jakoiu IP adresu má Mikrotik, nebo jestli vůbec nějako umá.



Ve chvíli, kdy máme zařízení čisté, je možné pustit se do další konfigurace.

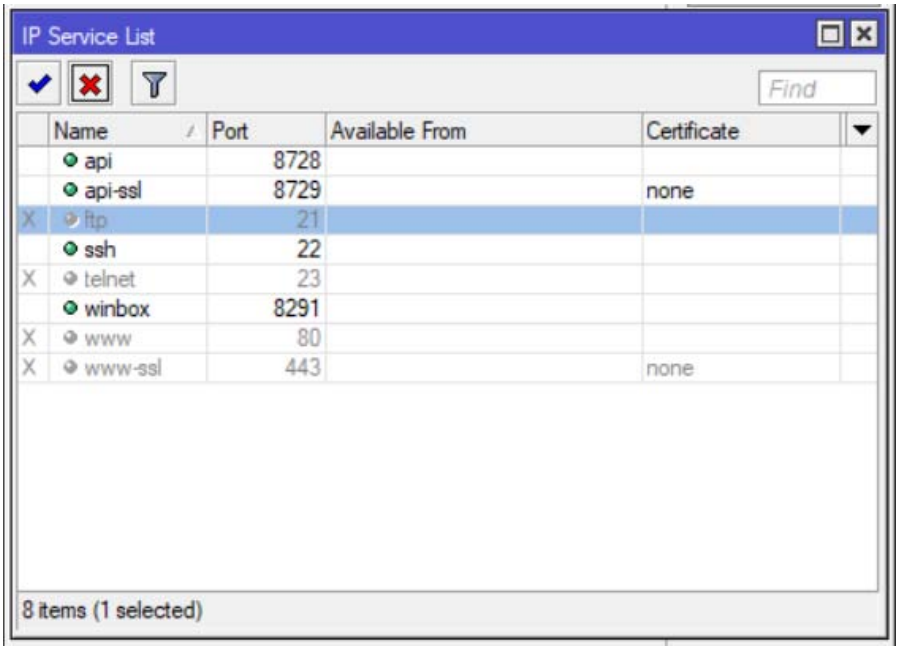
Uživatelé (**`System / Users`**)

- Doporučuji **vytvořit** si `nového uživatele` s právem stejným, jako má admin, tedy Group = full a nastavit bezpečné heslo
- Uživateli `admin` nastavit **bezpečné heslo**
- Uživateli admin nastavit možnost přihlášení pouze z interní adresy (nebo adres), aby nám do tohoto účtu někdo nebušil, až jej vystrčíme do internetu (`Allowed Address`)



Služby Mikrotiku (`IP / Services`)

Zde doporučuji vypnout ty služby, které nejsou vysloveně potřebné, aby byly na Mikrotiku dosažitelné. Osobně nechávám zapnuty pouze služby Winbox a SSH, výjimečně ještě API, ale to většina domácích uživatelů nevyužije. Co rozhodně doporučuji vypínat je www rozhraní a telnet a to z důvodu bezoečnosti.



Pokud by si chtěl někdo ulehčit práci a použít skript (ve Winbox tlačítko New Terminal)

```
/ip service
```

```
set telnet disabled=yes
set ftp disabled=yes
set www disabled=yes
/ip service print
```

Nastavení interního switche (Interfaces)

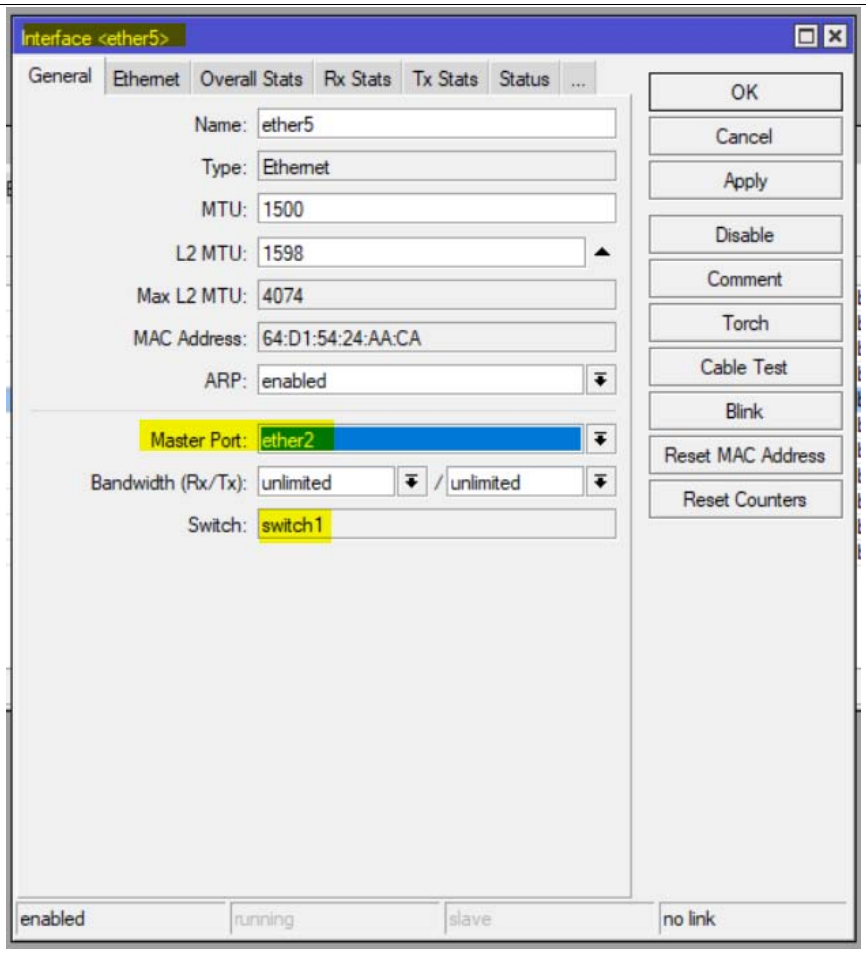
Celou tuhle mašinerii podstupuji jen proto, že se potřebuji připojit k Loxone Miniserveru, který očekává IP adresu z DHCP serveru a sám osobě nefunguje jako switch, takže tyto funkcionality musím „outsourcovat“. Za normálních okolností bych použil hloupý switch, do něj bych připojil Loxone a svůj počítač a možná bych dokázal vnuknout Loxone i nějakou IP adresu, ale switch po ruce nemám. A můj Mikrotik má switche rovnou dva, jeden je pro prvních 5 Gigabitových portů a druhý switch je pro dalších pět 100Mbps portů.

Je potřeba si uvědomit, že fyzický swich jako kus hardware neznamená, že zařízení by se samo o sobě chovalo jako switch. Abychom toho docílili, je potřeba ještě dodatečná konfigurace. Asi by bylo na snadě, že pro konfiguraci switchingu slouží záložka Switch. Bohužel to není tak jednoduché. Toto nastavení obsahuje „pouze“ nastavení tákající se Host tabulek, VLAN a pravidel pro switching. Toť vše.

Správná cesta je přes záložku `Interfaces`. Jak jsem již předesílal, konfigurace, kterou nyní dělám, je dočasná a bude se v průběhu času měnit, i vzhledem s network designu, který jsem si vymyslel. Co chci tedy udělat, propojit fyzické porty na Mikrotiku tak, aby mezi sebou fungovaly jako switch, respektive jako dva switche. Proč dva vysvětlím později.

V záložce `Interfaces` je seznam všech rozhraní, nás zajímají jen fyzické porty, ty mají takovou červeno-černou ikonku ve tvaru < -|- >. Je potřeba dvojklikem rozkliknout postupně ta rozhraní, která `nebudou Master` (v mém případě `ETH3 – ETH5` a `ETH7 – ETH10`) a nastavit u nich příslušný Mater Port (v mém případě `ETH2` nebo `ETH6`).

> ***Pozor! Mater port na rozhraní lze nastavit vždy jen v rámci fyziského switche, to znamená, že pro porty ze switche 2 nebohu nastavit masterport například ETH2.***



Jak mám tedy porty na switchi rozvrženy:

- Fyzický switch 1
  - Port ETH1 ponechám mimo, bude v budoucnu určen pro internetovou konektivitu
  - Port **ETH2** bude Master port
  - Porty ETH3 – ETH5 budou Slave porty s vazbou na Master port 2
- Fyzický switch 2
  - Port **ETH6** bude Master port
  - Porty ETH7 – ETH10 budou Slave porty s vazbou na Master port 6

Druhou variantou je udělat nad porty BRIDGE. Toto řešení ale silně **nedoporučuji**, ikdyž je uživateli hojně využíváno. Jednak proto, protože bridge je čistě softwarová záležitost a switching je záležitost HW. Bridge CPU (to stejné CPU, které řeší i ostatní věci) namísto ASIC.

Nastavení Master port skriptem:

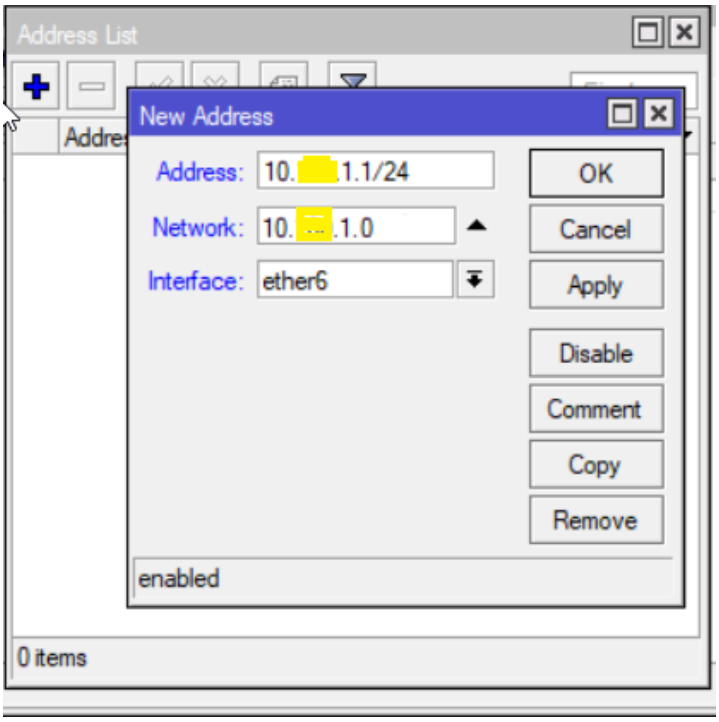
```
/interface ethernet
set ether3 master-port=ether2
set ether4 master-port=ether2
set ether5 master-port=ether2
set ether7 master-port=ether6
set ether8 master-port=ether6
set ether9 master-port=ether6
```

```
8 set ether10 master-port=ether6
```

Nastavení IP adresy rozhraní (IP / Addresses)

Aby byl Mikrotik dostupný na síti, je potřeba mu nastavit IP adresu. IP adresa se vždy přiděluje na konkrétní rozhraní (fyzické, bond, VRRP apod.). V mém případě přidělím IP adresu `10.x.1.1` na rozhraní ETH6, což je Masterport switche. Takže jak na IP adresu. V menu **IP – Addressess** je potřeba kliknout na velké modré plus a správně vyplnit příslušná políčka:

- **Address:** `10.x.1.1/24` – do tohoto políčka je potřeba jednak vyplnit IP adresu Mikrotiku (včetně masky), která bude sloužit zároveň jako výchozí brána. Do budoucna se design ještě změní, protože každá VLAN bude mít vlastní adresu pro výchozí bránu, ale to sem teď motat nebudeme.
  Pozn. Maska 255.255.255.0 = /24
- **Network:** `10.x.1.0` – adresa sítě
- **Interface:** `ether6` – rozhraní, na které bude IP adresa „viset“, v mém případě to je Masterport druhého switche, tedy fyzický port ETH6



Skript:

```
1 /ip address
2 add address=10.x.1.1/24 interface=ether6 network=10.x.1.0
```

Nastavení Firewallu (IP / Firewall)

Předesílám, že toto je pouze základní nastavení firewallu tak, jak jej mám nastaveno já pro své potřeby; Zabezpečení lze určitě ještě zvýšid dalšími sofistikovanými pravidli, já se tímto budu ale zabývat až ve fázi 2, kdy už mě to bude opravdu pálit; Aplikaci pravidel proto provádějte s rozumem na vlastní riziko.

Ještě před tím, než se Mikrotik vůbec připojí do sítě, zejména to platí u připojení k internetu, je důležité Mikrotik zabezpečit. Jedna věc je, že jsme již na začátku vypnuli ty služby, které na Mikrotiku nepotřebujeme. To ale nestačí, je potřeba zabezpečit systém proti útokům zvenku (a v mnohých případech i zevnitř). Zde uvádím jen pár základních tipů, jak router zabezpečit tak, aby mi to v tuto chvíli vyhovovalo. Do budoucna určitě připravím další článek, kde to celé ještě učešu. Ještě doplním, že je potřeba si uvědomit že **pravidla na mikrotiku se zpracovávají odshora dolů, pokud tedy nahoře dáte drop pravidlo, tak na dalším řádku se accept již neprovede** (samozřejmě za daných podmínek, kterým musí pravidlo vyhovět).

„Klientská“ pravidla

- navázaná spojení jdou do FastTrack, což znamená vyšší propustnost, protože firewall bude kontrolovat pouze nová spojení, navázaná bude pouštět rovnou dál
- zahazování invalidních spojení a jejich zalogování
- zahazovat příchozí packety, které nejsou natovány, `ether1` je rozhraní pro veřejný internet; tato akce se zapíše do logu jako „!NAT“
- zahazovat packety, které přijdou z internetu, ale nejsou z veřejných adres, `ether1` je rozhraní pro veřejný internet; tato akce se zapíše do logu jako „!public“
- *zahazovat packety z LAN, které nemají ip adresu z vnitřního rozsahu; vnitřní rozsah v mém případě je 10.x.0.0/20, brigge1 je uvažováno jako rozhraní pro vnitřní síť; toto pravidlo jsem zatím neaplikoval, protože vnitřní sítě nemám ve finálním stavu*

```
/ip firewall filter
add action=fasttrack-connection chain=forward comment=FastTrack
connection-state=established,related
add action=accept chain=forward comment="Established, Related"
connection-state=established,related
add action=drop chain=forward comment="Drop invalid" connection-
state=invalid log=yes log-prefix=invalid
add action=drop chain=forward comment="Drop incoming packets that are
not NATted" connection-nat-state=!dstnat connection-state=new in-
interface=ether1 log=yes log-prefix=!NAT
add action=drop chain=forward comment="Drop incoming from internet which
is not public IP" in-interface=ether1 log=yes log-prefix=!public src-
address-list=not_in_internet
&lt;em&gt;add action=drop chain=forward comment="Drop packets from LAN
that do not have LAN IP" in-interface=bridge1 log=yes log-
prefix=LAN_!LAN src-address=!10.x.0.0/20&lt;/em&gt;

/ip firewall address-list
add address=0.0.0.0/8 comment=RFC6890 list=not_in_internet
add address=172.16.0.0/12 comment=RFC6890 list=not_in_internet
```

```
add address=192.168.0.0/16 comment=RFC6890 list=not_in_internet
add address=10.0.0.0/8 comment=RFC6890 list=not_in_internet
add address=169.254.0.0/16 comment=RFC6890 list=not_in_internet
add address=127.0.0.0/8 comment=RFC6890 list=not_in_internet
add address=224.0.0.0/4 comment=Multicast list=not_in_internet
add address=198.18.0.0/15 comment=RFC6890 list=not_in_internet
add address=192.0.0.0/24 comment=RFC6890 list=not_in_internet
add address=192.0.2.0/24 comment=RFC6890 list=not_in_internet
add address=198.51.100.0/24 comment=RFC6890 list=not_in_internet
add address=203.0.113.0/24 comment=RFC6890 list=not_in_internet
add address=100.64.0.0/10 comment=RFC6890 list=not_in_internet
add address=240.0.0.0/4 comment=RFC6890 list=not_in_internet
add address=192.88.99.0/24 comment="6to4 relay Anycast [RFC 3068]"
list=not_in_internet
```

Ochrana před brute force útoky (FTP, SSH, telnet)

```
/ip firewall filter
add action=drop chain=input comment="drop ftp brute forcers" dst-port=21
protocol=tcp src-address-list=ftp_blacklist
add action=accept chain=output content="530 Login incorrect" dst-
limit=1/1m,9,dst-address/1m protocol=tcp
add action=add-dst-to-address-list address-list=ftp_blacklist address-
list-timeout=3h chain=output content="530 Login incorrect" protocol=tcp
add action=drop chain=input comment="drop ssh brute forcers" dst-port=22
protocol=tcp src-address-list=ssh_blacklist
add action=add-src-to-address-list address-list=ssh_blacklist address-
list-timeout=1w3d chain=input connection-state=new dst-port=22
protocol=tcp src-address-list=ssh_stage3
add action=add-src-to-address-list address-list=ssh_stage3 address-
list-timeout=1m chain=input connection-state=new dst-port=22
protocol=tcp src-address-list=ssh_stage2
add action=add-src-to-address-list address-list=ssh_stage2 address-
list-timeout=1m chain=input connection-state=new dst-port=22
protocol=tcp src-address-list=ssh_stage1
add action=add-src-to-address-list address-list=ssh_stage1 address-
list-timeout=1m chain=input connection-state=new dst-port=22
protocol=tcp
add action=drop chain=input comment="drop telnet brute forcers" dst-
port=23 protocol=tcp src-address-list=black_list
add action=add-src-to-address-list address-list=black_list address-
list-timeout=1d chain=input connection-state=new dst-port=23
protocol=tcp src-address-list=telnet_stage3
add action=add-src-to-address-list address-list=telnet_stage3 address-
list-timeout=1m chain=input connection-state=new dst-port=23
protocol=tcp src-address-list=telnet_stage2
add action=add-src-to-address-list address-list=telnet_stage2 address-
list-timeout=1m chain=input connection-state=new dst-port=23
protocol=tcp src-address-list=telnet_stage1
add action=add-src-to-address-list address-list=telnet_stage1 address-
list-timeout=1m chain=input connection-state=new dst-port=23
protocol=tcp
```

Povolení ICMP (PING)

```
/ip firewall filter
```

```
2 add action=accept chain=input protocol=icmp
```

Pokud by někoho zajímala bezpečnost poněkud více, doporučuji tyhle odkazy:

- https://wiki.mikrotik.com/wiki/Manual:Securing_Your_Router
- https://wiki.mikrotik.com/wiki/Tips_and_Tricks_for_Beginners_and_Experienced_Users_of_RouterOS
- https://wiki.mikrotik.com/wiki/Manual:IP/Firewall/Filter#Basic_examples
- https://wiki.mikrotik.com/wiki/Manual:Securing_Your_Router#Loading_A_Firewall

Nastavení DHCP serveru

Jeden z důvodů, proč jsem vlastně byla potřeba serveru, který by rozdával IP adresy, protože Loxone ve výchozám nastavení očekávám, že mu IP adresa bude přidělena. Toto nastavení osobně nechápu, protože Loxone se většinou osazuje ještě na stavbě, kdy ještě nejsou hotové sítě, takže nastává přesně problém, se kterým se potýkám i já. Kdyby v Loxone nastavili pevnou IP adresu z jakéhokoliv rozsahu, byl by život jednodušší. Takže jak na DHCP server.

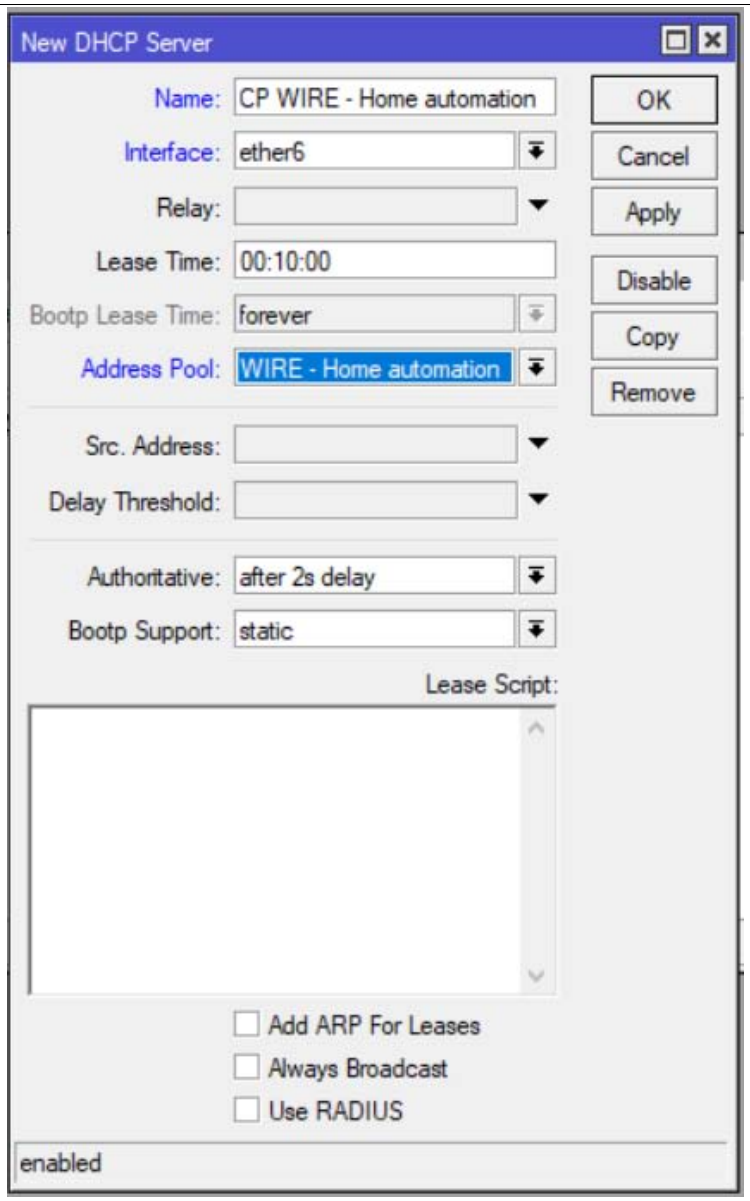
Nejprve je potřeba vytvořit pool, ze kterého budou IP adresy přidělovány. To se dělá v nastavení **`IP / Pool`**. Modrým pluskem se vyvolá okno, do kterého je třeba zadat patřičné parametry:

- **`Name:`** `WIRE – Home automation` – název poolu, já jej pojmenovávám podle názvu VLANy. Tento pool bude v budoucnu odstraněn, protože nechci, aby mi běhalo DHCP zrovna v síti pro automatizaci domu
- **`Addresses:`** `10.x.1.100-10.x.1.200` – rozsah adres, které budou DHCP serverem přidělovány.
- **`Next Pool:`** `none` – netřeba vyplňovat, definuje další pool, ze kterého budou přidělovány adresy v případě, kdy byl aktuální pool vyčerpán a nemá již žádnou volnou adresu

Když je pool vytvořený, přejdeme do nastavení **`IP / DHCP Server`**.

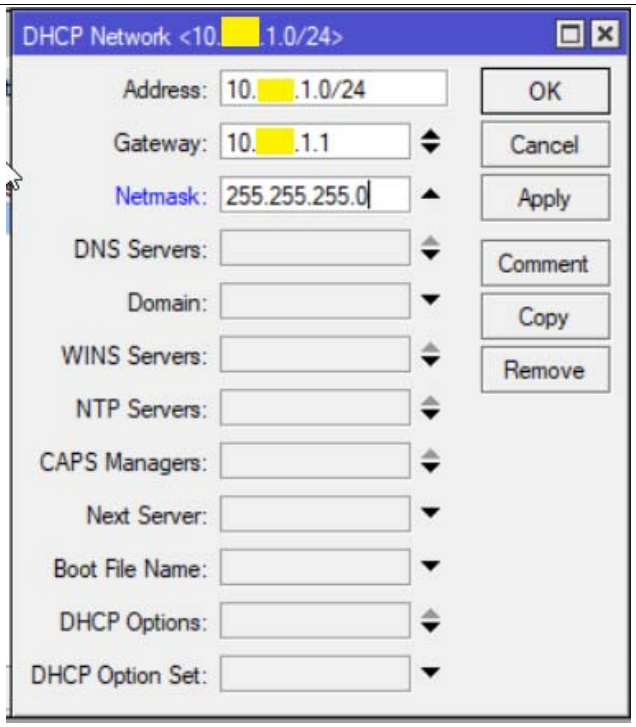
`Nový DHCP server` vytvoříme tak, že v záložce DHCP klikneme na modré plus a do zobrazeného okna vložíme tyto parametry:

- **`Name:`** `DHCP WIRE – Home automation` – název DHCP serveru, já používám opět název VLANy
- **`Interface:`** `ether6` – rozhraní, na kterém poběží DHCP server; ether6 je Master port pro switch 2, takže DHCP bude rozdávat adresy v rámci celého swithe 2
- **`Address Pool:`** `WIRE – Home automation` – vybereme předem vytvořený pool, ze kterého budeme přidělovat adresy.
- Další parametry je možné nechat ve výchozím stavu, k jednotlivým parametrům se vrátím v některém z budoucích článků



Když máme server vytvořený, je potřeba ještě vytvořit příslušnou síť. Je to zvláštně pojmenované, ale vytvořením sítě (záložka **`Network`**) způsobíme, že se pomocí DHCP přidělí klientovi správná maska, výchozí brána, DNS servery a případně další parametry. V záložce **`Network`** tedy opět použijeme modré plus a do nového okna vyplníme následující údaje:

- **`Address`**: `10.x.1.0/24` – adresa sítě včetně masky
- **`Gateway`**: `10.x.1.1` – výchozí brána (v našem případě adresa Mikrotiku)
- **`Netmask`**: `255.255.255.0` nebo `24` – maska sítě (oba formáty jsou akceptovány)
- **`DNS Servers`**: **`8.8.8.8 a 8.8.4.4`** – adresa DNS serverů. Dočasně použijeme servery googlu, v některém z příštích článků si ukážeme, jak si postavit vlastní DNS server.



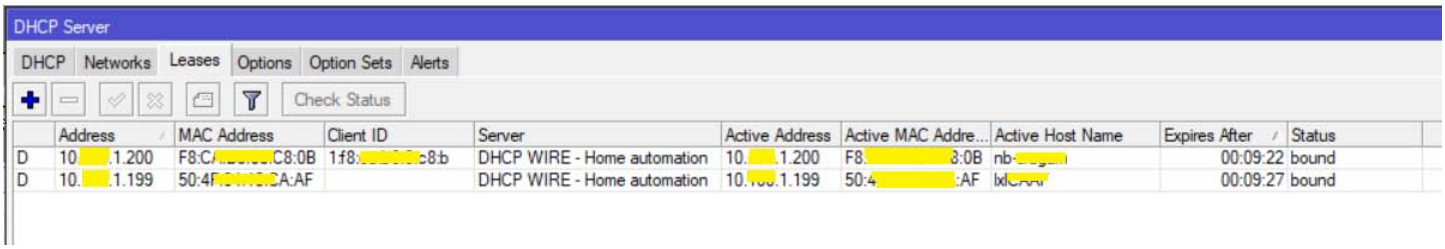
Nastavení DHCP skriptem:

```
/ip pool
add name="WIRE - Home automation" ranges=10.x.1.100-10.x.1.200

/ip dhcp-server
add address-pool="WIRE - Home automation" disabled=no interface=ether6
name="DHCP WIRE - Home automation"

/ip dhcp-server network
add address=10.x.1.0/24 dns-server=8.8.8.8,8.8.4.4 gateway=10.x.1.1
netmask=24
```

Pokud chceme zjistit, jaké IP adresy DHCP server již přidělil, nahlédneme do záložka **Leases**, kde uvidíme seznam přidelených IP adres.



Skript:

```
/ip dhcp-server lease
print
```

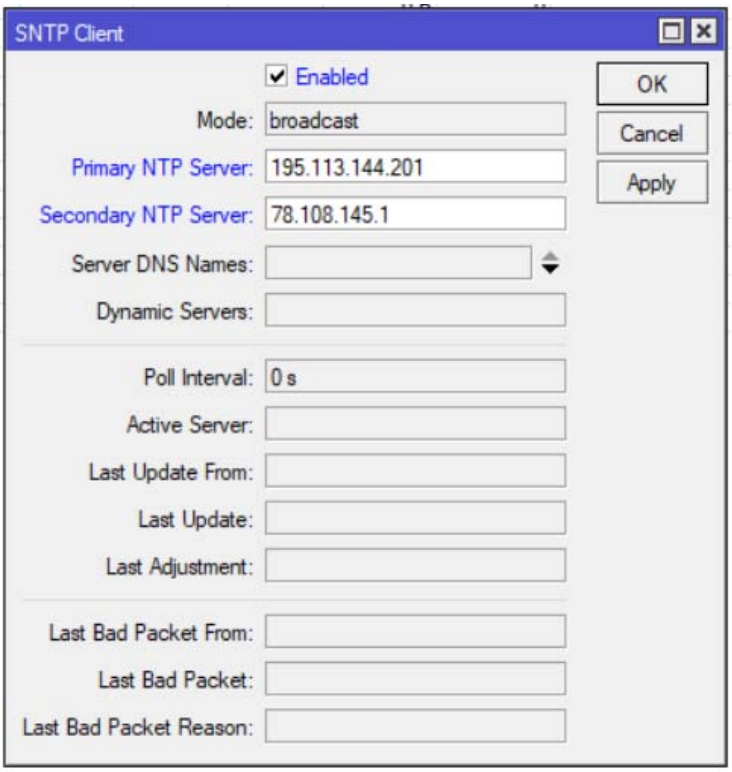
Dáme Mikrotiku čas, nastavení NTP (System / SNTP

Client)

Je žádocí, aby Mikrotik věděl, kolik je hodin. Jednat proto, že z něj později uděláme NTP server (server pro poskytování časových služeb) a také pro to, abychom se mohli vůbec orientovat v LOGu, který je při troubleshooting důležitý.

Adresy časových serverů se nastavuje v menu **`System / SNTP Client`**. Nejprve je potřeba zatrhnout checkbox „`Enabled`„, tím funkci NTP klienta zapneme. Následně vyplníme adresy časových serverů, osobne používám tyto, ale lze použít vaše oblíbené:

- Primary NTP server: 195.113.144.201
- Secondary NTP server: 78.108.145.1



Alternativně lze u novějších verzí Router OS použít DNS názvy NTP serverů, třeba například `ntp.nic.cz a time.windows.com`. Aby Mikrotik adresy těchto serverů správně resolvoval, je potřeba Mikrotiku říct, jaké DNS servery má používat (`IP / DNS`).

Maškaráda – pustíme Loxone na internet?

Tak aby bylo jasno, síť 10.x.1.0/24 na internet pusíme, ale Loxone Miniserver rozhodně do internetu komunikovat nebude, jednak nemám zájem na tom, aby mi někdo natlačil novou verzi (v 9) a za druhé Loxonu už moc nevěřím a nechci, aby Miniserver něco bonzoval domů do centrály.

Povolíme internet:

out interface mám `ppp-out1`, protože používám USB 3G modem pro přístup do internetu, většina z vás bude mít `ether1`, tedy rozhraní do internetu.

```
/ip firewall nat
add action=masquerade chain=srcnat out-interface=ppp-out1 src-
address=10.x.1.0/24
```

Zakážeme miniserveru komunikovat do internetu:

- src-address = adresa Loxone Miniserveru
- log-prefix="LOX OUT" – pokud se Loxone pokusí komunikovat, zapíše se tato aktivita do LOGu

```
/ip firewall filter add action=drop chain=forward log=yes log-prefix="LOX
OUT" out-interface=ppp-out1 src-address=10.x.1.y
```

Tak, to by pro začátek stačilo, příště si povíme něco o zprovoznění USB modemu a 3G internetu. Třeba.

Štítky: DHCP DNS domácí síť maska MikroTik VLAN

Admin 23.9.2017 Networking, TZB komentářů 8

← Návrh sítě pro chytrý dům

Stěhování – první dějství →

8 komentářů u „Mikrotik poprvé: základní konfigurace“

Pingback:Napadený Mikrotik – tuning zabezpečení – Stavba našeho rodinného domku

Pavel
19.6.2018 (17:05)
Trvalý odkaz

Skvělé, děkuji ….

Reagovat

Michal
2.4.2018 (9:34)
Trvalý odkaz

Děkuji za velmi pěkný článek. Bylo by ho možné rozšířit o to, jak propojit switch1 a switch2? Píšete, že Brigde nedoporučujete a kabel zabere 2 porty. A potom aby dhcp fungoval pro oba switche včetně wifi.

Reagovat

Admin Autor příspěvku
8.4.2018 (17:27)
Trvalý odkaz

Zdravím a děkuji za feedback. Od nastěhování je to jeden velký boj a na články se tžko hledá čas. Nějaké články mám zatím jako draft a každým dnem bych je měl zveřejnit. SW1 a SW2 – dobrý nápad na článek, každopádně cesta je, viz článek s tím, že switch1 a switch2 se propojí tak, že se udělá bridge, ale POUZE nad oběma Master porty, nebridgují se všechny porty.

Reagovat

Petr
16.9.2018 (16:37)
Trvalý odkaz

porty 6-10 navážeš na master 2, resp. nnyní už pouze uděláš bridge (bohužel) Ví někdo proč