Obsah

1 Úvod

Konfiguraci síťových prvků provádějí profesionální administrátoři především prostřednictvím příkazového řádku. Tento způsob konfigurace je komplexní, lze s ním nastavit veškeré parametry zařízení. Od administrátora ale očekává znalost přesné syntaxe příkazů. Může být proto obtížnější pro výuku studentů, pro které není snadné soustředit se jak na syntaxi, tak na význam zadávaných příkazů. Tato diplomová práce si klade za cíl prozkoumat softwarové nástroje umožňující konfiguraci síťových zařízení a nevyžadující znalost syntaxe příkazů, nalézt jejich nedostatky a na základě těchto nedostatků navrhnout a realizovat aplikaci či rozšíření aplikace, které tyto nedostatky budou eliminovat.

Teoretická část práce se zabývá nejprve srovnáním vybraných softwarových nástrojů pro konfiguraci síťových prvků výrobců Cisco a Juniper, dvou nejvýznamnějších světových výrobců těchto zařízení. Srovnává jejich možnosti a nedostatky. Další kapitola práce provází konfigurací síťových zařízení Cisco. Komplexně představuje také jeden z nejdůležitějších protokolů internetu, směrovací protokol BGP, včetně jeho konfigurace.

Praktická část provádí návrh aplikace s důrazem na možnost budoucího rozšíření. Seznamuje s výběrem technologií a implementačními detaily, jako jsou diagramy tříd a jejich popis. V další části je provedeno testování funkčnosti aplikace a ověření vygenerovaných konfigurací na reálném operačním systému síťového zařízení.

Závěr shrnuje postup prací a dosažené výsledky.

Práce předpokládá teoretickou znalost nastavovaných parametrů a protokolů.

2 Software pro konfiguraci síťových prvků

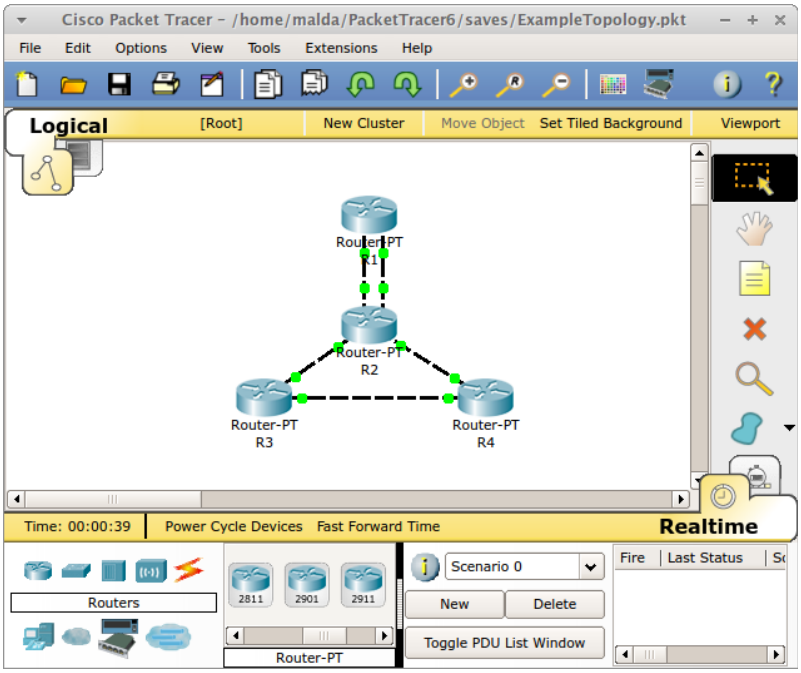
Všichni výrobci enterprise síťových zařízení používají k jejich konfiguraci příkazový řádek, vyžadující znalost syntaxe zadávaných příkazů. Konfiguraci těchto síťových prvků usnadňují nástroje, které umožňují nastavit zařízení i bez znalosti syntaxe – např. v grafickém uživatelském rozhraní.

Pro srovnání byly vybrány nástroje umožňující nastavení síťových prvků dvou největších světových výrobců – společností Cisco a Juniper. Srovnání se zaměřuje zejména na konfiguraci směrovačů, jejich zabezpečení a konfiguraci směrovacích protokolů.

2.1 Cisco Packet Tracer

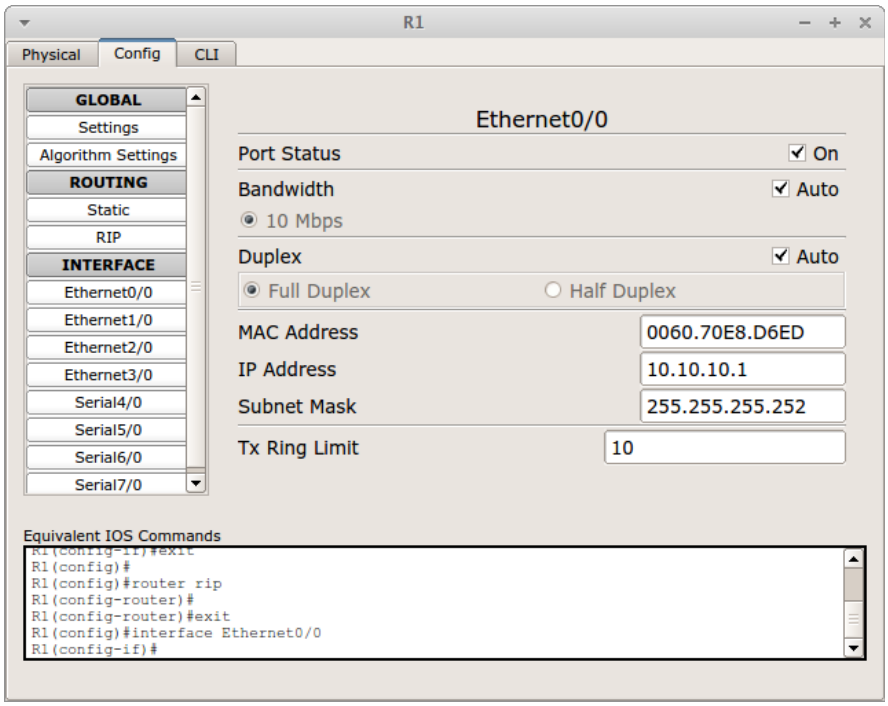
Cisco Packet Tracer je konfigurační a simulační nástroj, vyvinutý společností Cisco převážně pro studijní účely [1]. Není volně dostupný (je dostupný pouze pro osoby zainteresované v programu Cisco Networking Academy). Jedná se o desktopovou aplikaci, testovaná verze 6.0.1 je k dispozici pro platformy Microsoft Windows a Linux. Umožňuje vytvoření virtuální síťové topologie, do které lze vložit několik typů zařízení, zejména směrovačů a přepínačů. Okno aplikace se vzorovou topologií zobrazuje obrázek 2.1.

Konfiguraci zařízení je možné provádět jak z integrované příkazové řádky, tak z grafického uživatelského rozhraní (GUI). Z grafického uživatelského rozhraní je podporována jen základní konfigurace síťových rozhraní (aktivace rozhraní, zadání IP adresy, síťové masky a několika dalších parametrů dle typu rozhraní) a základní konfigurace směrování (statické směrování, dynamické směrování – pouze protokol RIP – Routing Internet Protocol, [2] str. 296-304).



Obrázek 2.1: Cisco Packet Tracer, vzorová topologie

Obrázek 2.2 zobrazuje konfiguraci síťového rozhraní v Cisco Packet Traceru.



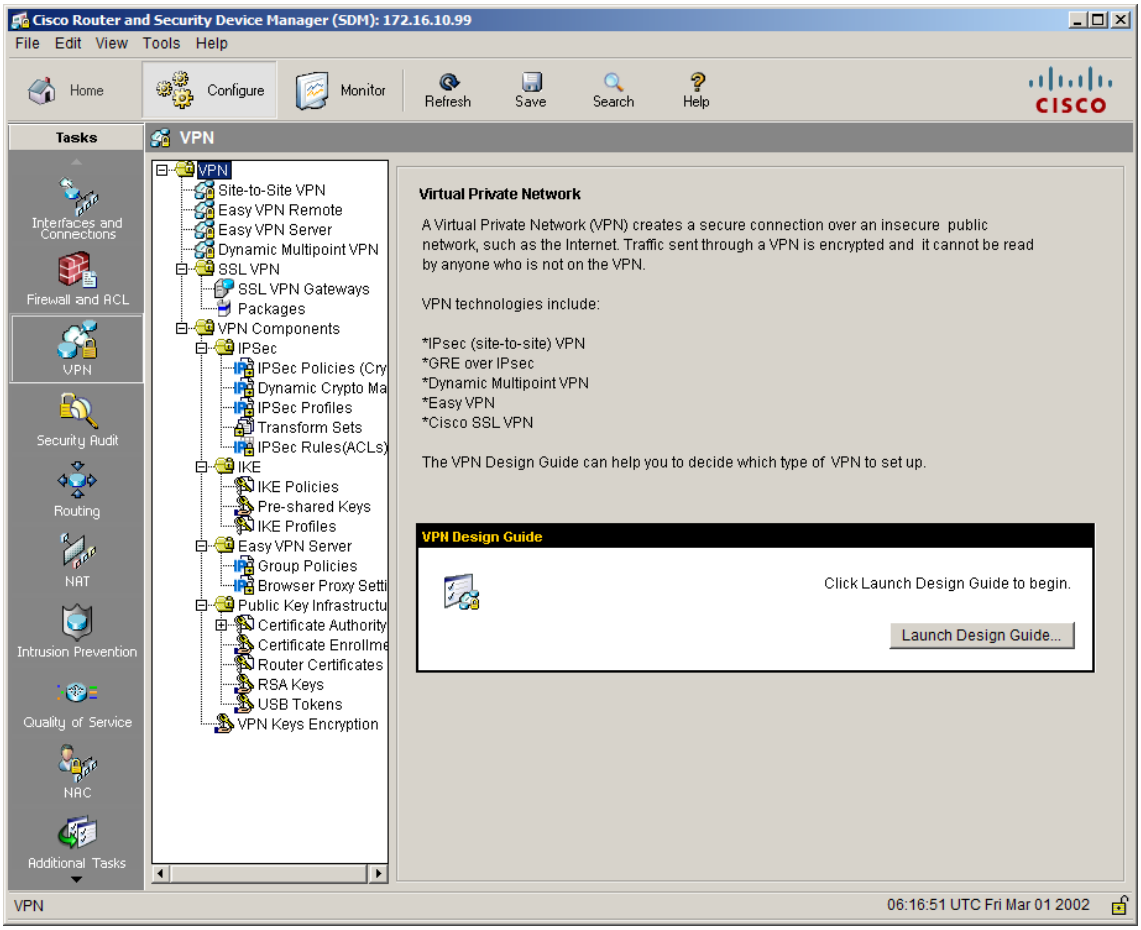
Obrázek 2.2: Cisco Packet Tracer, konfigurace síťového rozhraní

Z příkazové řádky (která se chová podobně jako u fyzických zařízení) lze nastavit další parametry a protokoly. Z pohledu bezpečnosti funguje nastavení většiny parametrů (zabezpečení privilegovaného módu, zabezpečení přístupových kanálů, uživatelé a oprávnění, služby). Je možné konfigurovat i přístupové listy (ACL, Access List, [3] str. 270-277). Z interních směrovacích protokolů (IGP, Interior Gateway Protocols – směrovací protokoly určené pro směrování uvnitř autonomního systému, [4] str. 279) je možné nakonfigurovat směrovací protokol OSPF (Open Shortest Path First, [4] str. 284-287) a EIGRP (Enhanced Interior Gateway Routing Protocol, [4] str. 283). Z rodiny externích směrovacích protokolů (EGP, Exterior Gateway Protocols – protokoly určené pro směrování mezi autonomními systémy, [4] str. 289) nechybí jeden z nejdůležitějších protokolů dnešního internetu – BGP (Border Gateway Protocol, [2] str. 269-290). Autonomním systémům se podrobně věnuje kapitola 4.2. V Cisco Packet Traceru chybí možnost vytvoření směrovacích map (`route-map`, [5]), složitější nastavení směrovacích protokolů tak není možné (např. lokální preference a metrika pro různé sítě u BGP).

2.2 Cisco Router and Security Device Manager

Cisco Router and Security Device Manager (SDM) je konfigurační nástroj, volně dostupný ke stažení po registraci na webu výrobce [6]. Cisco SDM ve verzi 2.5 je k dispozici jen pro platformu Microsoft Windows. Po spuštění nástroje se otevře konfigurační prostředí ve výchozím internetovém prohlížeči. Ke svému běhu potřebuje Javu verze 1.6 a vyšší. Nevýhodou je nutnost mít k dispozici podporované zařízení – nelze vygenerovat konfiguraci samostatně. Podporovaná

zařízení jsou např. směrovače Cisco řad 8xx, 18xx, 38xx a další, viz [6]. Konfigurační rozhraní nástroje přibližuje obrázek 2.3.



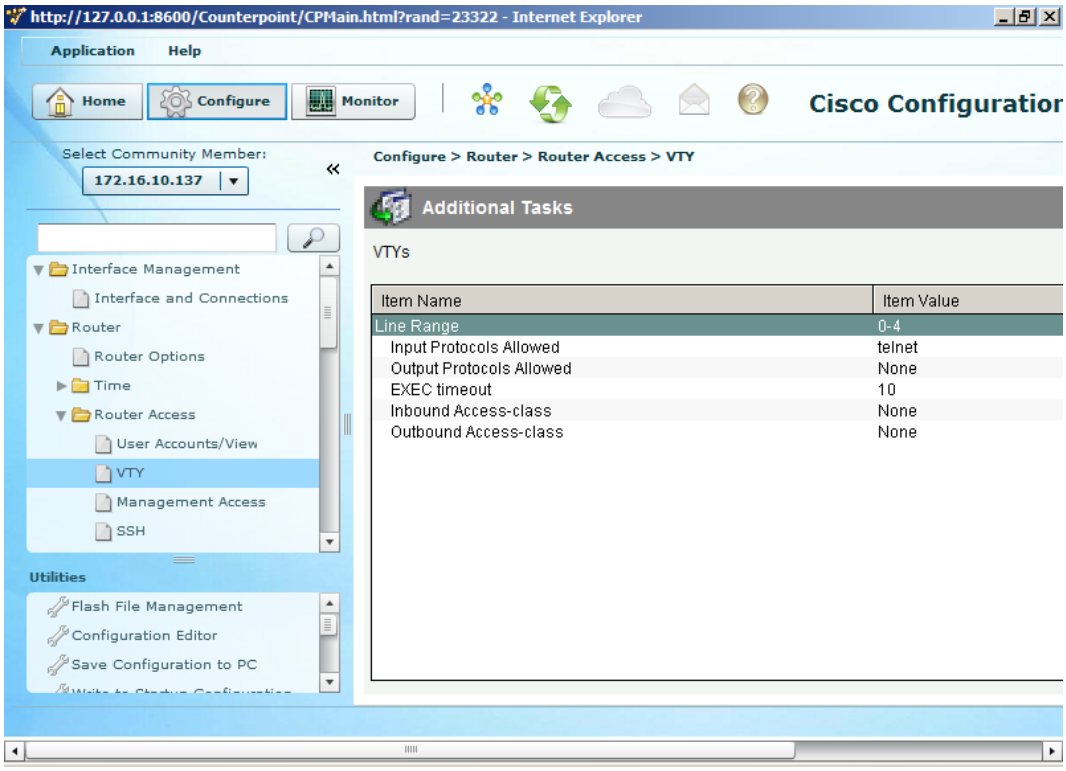
Obrázek 2.3: Cisco SDM, konfigurační rozhraní

Možnosti konfigurace jsou širší než u Packet Traceru, obsáhlá je zejména oblast pokročilého zabezpečení. Nechybí nastavení síťových rozhraní a směrovacích protokolů (statické směrování a dynamické směrování – protokoly RIP, OSPF a EIGRP). Směrovací protokol BGP bohužel zcela chybí. V oblasti bezpečnosti se nachází nastavení firewallu a přístupových listů (ACL). Snadno zde lze nastavit i překlad adres – NAT (Network Address Translation, [3] str. 204-208). Velká pozornost je věnována nastavení různých druhů virtuálních privátních sítí (VPN, Virtual Private Network, [3] str. 292-302) a tunelů. Nastavení

zabezpečení je možné konfigurovat pomocí průvodce (Security Audit) – nástroj projde zařízení a navrhne úpravy konfigurace.

2.3 Cisco Configuration Professional

Cisco Configuration Professional (CCP) je nástupcem nástroje Cisco SDM. Stejně jako jeho předchůdce je po registraci volně dostupný ke stažení na webu výrobce [7]. Nástroj Cisco CCP ve verzi 2.8 je dostupný jen pro platformu Microsoft Windows. Po spuštění nástroje se otevře konfigurační prostředí v internetovém prohlížeči. Ke svému běhu taktéž potřebuje Javu 1.6 a vyšší. Proti nástroji Cisco SDM má CCP zjednodušené a přehlednější grafické uživatelské rozhraní (viz obrázek 2.4).

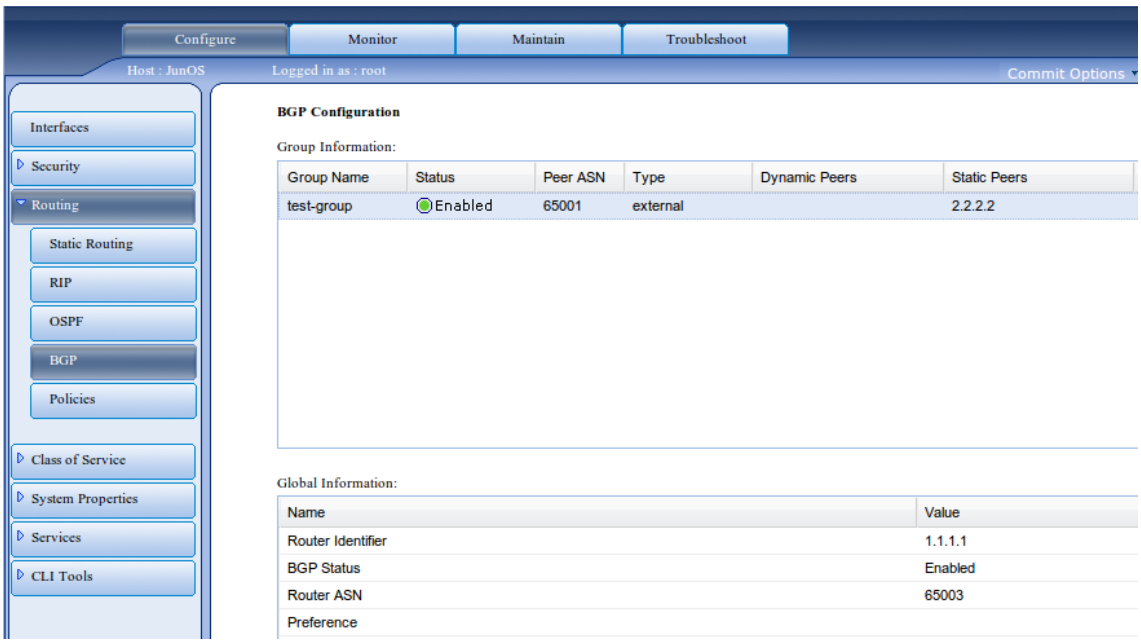


Obrázek 2.4: Cisco CCP, konfigurační rozhraní

Nástroj Cisco CCP přidává možnost nastavení dalších služeb, jako je DHCP (Dynamic Host Configuration Protocol, [2] str. 443-458), DNS (Domain Name System, [2] str. 461-481) či NTP (Network Time Protocol, [8]). Nastavení zabezpečení doznalo změn, je možné přehledně nastavit heslo do privilegovaného režimu, uživatelské účty, přístup pomocí telnetu ([2] str. 486-494) nebo SSH (Secure Shell, [9]).

2.4 Juniper J-Web

Pro srovnání byl vybrán i konfigurační nástroj J-Web společnosti Juniper [10], konkurenta společnosti Cisco na poli výroby síťových zařízení. Tento nástroj se liší od Cisco nástrojů zejména tím, že je instalován přímo v zařízení a přistupuje se do něj přes internetový prohlížeč (viz obrázek 2.5). Není nutné instalovat žádné podpůrné nástroje.



Obrázek 2.5: Juniper J-Web, detail konfiguračního rozhraní

Stejně jako u Cisco SDM či CCP je nutné mít podporované zařízení. Podporovaná zařízení jsou směrovače Juniper řady M a T či přepínače Juniper řady EX, viz [10]. V nástroji tohoto výrobce nechybí žádná z podstatných částí. Umožňuje podrobné nastavení zabezpečení i všech nejpoužívanějších směrovacích protokolů (RIP, OSPF i BGP). Chybí však konfigurátor VPN.

2.5 Hodnocení softwaru

Cisco Packet Tracer je skvělou volbou pro začínající studenty, díky vestavěnému simulačnímu módu je možné vidět i struktury přenášených paketů. V grafickém uživatelském rozhraní bohužel podporuje konfiguraci pouze základních parametrů a protokolů. Cisco SDM/CCP jsou nástroje, ve kterých chybí podrobnější konfigurace směrovacích protokolů. Cisco SDM neumožňuje jednoduše nastavit základní konfiguraci zabezpečení, tento nedostatek částečně napravuje novější Cisco CCP. Nastavit směrovací protokol BGP v grafickém uživatelském rozhraní dokáže pouze konfigurační nástroj Juniperu J-Web. Všechny představené nástroje jsou dostupné pouze pro omezenou část uživatelů. Cisco Packet Tracer jen pro studenty Cisco Networking Academy, Cisco SDM/CCP/Juniper J-Web pro vlastníky podporovaných zařízení. Tabulka 2.1 na základě vlastního průzkumu předkládá přehled nejdůležitějších vlastností, které jednotlivé nástroje podporují.

Tabulka 2.1: Přehled vlastností nástrojů pro konfiguraci síťových prvků

Název nástroje	**Zabezpečení: základní** Uživatelé, telnet, SSH	**Zabezpečení: pokročilé** Firewall, ACL	**Směrování: základní** Statické, RIP, OSPF	**Směrování: pokročilé** Další parametry	**BGP: základní** Sousedi, sítě	**BGP: pokročilé** Další parametry	**Konfigurace VPN**
Cisco Packet Tracer			✔				
Cisco SDM		✔	✔				✔
Cisco CCP	✔	✔	✔				✔
Juniper J-Web	✔	✔	✔	✔	✔	✔	

Ačkoliv je BGP jedním z nejdůležitějších protokolů dnešního internetu (zajišťuje směrování mezi autonomními systémy - sítěmi poskytovatelů, a tedy dostupnost všech sítí), z tabulky 2.1 vyplývá, že jen jeden nástroj je vhodný pro jeho konfiguraci v grafickém uživatelském rozhraní – J-Web od výrobce Juniper. Žádný software společnosti Cisco nepodporuje nastavení protokolu BGP v grafickém uživatelském rozhraní - předpokládají se pokročilé znalosti a konfigurace z příkazové řádky. Jejich použití pro výuku je proto omezeno. Konfigurace zabezpečení není ideální v žádném z uvedených nástrojů. Žádný z představených nástrojů nedisponuje možností rozšíření.

Zejména z důvodů uvedených v předchozím odstavci bude návrh vlastní aplikace zaměřen na konfiguraci zařízení Cisco, nastavení zabezpečení a nastavení protokolu BGP.

3 Úvod do konfigurace síťových zařízení Cisco

Operační systém síťových zařízení Cisco se nazývá Cisco IOS (Internetwork Operating System) [11]. Používají jej jak směrovače, tak přepínače tohoto výrobce. Profesionální administrátoři používají ke správě Cisco IOS příkazovou řádku – CLI (Command Line Interface) [12]. CLI umožňuje nastavit všechny vlastnosti zařízení.

3.1 Připojení k CLI

Připojení k CLI síťového zařízení je možné realizovat přes port fyzické konzole, port AUX nebo přes virtuální konzoli. Port fyzické konzole a port AUX jsou speciální porty umístěné přímo na zařízení.

Připojení k portu fyzické konzole se realizuje propojením tohoto portu na zařízení se sériovým portem klientského počítače. K propojení se používá speciální kabel. K CLI zařízení se přistupuje pomocí terminálového programu (např. PuTTY).

Port AUX se používá pro přístup k CLI zařízení pomocí modemu připojeného k telefonní lince. Je jednou z možností jak vzdáleně spravovat zařízení, pokud není možné připojení k virtuální konzoli např. z důvodu rozbitého směrování. Tato technologie je považována za překonanou a není proto zahrnuta v konfiguraci. [13]

Vzdálené připojení pomocí virtuální konzole se realizuje např. službou telnet či SSH. Pro prvotní nastavení zařízení se musí použít fyzická konzole, protože

připojení k virtuální konzoli je dostupné až po zprovoznění síťové komunikace a aktivaci telnetu či SSH (viz kapitola 3.4.6).

Po připojení k CLI je možné začít s konfigurací. Pro zpřehlednění jsou všechny příkazy zadávané uživatelem v následujících kapitolách **zvýrazněny červeně**.

3.2 Módy příkazové řádky

Uživatelské rozhraní příkazové řádky má několik různých módů [12]. Ihned po připojení se uživatel dostane do uživatelského módu, který je značně funkčně omezený. Příkazem `enable` se aktivuje privilegovaný mód:

```
Router>enable
Router#
```

V privilegovaném módu si uživatel může vypsat informace o aktuálním stavu zařízení. Pro přístup do globálního konfiguračního módu slouží příkaz `configure terminal`:

```
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
```

Z globálního konfiguračního módu se provádí konfigurace zařízení a vstupuje se do konfiguračního módu příslušných sekcí, viz další kapitoly.

3.3 Práce s konfiguračními soubory

Zařízení Cisco obsahují dva základní konfigurační soubory – `running-config` (aktuálně běžící konfigurace zařízení) a `startup-config` (konfigurace načtená po zapnutí zařízení z paměti NVRAM – Non-Volatile Random Access Memory).

Uložení běžící konfigurace do paměti NVRAM, odkud se konfigurace načítá po spuštění zařízení, se provádí příkazem `copy`.

```
R0#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

Konfigurační soubory je možné kopírovat (zálohovat) i do jiných destinací, více informací v [14].

3.4 Zabezpečení

V kapitole zabezpečení jsou probrány možnosti zabezpečení zařízení Cisco, od nastavení jména zařízení, šifrování hesel, uživatelských účtů, přes zabezpečení přístupových kanálů (fyzická konzole a virtuální konzole) po konfiguraci služeb běžících na směrovačích. Informace čerpány ze zdrojů [15], [16], [17] a [18].

3.4.1 Nastavení jména zařízení

Jako první by měl administrátor nastavit jméno zařízení (hostname). Nastavení se provede v globálním konfiguračním módu příkazem:

```
Router(config)#hostname IOS
```

3.4.2 Šifrování hesel

Zařízení Cisco v základním nastavení nešifrují některá hesla (v konfiguračním módu zadaná jako `password`). Ta jsou poté v konfiguraci uložena v čitelné formě. Jedná se například o heslo do privilegovaného módu (`enable password`), hesla k přístupovým kanálům (`line console`, `line vty`) či hesla používaná k šifrování směrovacích protokolů.

Bezpečnější možností, jak tato hesla šifrovat, je zadávat je jako `secret` (příkladem opět `enable secret`). Nelze je ale takto zadávat u přístupových kanálů – `line console`, `line vty` a také u směrovacích protokolů. Doporučeno je zapnout službu šifrování hesel globálně příkazem:

```
IOS(config)#service password-encryption
```

Po aktivaci služby jsou všechna hesla zašifrována – i ta, která byla zadána před spuštěním šifrování. Po případném vypnutí služby zůstanou již zašifrovaná hesla v nečitelné formě, nově zadaná budou znovu v konfiguraci čitelná.

Šifrování pomocí `service password-encryption` je bohužel slabé a bylo již prolomeno – existuje software na rychlé dešifrování. Toto šifrování znemožňuje pouze rychlé odpozorování hesla, např. při výpisu konfigurace na obrazovku. Pokud útočník získá zašifrovaný řetězec s heslem, jeho dešifrování je otázkou několika vteřin. Společnost Cisco doporučuje využít pro ověřování ověřovací servery, např. RADIUS (Remote Authentication Dial In User Service) nebo TACACS+ (Terminal Access Controller Access-Control System). [19]

3.4.3 Heslo do privilegovaného módu

Dalším krokem je nastavení hesla do privilegovaného módu. Viz kapitola 3.4.2, je lepší použít šifrované `secret`. Nastavení hesla `cisco`:

```
IOS(config)#enable secret cisco
```

3.4.4 Uživatelské účty

Správa uživatelských účtů se provádí příkazem `username`. Stejně jako u hesla do privilegovaného módu je v příkladu použito `secret`. Lze nastavit i úroveň oprávnění od 0 do 15, kde 15 je nejvyšší úroveň oprávnění. Přidání uživatelského účtu `cisco` s nejvyšší úrovní oprávnění a heslem `cisco`:

```
IOS(config)#username cisco privilege 15 secret cisco
```

Použití uživatelského účtu pro ověření je třeba nakonfigurovat u jednotlivých přístupových kanálů – u konzole, virtuální konzole či webového serveru, viz další kapitoly.

3.4.5 Zabezpečení fyzické konzole

Zabezpečení přístupových kanálů začíná u fyzické konzole (`line console`). Všechny parametry se zadávají z konfiguračního módu `line`.

```
IOS(config)#line console 0
```

Je možné nastavit přístupové heslo (`password`), vždy je nutné zapnout jeho vyžadování direktivou `login`.

```
IOS(config-line)#password cisco
IOS(config-line)#login
```

Nastavení přihlášení pomocí lokálního uživatelského jména a hesla, nastaveného v kapitole 3.4.4:

```
IOS(config-line)#login local
```

Užitečným parametrem je `logging synchronous` – zabraňuje zařízení, aby přerušilo výstupem (např. ladící zprávou) na konzoli uživatelem zapisovaný příkaz.

```
IOS(config-line)#logging synchronous
```

3.4.6 Zabezpečení virtuální konzole

Základní zabezpečení virtuální konzole (`line vty`) se neliší od zabezpečení fyzické konzole. Mimo vyžadování hesla je možné také nastavit služby, kterými se lze na virtuální konzoli připojit – jedná se zejména o nešifrovaný telnet a šifrované SSH. Všechny parametry se zadávají opět z konfiguračního módu `line`.

```
IOS(config)#line vty 0 4
```

Nastavení hesla a jeho vyžadování:

```
IOS(config-line)#password cisco
IOS(config-line)#login
```

Nastavení přihlášení pomocí lokálního uživatelského jména a hesla, nastaveného v kapitole 3.4.4:

```
IOS(config-line)#login local
```

Při přístupu přes telnet lze použít obě varianty ověření uživatele – jen heslem nebo uživatelským účtem. Aktivace služby telnet:

```
IOS(config-line)#transport input telnet
```

Pro přístup přes SSH je nutné nejprve nastavit doménové jméno, vygenerovat šifrovací klíč a nakonec nastavit verzi SSH protokolu. Až poté je možné

aktivovat přístup do zařízení přes SSH. Připojení přes SSH funguje pouze s nastaveným uživatelským jménem a heslem.

```
IOS(config)#ip domain-name domain.local
IOS(config)#crypto key generate rsa modulus 1024
The name for the keys will be: IOS.domain.local

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]
IOS(config)#
*Jun 10 20:03:31.065: %SSH-5-ENABLED: SSH 1.99 has been enabled
IOS(config)#ip ssh version 2
IOS(config)#line vty 0 4
IOS(config-line)#login local
IOS(config-line)#transport input ssh
```

Aktivace přístupu přes telnet i SSH:

```
IOS(config-line)#login local
IOS(config-line)#transport input telnet ssh
```

Také u virtuální konzole lze nastavit užitečný parametr `logging synchronous` – zabraňuje zařízení, aby přerušilo výstupem (např. ladící zprávou) na konzoli uživatelem zapisovaný příkaz.

```
IOS(config-line)#logging synchronous
```

3.4.7 Služby

Na směrovačích běží v základním nastavení služby, které mohou být zneužity a pro běžnou činnost nejsou potřebné. Některé z nich je vhodné před nasazením do provozu vypnout.

Zakázání „malých“ serverů

Mezi „malé“ servery se řadí služby echo, chargen, discard a daytime [20].

```
IOS(config)#no service tcp-small-servers
IOS(config)#no service udp-small-servers
```

Zakázání DNS překladu

Pokud není nastaven správný jmenný server, je vhodné z důvodu časové úspory vypnout službu překladu doménových jmen. Čas je uspořen díky tomu, že se nevysílá dotaz pokaždé, když dojde k překlepu v názvu příkazu.

```
IOS(config)#no ip domain-lookup
```

HTTP/HTTPs servery

Pokud nejsou využívány webové servery, je vhodné je vypnout.

```
IOS(config)#no ip http server
IOS(config)#no ip http secure-server
```

Vypnutí HTTP serverů má za následek nemožnost konfigurovat zařízení pomocí nástrojů Cisco SDM a Cisco CCP. Pokud je nutné tyto nástroje využívat, musí se povolit alespoň jeden server, a nastavit lokální ověřování. Doporučena je HTTPs varianta.

```
IOS(config)#ip http secure-server
IOS(config)#ip http authentication local
```

Zakázání stahování konfigurace ze sítě

V základním nastavení se zařízení pokouší stáhnout konfiguraci z TFTP (Trivial File Transfer Protocol, [3] str. 85-88) serveru. Pokud tato služba není využívána, je doporučeno službu vypnout.

```
IOS(config)#no service config
```

Vypnutí protokolu CDP

Protokol CDP (Cisco Discovery Protocol, [21]) umožňuje sousedům zjistit informace o zařízení a jeho konfiguraci - např. jméno síťového rozhraní, typ zařízení či verzi operačního systému. Pokud není tento protokol potřeba, doporučuje se ho vypnout.

```
IOS(config)#no cdp run
```

3.5 Síťová rozhraní

Základní parametry síťového rozhraní, jako je IP adresa, aktivace rozhraní nebo popis, se nastaví v konfiguračním módu daného rozhraní.

```
IOS(config)#interface Ethernet 0/0
IOS(config-if)#ip address 192.168.1.1 255.255.255.0
IOS(config-if)#no shutdown
IOS(config-if)#description IOS->R2
```

U sériových rozhraní má smysl nastavovat i další parametry, jako například `bandwidth` (šířka pásma, kb/s) či `clock rate` (nastavení rychlosti přenosu, b/s). Parametr `bandwidth` se používá např. u směrovacího protokolu OSPF pro výpočet metriky. Neovlivňuje skutečnou rychlost linky. Naopak parametr `clock rate` určuje hodinový takt - skutečnou rychlost linky.

```
IOS(config)#interface Serial 1/0
IOS(config-if)#ip address 192.168.2.1 255.255.255.0
IOS(config-if)#no shutdown
IOS(config-if)#description IOS->R3
IOS(config-if)#clock rate 64000
IOS(config-if)#bandwidth 64
```

3.6 Statické směrování

Statické směrování označuje administrátorem nastavené cesty k cílovým sítím. Statické směrování se používá v malých sítích, kde je předem známa cesta do cíle. V případě výpadku neexistuje náhradní trasa. ([4], str. 273)

Nastavení statického směrování se provádí příkazem `ip route`. Prvním parametrem je cílová síť s maskou sítě, dalším parametrem je následující skok – tím může být IP adresa, síťové rozhraní nebo i rozhraní Null 0 (používá se např. když je třeba mít ve směrovací tabulce sumární adresu).

```
IOS(config)#ip route 10.10.10.0 255.255.255.0 2.2.2.2
IOS(config)#ip route 10.10.20.0 255.255.255.0 Ethernet 0/0
IOS(config)#ip route 10.10.0.0 255.255.0.0 Null 0
```

3.7 Dynamické směrování

Dynamickým směrováním se označuje výběr nejlepší trasy do cílové sítě na základě parametrů linek. Tyto parametry se vyměňují pomocí zpráv mezi směrovači. Směrovací protokoly se rozdělují na IGP (interní směrovací protokoly – uvnitř autonomního systému) a EGP (externí směrovací protokoly – mezi autonomními systémy). ([4], str. 274)

3.7.1 Interní směrovací protokoly - RIP

Směrovací protokol RIP je tzv. distance-vector směrovací protokol – směrovače si posílají vektor vzdáleností k jednotlivým sítím. Metrikou je počet směrovačů na cestě k cílové síti. Nejvyšší metrikou je 15 směrovačů, 16 je již neplatná cesta. Výpočet nejlepší cesty je prováděn Bellman-Fordovým algoritmem. Nevýhodou je pomalá konvergence sítě. Existují dvě verze protokolu pro IPv4 sítě - RIPv1 pracuje pouze s classful sítěmi (sítě bez masky podsítě rozdělené do tříd, [2] str. 64), RIPv2 umí pracovat i s classless sítěmi (sítě s maskou podsítě, [2] str. 67). ([4] str. 279-282)

Základní konfigurace protokolu RIP

Proces směrovacího protokolu RIP se aktivuje příkazem `router rip`.

```
IOS(config)#router rip
IOS(config-router)#
```

Další konfigurace probíhá v konfiguračním módu směrovacího protokolu.

Verze protokolu se nastaví příkazem `version`.

```
IOS(config-router)#version 2
```

Vhodné je deaktivovat automatickou sumarizaci adres sítí, aby směrovač neposílal pouze classful sítě.

```
IOS(config-router)#no auto-summary
```

Propagování sítě RIP sousedům zajistí příkaz `network`. Ačkoliv RIPv2 podporuje classless sítě, zadává se adresa sítě bez masky. Propagace sítě `10.10.10.0`:

```
IOS(config-router)#network 10.10.10.0
```

RIP soused se přidá příkazem `neighbor`.

```
IOS(config-router)#neighbor 10.10.10.1
```

3.7.2 Interní směrovací protokoly - OSPF

Směrovací protokol OSPF je tzv. link-state směrovací protokol – směrovače si mezi sebou posílají zprávy o stavu linek. Tyto zprávy jsou zasílány ihned poté, co změna nastane. Metrika je odvozena od propustnosti síťového rozhraní – čím větší propustnost, tím nižší „cena“ cesty. Výpočet nejlepší cesty je prováděn Dijkstrovým algoritmem. Výhodou proti protokolu RIP je rychlá konvergence sítě, nevýhodou vyšší výpočetní náročnost. Každý OSPF směrovač si uchovává celou topologii sítě, topologii je možné dělit do tzv. areas – oblastí. Pro každou oblast si směrovač uchovává topologii zvlášť. ([4], str. 284-287)

Základní konfigurace protokolu OSPF

Proces směrovacího protokolu OSPF se aktivuje příkazem `router ospf id`, kde `id` je identifikátor procesu. Na směrovači může běžet více OSPF procesů.

```
IOS(config)#router ospf 1
IOS(config-router)#
```

Další konfigurace probíhá v konfiguračním módu směrovacího protokolu.

Propagování sítě OSPF sousedům zajistí příkaz `network`. Adresa sítě se zadává s tzv. wildcard maskou podsítě – jedná se o převrácenou masku podsítě. Je třeba nastavit oblast parametrem `area`.

```
IOS(config-router)#network 10.10.10.0 0.0.0.255 area 0
```

OSPF soused se přidá příkazem `neighbor`.

```
IOS(config-router)#neighbor 10.10.10.1
```

3.7.3 Externí směrovací protokoly - BGP

Představení a konfigurace směrovacího protokolu BGP jsou velmi rozsáhlé, proto je jim věnována celá kapitola 4.

4 Směrovací protokol BGP

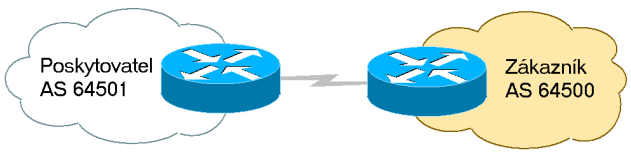
Směrovací protokol BGP patří mezi nejdůležitější protokoly internetu – je jeho „stavebním kamenem“. Patří do rodiny Exterior Gateway Protocols (EGP) a používá se ke směrování mezi autonomními systémy – zajišťuje propojení sítí různých poskytovatelů. Informace čerpány z [2], [3], [22] a [23].

4.1 Základní informace

Směrovací protokol BGP je tzv. path-vector směrovacím protokolem – směrovače si posílají vektor cesty k cílovým sítím. Vektorem cesty je posloupnost čísel autonomních systémů, kterými vede cesta k cílové síti. Protokol komunikuje na TCP portu 179. Nejpoužívanější verzí protokolu je jeho nejnovější verze 4.

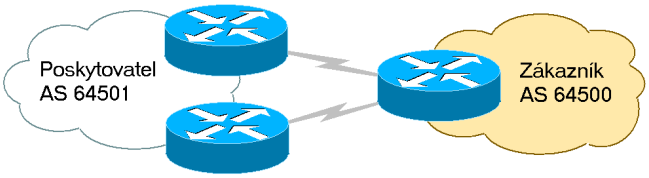
4.2 Autonomní systém (AS)

Pod pojmem autonomní systém si lze představit síť s jednotnou směrovací politikou, spravovanou jednou skupinou. Rozlišuje se několik typů autonomních systémů, podle způsobu připojení. Single-homed AS (stub, obrázek 4.1) je připojen pouze jednou linkou k jednomu poskytovateli/AS.



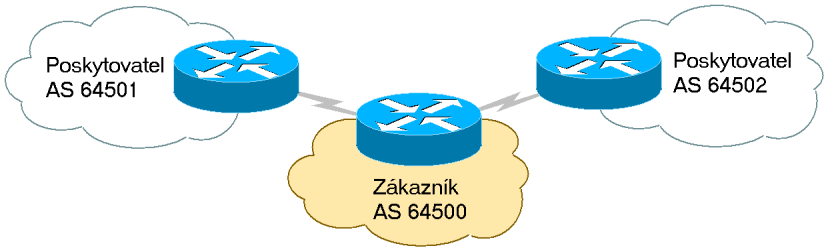
Obrázek 4.1: BGP, single-homed (stub) AS

Dual-homed AS (obrázek 4.2) je také připojen pouze k jednomu poskytovateli/AS, má však více než jednu linku.



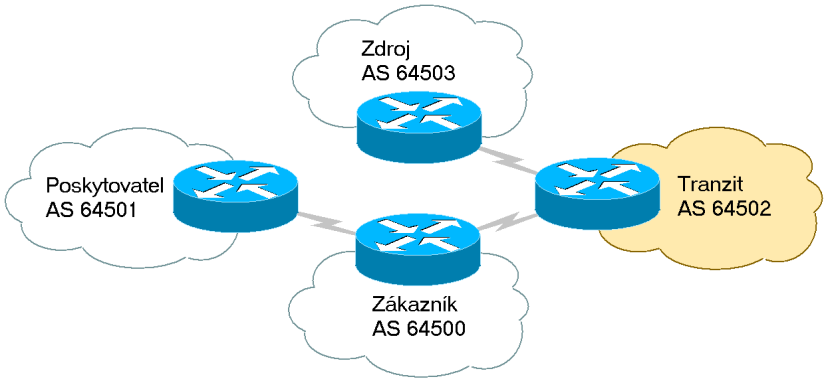
Obrázek 4.2: BGP, dual-homed AS

Jako multi-homed je označován AS, který je připojený k více než jednumu poskytovateli/AS (obrázek 4.3). Takový AS nepřipouští tranzit provozu, zákazník propaguje oběma poskytovatelům pouze své vlastní sítě.



Obrázek 4.3: BGP, multi-homed AS

Za tranzitní se považuje AS, který zprostředkovává spojení mezi dalšími AS – přenáší data jiných sítí (viz obrázek 4.4). Není tedy zdrojovým ani cílovým AS.



Obrázek 4.4: BGP, tranzitní AS

4.3 Číslo autonomního systému (ASN)

Autonomní systém se označuje číslem – Autonomous System Number (ASN). Může být buď 16bitové (ASN16), nebo 32bitové (ASN32). V případě, že zařízení neumí použít ASN32, používá se přechodové ASN16 23456. Přehled rezervovaných ASN čísel je v tabulce 4.1 (zdroj: [24]).

Tabulka 4.1: BGP - rezervovaná ASN čísla

ASN16	
0	Rezervováno
23456	Rezervováno – přechodový mechanismus
64000 - 64495	Rezervováno – organizace IANA
64496 - 64511	Rezervováno – použití v dokumentaci a příkladech
64512 - 65534	Rezervováno – privátní AS
65535	Rezervováno
ASN32	*Nižší hodnoty stejné jako ASN16*
65536 - 65551	Rezervováno – použití v dokumentaci a příkladech
65552 - 131071	Rezervováno
4200000000 - 4294967294	Rezervováno – privátní AS
4294967295	Rezervováno

Čísla ASN se mohou zapisovat třemi různými způsoby, viz tabulka 4.2.

Tabulka 4.2: BGP - zápis ASN čísla

Typ zápisu	**Příklad ASN16**	**Příklad ASN32**
ASPLAIN (Cisco výchozí)	64500 → 64500	65550 → 65550
ASDOT	64500 → 64500	65550 → 1.14
ASDOT+	64500 → 0.64500	65550 → 1.14

Výpočet ASDOT nebo ASDOT+ formátu je jednoduchý – ASN číslo se vydělí číslem 65536, číslo před tečkou je podíl, za tečkou zbytek po dělení.

4.4 BGP zprávy

BGP směrovače si vyměňují čtyři typy zpráv – OPEN, UPDATE, KEEPALIVE a NOTIFICATION ([2], str. 278-289).

Zpráva OPEN

Zpráva OPEN se vyměňuje při navázání sousedství s jiným BGP směrovačem. Obsahuje informaci o verzi protokolu, čísle autonomního systému, hold time (čas, po kterém je BGP soused označen za nefunkčního) a BGP identifikátor (u směrovačů Cisco router-id - IP adresa nastavená na směrovači, s nejvyšší hodnotou). Dále může obsahovat volitelné parametry jako např. zabezpečení heslem.

Zpráva UPDATE

Zpráva UPDATE nese vlastní směrovací informace. V první části nese informace o cestách, které už nejsou platné a které má směrovač vymazat. V druhé části nese informace o nových cestách a jejich atributech (viz kapitola 4.6).

Zpráva KEEPALIVE

Zpráva KEEPALIVE je vyměňována za účelem ověření funkčnosti spojení mezi sousedy. BGP soused je považován za nefunkčního, nepřijde-li od něj zpráva po dobu domluvenou ve zprávě OPEN (hold time).

Zpráva NOTIFICATION

Pokud se vyskytne chyba v přijaté zprávě OPEN či UPDATE, je protistraně zaslána zpráva NOTIFICATION s kódem chyby a je ukončeno spojení. Zpráva se také zasílá, pokud není obdržena zpráva KEEPALIVE v daném limitu.

4.5 Interní a externí BGP

BGP sousedství se dělí na dva typy – interní a externí. O interním BGP (iBGP) se hovoří, když je navázáno sousedství dvou směrovačů v rámci jednoho AS. Externím BGP (eBGP) se označuje sousedství dvou směrovačů v různých AS.

U externího BGP se zamezuje směrovacím smyčkám (routing loops) tak, že se zahodí cesta, která již obsahuje číslo lokálního AS. Tato podmínka se však nemůže uplatnit u interního BGP – byly by zahozeny všechny cesty. Předávání směrovací informace v rámci jednoho AS se proto řídí dalšími podmínkami:

1. Informace získané z iBGP se šíří eBGP sousedům, ale nešíří se iBGP sousedům.
2. Informace získané z eBGP se šíří všem – iBGP i eBGP sousedům.

Pokud je v jednom AS několik BGP směrovačů, je třeba, aby byl každý s každým sousedem – z důvodu zajištění konzistence směrovací informace v rámci AS (důvodem je podmínka 1). Není nutné, aby směrovače byly přímými sousedy, stačí když jsou dostupné přes interní směrovací protokol. [22]

4.6 BGP atributy

Protokol BGP přenáší ve svých zprávách mnoho atributů. Atributy se rozdělují do 4 kategorií (viz tabulka 4.3). [23]

Tabulka 4.3: BGP - kategorie atributů

Kategorie	**Implementace**
Well-known, mandatory	Atribut musí být implementován (směrovač ho musí rozpoznat) a musí být součástí každé cesty v UPDATE zprávě.
Well-known, discretionary	Atribut musí být implementován (směrovač ho musí rozpoznat), ale nemusí být součástí každé cesty v UPDATE zprávě.
Optional, transitive	Atribut nemusí být implementován (směrovač ho nemusí rozpoznat), ale musí ho vždy v nezměněné formě přeposlat.
Optional, non-transitive	Atribut nemusí být implementován (směrovač ho nemusí rozpoznat). Pokud směrovač atribut nerozpozná, nepřeposílá ho dále.

ORIGIN (Well-known, mandatory)

Atribut ORIGIN označuje zdroj cesty – IGP (i/I na konci AS_PATH), EGP (e/E na konci AS_PATH) nebo INCOMPLETE (? na konci AS_PATH).

AS_PATH (Well-known, mandatory)

AS_PATH označuje vektor AS, přes které je cesta propagována. V případě, že je AS připojen k více než jednomu poskytovateli, lze ovlivněním tohoto atributu znevýhodnit konkrétní trasu. Toto znevýhodnění se nazývá AS_PATH prepending – jednomu z poskytovatelů se propagují cesty s uměle prodlouženou AS_PATH (lokální ASN se jednou nebo několikrát vloží do AS_PATH).

NEXT_HOP (Well-known, mandatory)

NEXT_HOP definuje IP adresu dalšího směrovače, přes který je cesta dostupná.

MULTI_EXIT_DISC (Optional, non-transitive)

Multiple Exit Discriminator, též MED. Pokud existuje více tras do sousedního AS, pak tento atribut umožňuje nastavit jejich preferenci (říká sousednímu AS, kterou z tras má použít).

LOCAL_PREF (Well-known, discretionary)

Lokální preference se předává pouze ve vlastním AS pomocí iBGP. Určuje, která trasa je upřednostňována pro odchozí provoz.

ATOMIC_AGGREGATE (Well-known, discretionary)

Atribut ukazující, že došlo k agregaci (sumarizaci) cest.

AGGREGATOR (Optional, transitive)

Identifikace AS a konkrétního směrovače, na kterém k agregaci (sumarizaci) cest došlo.

COMMUNITY (Optional, transitive)

Atribut slouží k označení cest.

WEIGHT (Cisco proprietární)

Váha je Cisco proprietární atribut, používá se pouze na lokálním směrovači k určení preferované odchozí trasy.

4.7 Volba nejlepší cesty

Algoritmus volby nejlepší cesty popisuje tabulka 4.4. [23]

Tabulka 4.4: BGP - algoritmus volby nejlepší cesty

	Atribut	
1	WEIGHT	Preferuje se vyšší hodnota (Cisco proprietární) Výchozí 0, lokální cesty 32768
2	**LOCAL_PREF**	Preferuje se vyšší hodnota Výchozí 100
3	*Lokální cesty*	Preferují se cesty zadané příkazem `network`, `aggregate` nebo redistribuce z IGP
4	**AS_PATH**	Preferuje se kratší cesta
5	**ORIGIN**	Preferuje se nižší z hodnot IGP (i) < EGP (e) < INCOMPLETE (?)
6	**MULTI_EXIT_DISC**	Preferuje se nižší hodnota Výchozí 0
7	*eBGP / iBGP*	Preferují se cesty získané přes eBGP nad cestami získanými přes iBGP
8	*eBGP / iBGP*	(iBGP) Preferuje se nižší IGP metrika (eBGP) Preferuje se dříve přijatý prefix
9	*BGP identifikátor*	Preferuje se nižší BGP identifikátor (router-id) souseda
10	*IP adresa souseda*	Preferuje se nižší IP adresa souseda

4.8 Konfigurace

Základní nastavení směrovacího protokolu BGP se provádí v globálním konfiguračním módu. Proces BGP s lokálním ASN 64500 se aktivuje příkazem:

```
IOS(config)#router bgp 64500
```

Další konfigurace probíhá v konfiguračním módu směrovacího protokolu.

Propagování sítě (prefixu) BGP sousedům zajistí příkaz `network`. Je nutné, aby prefix byl ve směrovací tabulce, jinak nebude směrovačem propagován. Propagace sítě `10.10.10.0/24`:

```
IOS(config-router)#network 10.10.10.0 mask 255.255.255.0
```

Nastavení propagace výchozí cesty:

```
IOS(config-router)#default-information originate
```

BGP soused se přidá příkazem `neighbor`, jako první se musí specifikovat vzdálené ASN.

```
IOS(config-router)#neighbor 10.10.10.1 remote-as 64501
```

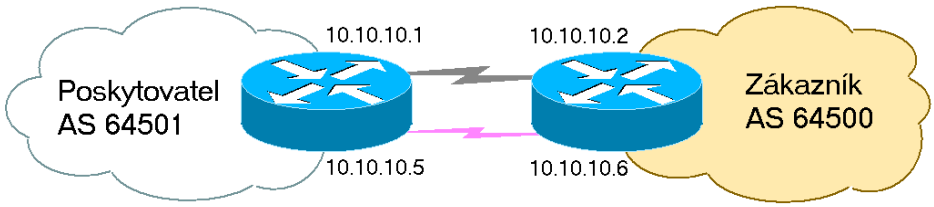
Až po nastavení vzdáleného ASN je možné konfigurovat další parametry (příkladem popis).

```
IOS(config-router)#neighbor 10.10.10.1 description IOS->ISP
```

BGP relaci je možné zabezpečit heslem (MD5). Heslo musí mít shodně nastaveno oba sousedi, jinak není relace navázána.

```
IOS(config-router)#neighbor 10.10.10.1 password heslo
```

Atributy, jako například MED či lokální preference, se mohou modifikovat aplikováním `route-map` na cesty získané od BGP souseda, a to buď v příchozím, nebo odchozím směru. Příkladem je modifikace dvou zmíněných atributů pro situaci na obrázku 4.5 – zákazník má dvě linky od stejného poskytovatele, ale z nějakého důvodu chce pro odchozí i příchozí preferovat právě jednu z nich. V `route-map` je možné specifikovat více omezujících podmínek, pro jednoduchost ale nebudou nastaveny.



Obrázek 4.5: BGP, dual-homed zákazník

Zákazník si přeje preferovat spodní trasu pro odchozí i příchozí provoz. Pro tuto situaci se vytvoří 4 `route-map` – pro horní trasu s vyšším MED a nižší lokální preferencí, pro spodní trasu s nižším MED a vyšší lokální preferencí.

```
IOS(config)#route-map TOP-MED
IOS(config-route-map)#set metric 100
IOS(config-route-map)#route-map TOP-LP
IOS(config-route-map)#set local-preference 0
IOS(config-route-map)#route-map BOTTOM-MED
IOS(config-route-map)#set metric 0
IOS(config-route-map)#route-map BOTTOM-LP
IOS(config-route-map)#set local-preference 1000
```

Následně se tyto `route-map` aplikují jednotlivým BGP sousedům – MED na všechny odchozí cesty, lokální preference na všechny příchozí cesty.

```
IOS(config-router)#neighbor 10.10.10.1 route-map TOP-MED out
IOS(config-router)#neighbor 10.10.10.1 route-map TOP-LP in
IOS(config-router)#neighbor 10.10.10.5 route-map BOTTOM-MED out
IOS(config-router)#neighbor 10.10.10.5 route-map BOTTOM-LP in
```

Díky této konfiguraci bude AS 64500 (zákazník) využívat pro odchozí provoz spodní trasu – díky nastavené vyšší hodnotě lokální preference. Poskytovatel bude pro provoz do AS 64501 využívat také spodní trasu – má nastaven parametr MED na nižší hodnotu.

5 Návrh aplikace

Při testování nástrojů v kapitole 2 bylo zjištěno několik nedostatků. Nástroje pro konfiguraci síťových zařízení výrobce Cisco většinou umožňují pouze základní konfiguraci směrovacích protokolů. Úplně chybí nastavení směrovacího protokolu BGP v grafickém uživatelském rozhraní. Nastavení zabezpečení nepokrývá všechny důležité oblasti. Žádná z prozkoumaných aplikací nemá možnost rozšíření, proto se návrh soustředí na vytvoření nové aplikace.

Výše zmíněné nedostatky budou hrát hlavní roli při návrhu aplikace. Aplikace bude umožňovat vytvoření virtuální topologie sítě a pokročilou konfiguraci síťových prvků značky Cisco v grafickém uživatelském rozhraní.

5.1 Očekávaný přínos aplikace

Aplikace by měla svým uživatelům přinést přívětivé nastavení síťových zařízení Cisco, s důrazem na zabezpečení a směrovací protokol BGP, bez nutnosti znát syntaxi příkazů. Může být využita ve výuce, neboť na rozdíl od zkoumaných nástrojů nemá žádná licenční omezení.

5.2 Uživatelé

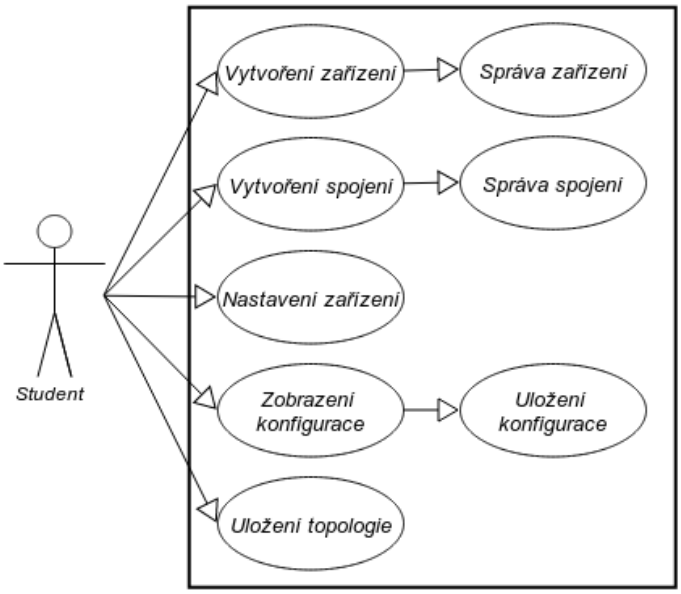
Uživateli aplikace budou zejména studenti, zajímající se o konfiguraci síťových prvků. Předpokládá se teoretická znalost nastavovaných zařízení či protokolů, naopak se neočekává přesná znalost syntaxe příkazů.

5.3 Klíčové vlastnosti

Aplikace musí umožnit vytvoření virtuální topologie – vložení zařízení a jeho propojení s ostatními. Musí poskytovat snadnou konfiguraci síťových prvků v grafickém uživatelském rozhraní – zejména nastavení síťových rozhraní, zabezpečení a směrování. Veškeré změny se okamžitě po provedení promítnou ve vygenerované konfiguraci. Konfiguraci je možné zkopírovat či uložit do souboru.

5.4 Diagram případů užití

Obrázek 5.1 zobrazuje diagram případů užití aplikace. Obsahuje všechny akce, které může uživatel (student) v aplikaci provést.



Obrázek 5.1: Diagram případů užití

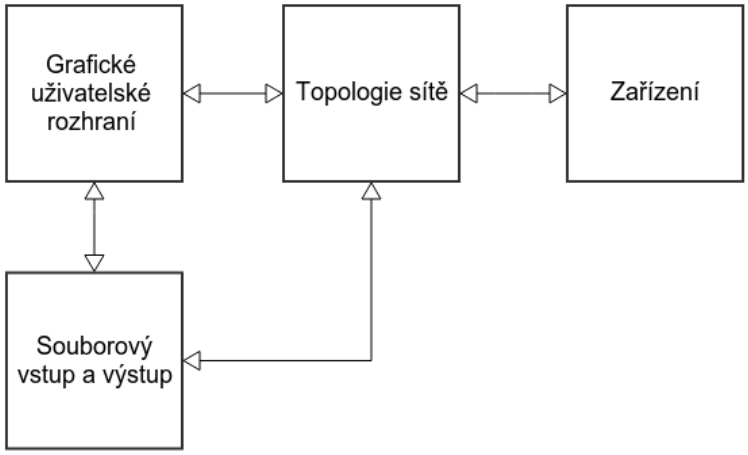
5.5 Funkční požadavky

Seznam níže popisuje funkční požadavky na aplikaci. Požadavky jsou seřazeny podle jejich priority, od nejvyšší po nejnižší.

- Zobrazení celé topologie
- Vytvoření síťového zařízení
- Vytvoření spojení mezi zařízeními
- Nastavení jména zařízení (hostname)
- Nastavení zabezpečení
- Nastavení síťových rozhraní
- Nastavení statického směrování
- Nastavení dynamického směrování – BGP
- Zobrazení aktuální konfigurace zařízení
- Uložení vygenerované konfigurace do souboru
- Smazání zařízení
- Smazání spojení
- Uložení celé topologie do souboru
- Nahrání celé topologie ze souboru
- Barevné odlišení spojení podle jeho typu

5.6 Logická struktura aplikace

Logickou strukturu aplikace ukazuje obrázek 5.2. Skládá se ze čtyř částí – grafické uživatelské rozhraní, topologie sítě, datové struktury zařízení a souborový vstup a výstup. Šipky naznačují komunikaci mezi moduly.



Obrázek 5.2: Logická struktura aplikace

5.6.1 Grafické uživatelské rozhraní

Grafické uživatelské rozhraní zajišťuje komunikaci s uživatelem. Zobrazuje topologii sítě, pomocí dialogů nastavuje parametry zařízení.

5.6.2 Topologie sítě

Celá topologie sítě zařízení je ukládána v abstraktní struktuře graf. Díky této struktuře lze snadno získat všechny přímé sousedy a provádět další grafové operace nad topologií.

5.6.3 Datové struktury zařízení

Data a parametry nastavovaných zařízení jsou uloženy v datových strukturách – základní nastavení, nastavení zabezpečení, nastavení síťových rozhraní, směrování a další. Datová struktura představující jedno zařízení je ukládána v grafu topologie.

5.6.4 Souborový vstup a výstup

Souborový vstup a výstup načte vstupní soubor a celou topologii uloží do grafu, nebo naopak z grafu celou topologii načte a uloží do souboru.

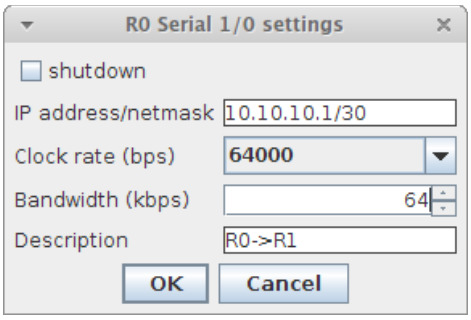
6 Výběr technologií

Aplikace by měla být snadno přenositelná na různé platformy (Microsoft Windows, Linux). Programovací jazyk by měl umožňovat autorovi snadno implementovat grafické uživatelské rozhraní.

Z výše uvedených důvodů a zejména z pohledu autorových zkušeností s programovacím jazykem byla zvolena Java. Grafické uživatelské rozhraní bude naprogramováno ve frameworku Swing, jenž je standardní součástí Javy. Bude využito i několik externích knihoven/frameworků, které jsou popsány v dalších kapitolách. Překlad a sestavení programu bude prováděno pomocí nástroje Apache Maven.

6.1 MiG Layout pro Swing

Ačkoliv framework Swing obsahuje několik vestavěných layout managerů, jejich použití je většinou buď těžkopádné (BorderLayout), nebo příliš složité (GridBagLayout). Z tohoto důvodu bude použit open-source layout manager MiG Layout, dostupný v repozitářích nástroje Maven. Tento layout manager se od vestavěných liší v jednoduchosti použití. Veškeré prvky se ukládají do mřížky a programátor pomocí několika parametrů řídí jejich umístění. Velmi snadno se tak s tímto layout managerem sestavují formuláře.



Obrázek 6.1: MiG Layout, příklad jednoduchého formuláře

Příklad snadného použití je uveden na jednoduchém formuláři (obrázek 6.1). Umístění komponent a jejich chování je dáno parametry předanými metodou `add()`. Parametr `wrap` zajišťuje zalomení na další řádek. Pokud je nutné spojit 2 sousední buňky do jedné, použije se parametr `split 2`. Roztažení buňky přes 2 pozice zajistí parametr `span 2`. Roztažení obsahu buňky provádí parametr `growx`. Posledním použitým parametrem je center – nastaví zarovnání buňky na střed. Více příkladů v MiG Layout Quick Start Guide [25].

```
/* Inicializace GUI komponent vynechána */
setLayout(new MigLayout()); // Nastavení layout manageru

/* Přidání komponent do dialogu */
add(jCheckBoxShutdown, "split 2, wrap");
add(jLabelIp);
add(jTextFieldIp, "wrap");
add(jLabelClockrate);
add(jComboBoxClockrate, "wrap, growx");
add(jLabelBandwidth);
add(jSpinnerBandwidth, "wrap, growx");
add(jLabelDescription);
add(jTextFieldDescription, "wrap");
add(jButtonOk, "split 2, span 2, center");
add(jButtonCancel);
```

6.2 JUNG - Java Universal Network/Graph Framework

Pro potřeby zobrazení a editace topologie sítě bylo třeba najít volně dostupnou knihovnu či framework, která zajistí uložení topologie do grafu a následně ji dokáže vykreslit do grafického uživatelského rozhraní. Open-source framework JUNG tyto požadavky splňuje – obsahuje abstraktní reprezentaci různých grafů a jejich vizualizátor. Vizualizace dat a jejich reprezentace jsou tak oddělené. Framework je dostupný v repozitářích nástroje Maven. Příklad vytvoření neorientovaného multigrafu a vložení vrcholů (zařízení) a hrany (spojení):

```
UndirectedSparseMultigraph<Device, Connection> graph;
Device device1, device2;

graph = new UndirectedSparseMultigraph();
graph.addVertex(device1);
graph.addVertex(device2);
graph.addEdge(new Connection(), device1, device2);
```

Pro vizualizaci se využívá komponenta VisualizationViewer, většina parametrů se nastavuje pomocí Transformerů – převádí jednotlivé objekty na objekt jiného typu (popisky, barvy atd.). Příklad jednoduché vizualizace výše nadefinovaného grafu:

```
VisualizationViewer<Device, Connection> vv;
vv = new VisualizationViewer(layout, new Dimension(550, 420));

// získání popisku zařízení (vrcholu)
Transformer<Device, String> labelTransformer;
labelTransformer = new Transformer<Device, String>() {
  @Override
  public String transform(Device dev) {
  return dev.getHostname();
  }
};
vv.getRenderContext().setVertexLabelTransformer(labelTransformer);
```

```
// nastavení pozice popisku
vv.getRenderer().getVertexLabelRenderer().setPosition(
  Renderer.VertexLabel.Position.S);

// získání tooltipu spojení (hrany)
Transformer<Connection, String> tooltipTransformer;
tooltipTransformer = new Transformer<Connection, String>() {
  @Override
  public String transform(Connection conn) {
    return conn.getLabel();
  }
};
vv.setEdgeToolTipTransformer(edgeTooltipTransformer);

// nastavení pozadí
vv.setBackground(Color.WHITE);
```

Po nastavení se objekt vizualizátoru přidá do layoutu jako jakákoliv standardní Swing komponenta.

Více informací o frameworku na [26].

6.3 Souborový vstup a výstup

Pro potřeby souborového vstupu a výstupu bylo zvoleno ukládání a načítání do souborů XML, z důvodu dobré čitelnosti a přenositelnosti. XML soubory lze též snadno validovat na správnost dat.

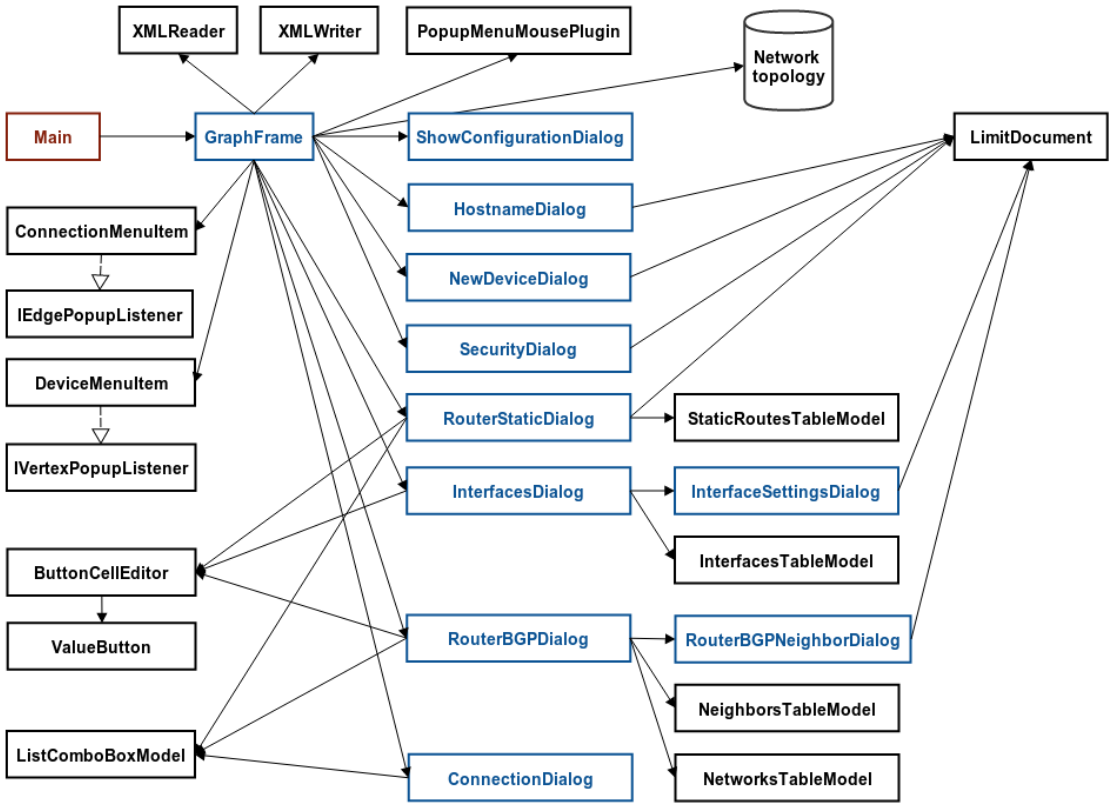
Ukázka vstupně/výstupního souboru s uloženou jednoduchou topologií se nachází v příloze A.

7 Realizace aplikace

V následujících kapitolách jsou zobrazeny diagramy tříd, popsány všechny třídy aplikace a představeny možnosti rozšíření aplikace.

7.1 Diagram tříd GUI

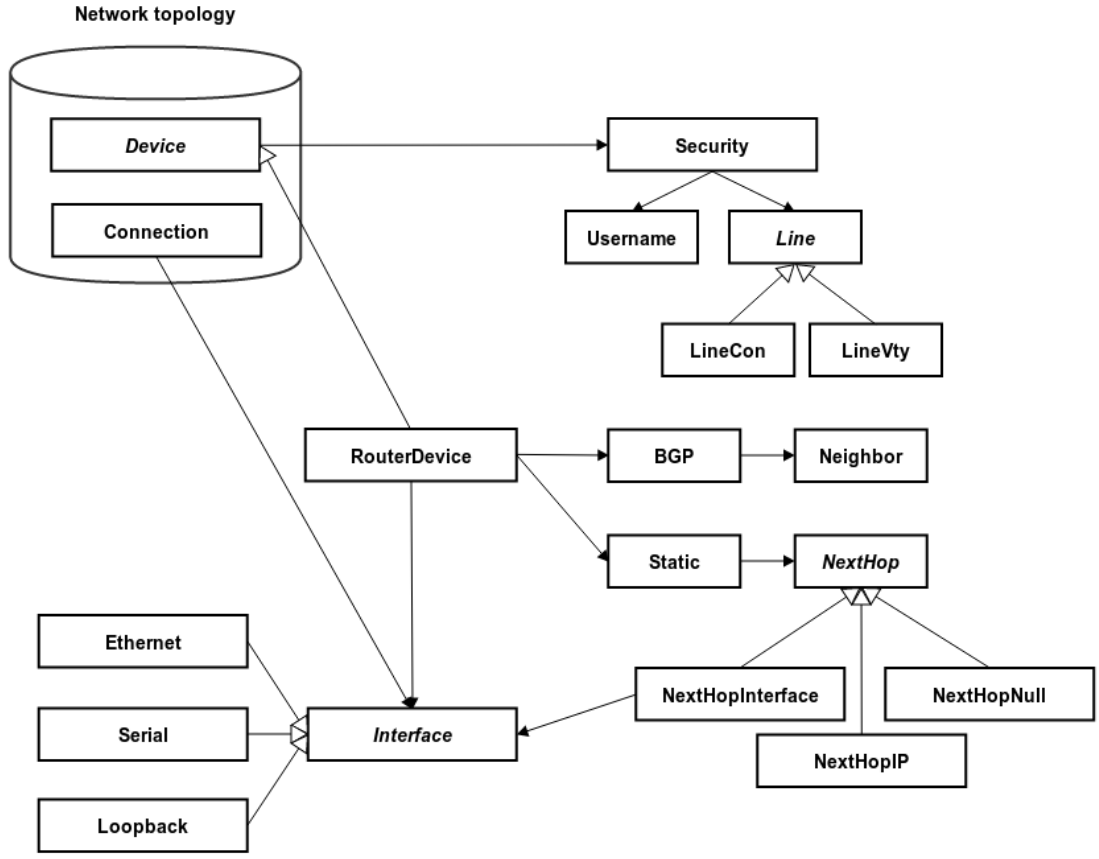
Diagram tříd (obrázek 7.1) zobrazuje vztahy mezi třídami grafického uživatelského rozhraní. Modře jsou vykresleny hlavní prvky grafického uživatelského rozhraní (JFrame, JDialog), červeně hlavní třída Main. Graf topologie (zobrazen jako datové úložiště) ukládá datové struktury, jejichž objektový diagram se nachází v následující kapitole 7.2.



Obrázek 7.1: Diagram tříd GUI

7.2 Diagram datových tříd

Diagram datových tříd (obrázek 7.2) zobrazuje vztahy mezi datovými třídami.



Obrázek 7.2: Diagram datových tříd

7.3 Popis tříd

Kapitola poskytuje popis všech tříd aplikace. Třídy jsou logicky rozděleny do několika balíků.

Datové třídy jsou uloženy do balíku `org.malecek.ciscoconf.device` a dalších balíků jemu podřazených. Obsahují datovou strukturu síťového zařízení.

V balíku `org.malecek.ciscoconf.gui` a jemu podřazených balících se nacházejí třídy grafického uživatelského rozhraní.

7.3.1 Balík org.malecek.ciscoconf

Main

Hlavní třídou aplikace je třída Main. Zajišťuje inicializaci grafického uživatelského rozhraní (GUI).

7.3.2 Balík org.malecek.ciscoconf.device

Device

Abstraktní třída reprezentující zařízení. Obsahuje základní atributy každého zařízení – jedinečný identifikátor zařízení v aplikaci, jméno zařízení (hostname), zabezpečení a seznamy síťových rozhraní. Metody pro vygenerování síťových rozhraní musí implementovat všichni potomci.

RouterDevice

Potomek třídy Device, reprezentující směrovač. Obsahuje další atributy – seznam statických tras a směrovací protokol BGP. Implementuje metody pro vygenerování fixního počtu síťových rozhraní.

Connection

Třída reprezentující spojení mezi síťovými rozhraními. Obsahuje jediné dva atributy – zdrojové a cílové rozhraní.

7.3.3 Balík org.malecek.ciscoconf.device.interfaces

Interface

Abstraktní třída reprezentující síťové rozhraní. Obsahuje základní atributy síťového rozhraní – název, popis, IP adresu, příznak vypnutí a odkaz na rodičovské zařízení. Potomci musí implementovat metodu pro získání textové konfigurace rozhraní.

Ethernet, Loopback

Potomci třídy Interface, reprezentující síťové rozhraní typu Ethernet a virtuální síťové rozhraní Loopback. Implementují pouze metodu pro získání textové konfigurace.

Serial

Potomek třídy Interface, reprezentující síťové rozhraní typu Serial. Obsahuje dva další atributy, clock rate a bandwidth. Implementuje metodu pro získání textové konfigurace.

7.3.4 Balík org.malecek.ciscoconf.device.router.bgp

BGP

Třída uchovávající informace o směrovacím protokolu BGP. Jejími atributy jsou číslo autonomního systému ASN, příznak propagování výchozí cesty, seznam BGP sousedů a seznam propagovaných sítí. Obsahuje metodu pro získání textové konfigurace směrovacího protokolu.

Neighbor

Třída reprezentující BGP souseda. Uchovává atributy souseda – IP adresu, popis, číslo vzdáleného autonomního systému, MD5 heslo a atributy protokolu BGP MED a lokální preferenci.

7.3.5 Balík org.malecek.ciscoconf.device.router.staticroutes

Static

Třída představující statickou trasu. Má dva atributy – cílovou síť a odkaz na následující skok (next-hop). Implementuje metodu pro získání textové konfigurace statické trasy.

NextHop

Abstraktní třída reprezentující následující skok statické trasy. Všichni potomci musí implementovat metodu pro získání názvu následujícího skoku.

NextHopIP, NextHopInterface, NextHopNull

Potomci třídy NextHop, reprezentující následující skok v různých formách. NextHopIP a NextHopInterface obsahují jediný atribut - následující skok ve formě IP adresy či síťového rozhraní. NextHopNull nemá žádný atribut. Implementuje metodu pro získání názvu následujícího skoku.

7.3.6 Balík org.malecek.ciscoconf.device.security

Security

Třída reprezentující nastavení zabezpečení síťového zařízení. Jejími atributy jsou uživatelský účet, heslo do privilegovaného módu, přístupové kanály a příznaky aktivovaných služeb. Implementuje metodu pro získání textové konfigurace zabezpečení zařízení.

Username

Třída reprezentující uživatelský účet. Jako atributy uchovává uživatelské jméno a heslo. Implementuje metodu pro získání textové konfigurace uživatelského účtu.

Line

Abstraktní třída reprezentující přístupový kanál do zařízení.

LineCon

Potomek třídy Line, reprezentuje fyzickou konzoli zařízení. Implementuje metodu pro získání textové konfigurace fyzické konzole.

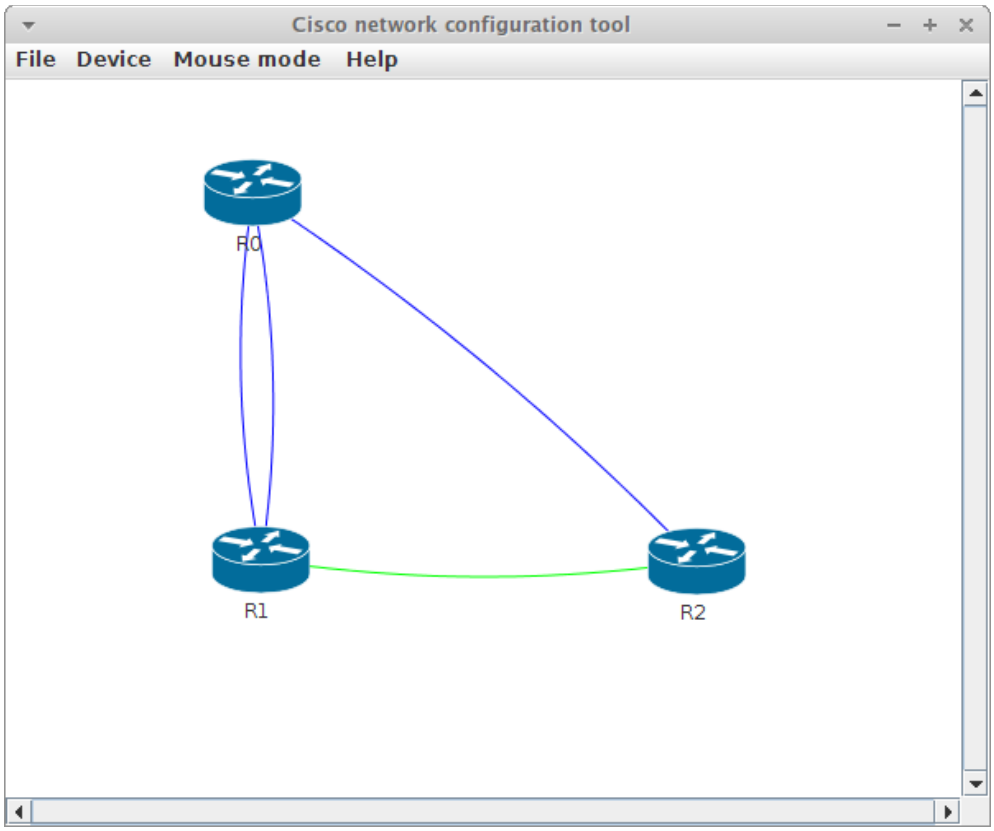
LineVty

Potomek třídy Line, reprezentuje virtuální konzole zařízení. Obsahuje navíc dva atributy – příznak aktivované služby telnet a SSH. Implementuje metodu pro získání textové konfigurace virtuální konzole.

7.3.7 Balík org.malecek.ciscoconf.gui

GraphFrame

Potomek třídy JFrame. Hlavní třída uživatelského rozhraní, vykresluje hlavní okno aplikace (obrázek 7.3). Důležitými atributy jsou graf (obsahující všechny zařízení a spojení) a vizualizační komponenta vykreslující graf.



Obrázek 7.3: GUI, GraphFrame

7.3.8 Balík org.malecek.ciscoconf.gui.dialog

ConnectionDialog

Potomek třídy JDialog. Vykresluje dialog pro přidání nového spojení mezi zařízeními. Dostupná cílová rozhraní jsou generována automaticky – při výběru zdrojového rozhraní se načtou zařízení, která mají volné cílové rozhraní stejného typu. Po vybrání cílového zařízení se načtou tato volná rozhraní. Pokud nejsou žádná rozhraní dostupná, vypíše se uživateli chybová hláška.

HostnameDialog

Potomek třídy JDialog. Vykresluje dialog pro změnu jména (hostname) stávajícího zařízení. Obsahuje metodu pro validaci uživatelského vstupu.

InterfacesDialog

Potomek třídy JDialog. Vykresluje dialog s tabulkovým výpisem všech fyzických a virtuálních síťových rozhraní. Umožňuje otevření dialogu s podrobnou konfigurací síťového rozhraní.

InterfaceSettingDialog

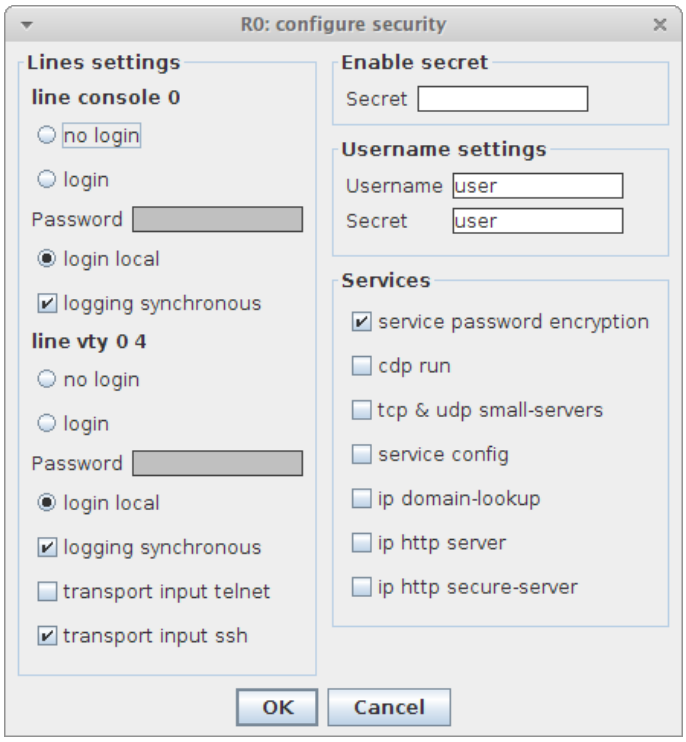
Potomek třídy JDialog. Vykresluje dialog s nastavením podrobných parametrů síťového rozhraní. Obsahuje metodu pro validaci uživatelských vstupů.

NewDeviceDialog

Potomek třídy JDialog. Vykresluje dialog pro přidání nového zařízení. Obsahuje metodu pro validaci uživatelského vstupu.

SecurityDialog

Potomek třídy JDialog. Vykresluje dialog s nastavením zabezpečení zařízení (obrázek 7.4). Obsahuje metodu pro validaci uživatelských vstupů.



Obrázek 7.4: GUI, SecurityDialog

RouterBGPDialog

Potomek třídy JDialog. Vykresluje dialog s nastavením směrovacího protokolu BGP a tabulkami s BGP sousedy a propagovanými sítěmi. Nechybí formuláře pro jejich přidání. Umožňuje otevření dialogu s podrobnou konfigurací BGP souseda. Obsahuje metodu pro validaci uživatelských vstupů.

RouterBGPNeighborDialog

Potomek třídy JDialog. Vykresluje dialog s nastavením podrobných parametrů BGP souseda. Obsahuje metodu pro validaci uživatelských vstupů.

RouterStaticDialog

Potomek třídy JDialog. Vykresluje dialog s tabulkovým výpisem statických tras a formulář pro jejich přidání. Obsahuje metodu pro validaci uživatelských vstupů.

ShowConfigurationDialog

Potomek třídy JDialog. Vykresluje dialog s výpisem textové konfigurace zařízení. Umožňuje uložení konfigurace do souboru.

7.3.9 Balík org.malecek.ciscoconf.gui.document

LimitDocument

Potomek třídy PlainDocument. Omezuje vstupní řetězec na N znaků, toto omezení se nastavuje v konstruktoru třídy. Využito u textových polí JTextField u většiny formulářů v aplikaci.

7.3.10 Balík org.malecek.ciscoconf.gui.listener

IEdgePopupMenuListener

Rozhraní s jedinou metodou setEdge(). Rozhraní musí implementovat třída, do níž se má nastavit objekt hrany (spojení). Využito v položkách kontextového menu spojení.

IVertexPopupMenuListener

Rozhraní s jedinou metodou setVertex(). Rozhraní musí implementovat třída, do níž se má nastavit objekt vrcholu (zařízení). Využito v položkách kontextového menu zařízení.

7.3.11 Balík org.malecek.ciscoconf.gui.menu

ConnectionMenuItem

Potomek třídy JMenuItem, implementuje rozhraní IEdgePopupMenuListener. Přidává atribut – odkaz na spojení. Využito v grafu, kde se po kliknutí na určité spojení toto nastaví do ConnectionMenuItem, odkud se předává obslužné události.

DeviceMenuItem

Potomek třídy JMenuItem, implementuje rozhraní IVertexPopupMenuListener. Přidává atribut – odkaz na zařízení. Využito v grafu, kde se po kliknutí na určité zařízení toto nastaví do DeviceMenuItem, odkud se předává obslužné události.

7.3.12 Balík org.malecek.ciscoconf.gui.model

ListComboBoxModel

Potomek třídy DefaultComboBoxModel. Používá seznam jako datový model pro JComboBox.

7.3.13 Balík org.malecek.ciscoconf.gui.plugin

PopupMenuMousePlugin

Třída implementující plugin obsluhující události myši ve vizualizátoru grafu. Vyhodnotí událost (pravý klik v oblasti grafu), v případě kliknutí na zařízení či na spojení nastaví tento objekt do položek daného menu a menu zobrazí na místě kliknutí.

7.3.14 Balík org.malecek.ciscoconf.gui.table

ButtonCellEditor

Potomek třídy AbstractCellEditor, implementuje rozhraní TableCellEditor a TableCellRenderer. Vykresluje buňku tabulky jako tlačítko ValueButton s popiskem nastaveným v konstruktoru. Mapuje množinu tlačítek na různé objekty. Implementuje metodu pro nastavení obsluhy události po kliknutí na tlačítko.

InterfacesTableModel

Potomek třídy AbstractTableModel. Používá seznam rozhraní jako datový model tabulky. Využito v dialogu s tabulkovým výpisem síťových rozhraní.

NeighborsTableModel, NetworksTableModel

Potomci třídy AbstractTableModel. Používají seznam BGP sousedů nebo seznam propagovaných sítí jako datový model tabulky. Využito v dialogu nastavení směrovacího protokolu BGP.

StaticRoutesTableModel

Potomek třídy AbstractTableModel. Používá seznam statických tras jako datový model tabulky. Využito v dialogu s tabulkovým výpisem statických tras.

ValueButton

Potomek třídy JButton. Uchovává hodnotu objektu předanou v konstruktoru, využito v třídě ButtonCellEditor (pro editaci či smazání objektu).

7.3.15 Balík org.malecek.ciscoconf.utils

IPv4

Třída reprezentující IPv4 adresu. Obsahuje mnoho metod pro práci s IP adresami - např. získání IP adresy v CIDR (Classless Inter-Domain Routing) formátu, získání adresy sítě, získání adresy všesměrového vysílání, získání masky podsítě a další. Při vytvoření objektu kontroluje správnost zadané IP adresy.

Třída byla převzata z [27] a upravena k použití v aplikaci. Třída je volně šiřitelná s podmínkou uvedení jmen autorů. Autory jsou Saddam Abu Ghaida, Nicolai Tufar a Nico Coetzee.

7.3.16 Balík org.malecek.ciscoconf.xml

XMLReader

Třída načítající XML dokument, který převede na množinu objektů reprezentující topologii zařízení.

XMLWriter

Třída ukládající množinu objektů reprezentující topologii zařízení do XML dokumentu.

7.4 Možnosti rozšíření

Aplikace je naprogramována s důrazem na možnost budoucího rozšíření. V kapitole je popsán postup pro přidání parametru zabezpečení a pro přidání dalšího směrovacího protokolu.

7.4.1 Přidání nového parametru zabezpečení

Postup pro přidání nového parametru zabezpečení:

1. Přidání atributu ve třídě Security, vytvoření getterů a setterů, modifikace metody `getConfiguration()` tak, aby zahrnovala konfiguraci nového atributu.
2. Úprava dialogu SecurityDialog:
   1. Přidání prvků umožňující nastavení parametru (JCheckBox, JTextField) a jejich začlenění do stávajícího formuláře.
   2. Úprava metody `fillForm()` tak, aby při načtení dialogu vyplnila hodnotu nového atributu do formuláře.
   3. Úprava metody `checkForm()` tak, aby při odeslání formuláře zkontrolovala nový atribut a v případě chybného vyplnění vrátila chybovou hlášku.
3. Modifikace tříd XMLReader a XMLWriter tak, aby umožnily uložení a načtení nového parametru.

7.4.2 Přidání nového směrovacího protokolu

Postup pro přidání nového směrovacího protokolu:

1. Vytvoření datové třídy, uchovávající parametry směrovacího protokolu.
2. Přidání nově vytvořené třídy jako atributu do třídy zařízení.
3. Modifikace metody pro získání textové konfigurace zařízení tak, aby zahrnovala i nový směrovací protokol.
4. Vytvoření nové třídy dialogu pro nastavení směrovacího protokolu, včetně vyplnění formuláře a jeho kontroly.
5. Vytvoření položky menu v třídě GraphFrame, její začlenění do menu pro zařízení. Implementace obsluhy události (spuštění dialogu).
6. Modifikace tříd XMLReader a XMLWriter tak, aby umožnily uložení a načtení nového směrovacího protokolu.

8 Ověření funkčnosti

Ověření funkčnosti aplikace je rozděleno do dvou hlavních částí – první se věnuje testování uživatelských vstupů, druhá ověření vygenerovaných konfigurací na reálných síťových zařízeních.

8.1 Ošetření uživatelských vstupů

Veškeré uživatelské vstupy musí být ošetřeny tak, aby uživatel nemohl zadat nesprávná či nevyhovující data. Vstupy se ověřují před uložením dat.

8.1.1 Jméno zařízení

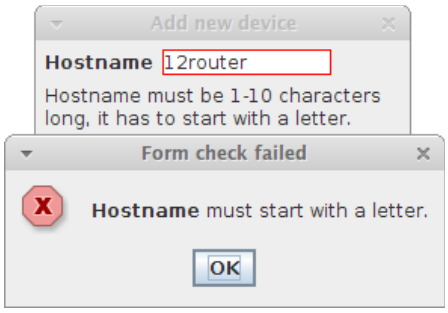
Jméno zařízení (hostname) se zadává při vložení nového zařízení či při jeho změně. Hostname nesmí být prázdné, musí začínat písmenem, nesmí obsahovat speciální znaky kromě tečky, pomlčky nebo podtržítka. Délka je omezena na 10 znaků, formulář nepovolí zadání delšího řetězce. V tabulce 8.1 jsou zadány testovací vstupy s očekávanými výsledky.

Tabulka 8.1: Testování vstupů - jméno zařízení

Vstup: jméno zařízení	**Očekávaný výsledek**
Prázdný vstup	Chyba – prázdný vstup
`12router`	Chyba – nezačíná písmenem
`router!12`	Chyba – obsahuje speciální znak
`router12`	OK – uložení zařízení

Jméno zařízení: výsledky testování

Výstupy testování odpovídají očekávaným výsledkům. Jeden z chybových výstupů je pro ilustraci zobrazen na obrázku 8.1, další výstupy jsou obdobného charakteru. Při chybném vstupu je pole ve formuláři zvýrazněno a uživateli je zobrazena chybová hláška. Pokud je vstup korektní, zařízení je přidáno.



Obrázek 8.1: Testování vstupů, hostname (chybný vstup)

8.1.2 Uživatelský účet

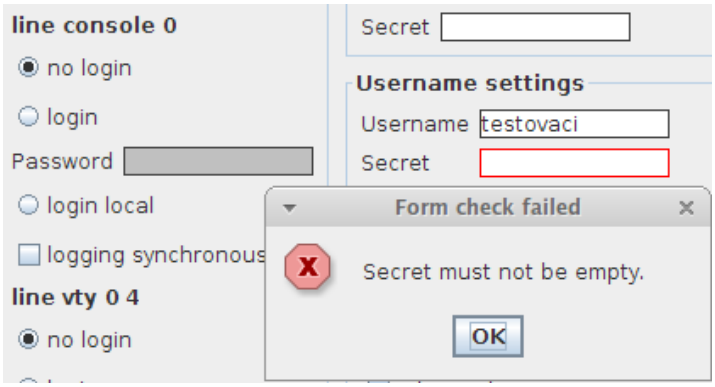
Jméno uživatele a heslo (tvořící uživatelský účet) se zadávají v dialogu zabezpečení. Obě pole musí být vyplněna, pokud je nějaký z přístupových kanálů nastaven na lokální přihlášení (`login local`). V ostatních případech mohou zůstat nevyplněná. Pokud je ale vyplněn alespoň jeden z údajů, je vyžadován i druhý. Obě pole nesmí obsahovat otazník. Délka obou polí je omezena – jméno uživatele na 10 znaků, heslo na 20 znaků. Formulář nepovolí zadání delších řetězců. V tabulce 8.2 jsou zadány testovací vstupy s očekávanými výsledky.

Tabulka 8.2: Testování vstupů - uživatelský účet

Vstup: uživatelský účet	**Očekávaný výsledek**
Lokální přihlášení `line con 0` Jméno ani heslo nevyplněno	Chyba – prázdný vstup
Jméno `testovaci`, heslo nevyplněno	Chyba – prázdný vstup
Jméno `testo?vaci`, heslo `testovaci`	Chyba – speciální znak
Vypnuté přihlášení `line con 0` Jméno ani heslo nevyplněno	OK – uložení nastavení
Lokální přihlášení `line con 0` Jméno `testovaci`, heslo `testovaci`	OK – uložení nastavení

Uživatelský účet: výsledky testování

Výstupy testování odpovídají očekávaným výsledkům. Jeden z chybových výstupů je pro ilustraci zobrazen na obrázku 8.2, další výstupy jsou obdobného charakteru. Při chybném vstupu jsou pole ve formuláři zvýrazněna a uživateli je zobrazena chybová hláška. Pokud je vstup korektní, hodnoty jsou uloženy.



Obrázek 8.2: Testování vstupů, uživatelský účet (chybný vstup)

8.1.3 IP adresa

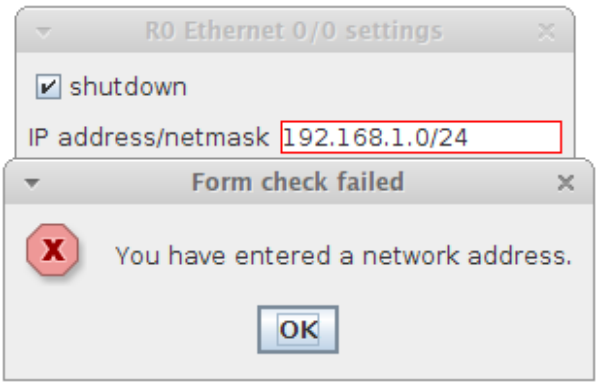
IP adresa se zadává v několika dialozích – např. v dialogu nastavení síťového rozhraní. Musí být zadána v CIDR (Classless Inter-Domain Routing) tvaru – čili IP adresa/délka prefixu v bitech. V tabulce 8.3 jsou zadány testovací vstupy s očekávanými výsledky.

Tabulka 8.3: Testování vstupů - IP adresa

Vstup: IP adresa	**Očekávaný výsledek**
`192.168.1`	Chyba – neplatná adresa
`192.168.1.0/24`	Chyba – adresa sítě
`192.168.1.255/24`	Chyba – adresa všesměrového vysílání
`192.168.1.1/24`	OK – uložení nastavení

IP adresa: výsledky testování

Výstupy testování odpovídají očekávaným výsledkům. Jeden z chybových výstupů je pro ilustraci zobrazen na obrázku 8.3, další výstupy jsou obdobného charakteru. Při chybném vstupu je pole ve formuláři zvýrazněno a uživateli je zobrazena chybová hláška. Pokud je vstup korektní, hodnota je uložena.



Obrázek 8.3: Testování vstupů, IP adresa (chybný vstup)

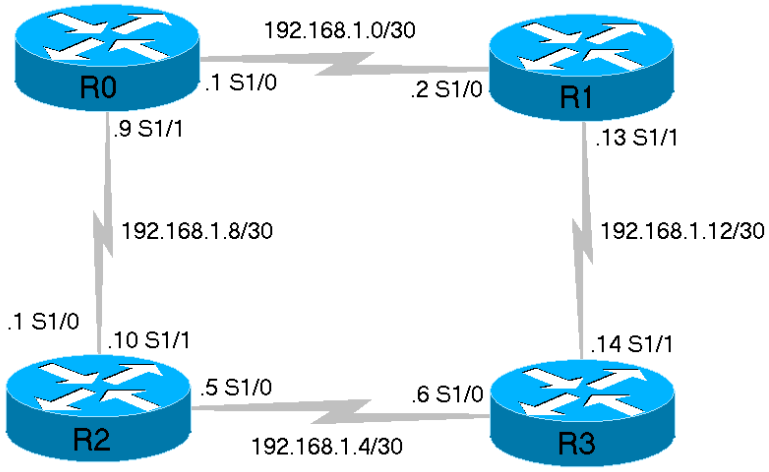
8.2 Ověření vygenerovaných konfigurací

Vygenerované konfigurace jsou otestovány na reálném operačním systému Cisco IOS na několika různých topologiích. Výpisy konfigurací směrovačů vybrané kapitoly (kapitola 8.2.3, Dual-homed BGP) jsou v příloze B. Všechny vygenerované konfigurace jsou k dispozici na přiloženém DVD.

Příkazy zadávané uživatelem k ověření konfigurace jsou zvýrazněny `červeně`, zajímavé výstupy, potvrzující funkčnost vygenerované konfigurace jsou zvýrazněny `zeleně`.

8.2.1 Zabezpečení

Konfigurace zabezpečení je otestována na jednoduché topologii se čtyřmi vzájemně propojenými směrovači. Topologii přibližuje obrázek 8.4.



Obrázek 8.4: Testování konfigurací, bezpečnost - topologie

Směrovače mají různě nastavené služby a ověřování, konkrétní nastavení jednotlivých směrovačů je v tabulce 8.4.

Tabulka 8.4: Testování konfigurací - bezpečnost - nastavení směrovačů

Služba	**R0**	**R1**	**R2**	**R3**
Fyzická konzole: heslo	✔			
Fyzická konzole: lokální přihlášení				✔
Virtuální konzole: heslo		✔		
Virtuální konzole: lokální přihlášení			✔	✔
Virtuální konzole: `transport input none`	✔			
Virtuální konzole: `transport input telnet`		✔		
Virtuální konzole: `transport input ssh`			✔	
Virtuální konzole: `transport input telnet ssh`				✔
`enable secret`		✔		
`service password-encryption`	✔			
`cdp run`	✔	✔		
`tcp-small-servers` `udp-small-servers`			✔	
`service config`			✔	
`ip domain-lookup`			✔	
`ip http server`				✔
`ip http secure-server`				✔

R0: Fyzická konzole, zabezpečení heslem

Připojení k fyzické konzoli směrovače R0 je chráněno heslem, jak potvrzuje výpis.

```
R0 con0 is now available
 Press RETURN to get started.
User Access Verification

Password: heslo
R0>
```

R0: Virtuální konzole, vypnutý telnet a SSH

Připojení k virtuální konzoli směrovače R0 ze směrovače R1 není možné, protože směrovač R0 má obě služby (telnet i SSH) vypnuté. Dle výpisu jsou pokusy o připojení správně odmítnuty.

```
R1>telnet 192.168.1.1
Trying 192.168.1.1 ...
% Connection refused by remote host

R1>ssh -l user 192.168.1.1
% Connection refused by remote host
```

R0: Služba šifrování hesel

Z výpisu běžící konfigurace je zřejmé, že heslo není v čitelné formě – je zašifrované.

```
R0#show running-config | section include line con
line con 0
 password 7 011B03175704
 logging synchronous
 login
```

R0+R1: Protokol CDP

Na směrovačích R0 a R1 je zapnutý protokol CDP. Ve výpisu CDP sousedů směrovače R0 je vidět směrovač R1. Směrovač R2 není vidět, protože má CDP vypnuté.

```
R0>show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater
Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R1               Ser 1/0            163           R       Linux Uni Ser 1/0
```

R1: Telnet, heslo virtuální konzole a heslo do privilegovaného módu

Připojení k virtuální konzoli směrovače R1 pomocí SSH s uživatelem `user` není možné, protože je služba SSH vypnutá. Dle výpisu je pokus o připojení správně odmítnut.

```
R0>ssh -l user 192.168.1.2
% Connection refused by remote host
```

Připojení pomocí telnetu je úspěšné a je vyžadováno heslo, stejně jako heslo do privilegovaného módu, viz výpis.

```
R0>telnet 192.168.1.2
Trying 192.168.1.2 ... Open
User Access Verification

Password: heslo
R1>enable
Password: heslo
R1#
```

R2: SSH, lokální přihlášení

Připojení k virtuální konzoli směrovače R1 pomocí telnetu není možné, protože je služba telnet vypnutá. Dle výpisu je pokus o připojení správně odmítnut.

```
R3>telnet 192.168.1.5
Trying 192.168.1.5 ...
% Connection refused by remote host
```

Připojení přes SSH s uživatelem `testovaci` je naopak úspěšné, je vyžadováno ověření, jak dokazuje výpis.

```
R3>ssh -l testovaci 192.168.1.5
Password: heslo
R2#
```

R2: „Malé“ servery

Směrovač R2 má aktivní služby tzv. „malých“ serverů. Příkladem ve výpisu je test služby echo (posílá zpět vše co mu přijde). Na R1 tyto služby aktivní nejsou, proto je spojení odmítnuto.

```
R3>telnet 192.168.1.5 echo
Trying 192.168.1.5, 7 ... Open
eecchhoo

R3>telnet 192.168.1.13 echo
Trying 192.168.1.13, 7 ...
% Connection refused by remote host
```

R2: Stahování konfigurace ze sítě

Směrovač R2 má aktivní službu stahování konfigurace ze sítě, ve výpisu jsou vidět pokusy o stažení konfigurace z TFTP serveru. Ostatní směrovače mají službu vypnutou.

```
%Error opening tftp://255.255.255.255/network-confg (Timed out)
%Error opening tftp://255.255.255.255/cisconet.cfg (Timed out)
%Error opening tftp://255.255.255.255/r2-confg (Timed out)
%Error opening tftp://255.255.255.255/r2.cfg (Timed out)
```

R2: Překlad DNS názvů

Směrovač R2 má aktivní službu překladu DNS názvů, proto se pokusí o překlad jakéhokoliv názvu, viz výpis. Ostatní směrovače mají službu vypnutou.

```
R2>ping test
Translating "test"...domain server (255.255.255.255)

% Unrecognized host or address, or protocol not running.
```

R3: Telnet + SSH, lokální přihlášení

Připojení k virtuální konzoli směrovače R3 pomocí telnetu i SSH s uživatelem `testovaci` je úspěšné. V obou případech je vyžadováno ověření uživatele, jak dokazuje výpis.

```
R2>telnet 192.168.1.6
Trying 192.168.1.6 ... Open

User Access Verification

Username: testovaci
Password: heslo
R3#

R2>ssh -l testovaci 192.168.1.6
Password: heslo
R3#
```

R3: Fyzická konzole, lokální přihlášení

Připojení k fyzické konzoli směrovače R3 je chráněno. Dle výpisu je nutné se přihlásit uživatelským jménem a heslem.

```
R3 con0 is now available
Press RETURN to get started.

User Access Verification
Username: testovaci
Password: heslo
R3#
```

R3: HTTP a HTTPs server

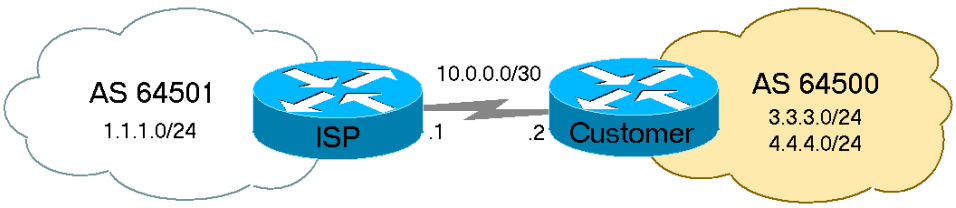
Směrovač R3 má povolen HTTP i HTTPs server, dle výpisů je připojení na výchozí porty 80 i 443 úspěšné.

```
R2>telnet 192.168.1.6 80
Trying 192.168.1.6, 80 ... Open
GET / HTTP/1.0
HTTP/1.1 401 Unauthorized
Date: Fri, 20 Jun 2014 00:42:52 GMT
Server: cisco-IOS
...

R2>telnet 192.168.1.6 443
Trying 192.168.1.6, 443 ... Open
```

8.2.2 Single-homed BGP

Jednoduchá BGP konfigurace (viz obrázek 8.5) je otestována na topologii se dvěma směrovači. Zákazník je připojen jednou linkou do sítě poskytovatele, propaguje své sítě, od poskytovatele dostává pouze výchozí cestu.



Obrázek 8.5: Testování konfigurací, single-homed BGP - topologie

Nastavení BGP sousedů a propagovaných sítí zobrazuje tabulka 8.5.

Tabulka 8.5: Testování konfigurací - single-homed BGP - sítě

	ISP	**Customer**
BGP propagované sítě	0.0.0.0/0 (výchozí cesta)	3.3.3.0/24, 4.4.4.0/24
BGP sousedi	eBGP 10.0.0.1	eBGP 10.0.0.2

BGP sousedi

Dle výpisu mají směrovače navázané vzájemné BGP sousedství.

```
ISP#show ip bgp neighbors
BGP neighbor is 10.0.0.2,  remote AS 64500, external link
  BGP version 4, remote router ID 4.4.4.4
  BGP state = Established, up for 00:01:31

Customer#show ip bgp neighbors
BGP neighbor is 10.0.0.1,  remote AS 64501, external link
  BGP version 4, remote router ID 1.1.1.1
  BGP state = Established, up for 00:02:14
```

Výchozí cesta

Směrovač Customer má dostupnou výchozí cestu propagovanou směrovačem ISP, jak potvrzuje výpis.

```
Customer#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 10.0.0.1 to network 0.0.0.0

     3.0.0.0/24 is subnetted, 1 subnets
C       3.3.3.0 is directly connected, Loopback0
     4.0.0.0/24 is subnetted, 1 subnets
C       4.4.4.0 is directly connected, Loopback1
     10.0.0.0/30 is subnetted, 1 subnets
C       10.0.0.0 is directly connected, Serial1/0
B*   0.0.0.0/0 [20/0] via 10.0.0.1, 00:03:00
```

Dostupnost sítě 1.1.1.0/24 byla ověřena nástrojem `ping`, viz výpis.

```
Customer#ping 1.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/24 ms
```

Zákaznické sítě

Dle výpisů má směrovač ISP cestu do zákaznických sítí – směrovač Customer je tedy správně propaguje. Dostupnost zákaznických sítí byla ověřena nástrojem `ping`.

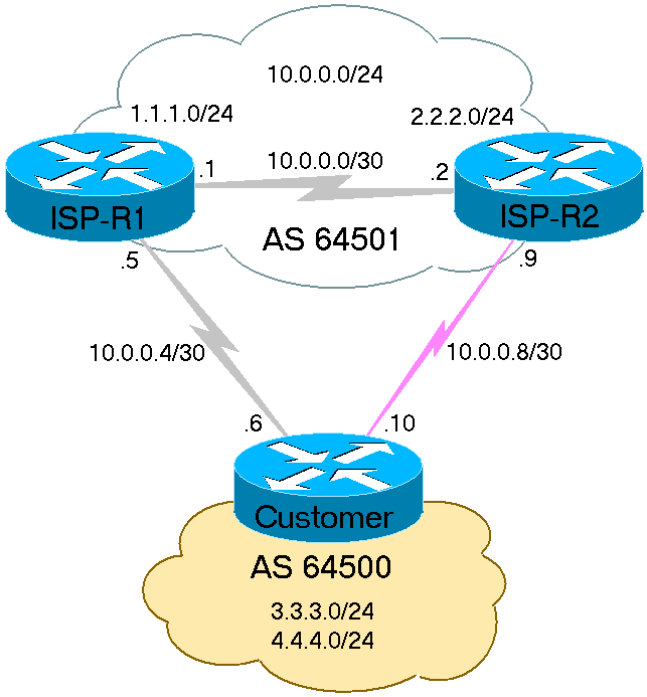
```
ISP#show ip bgp
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
*> 0.0.0.0          0.0.0.0                  0         32768 i
*> 3.3.3.0/24       10.0.0.2                 0             0 64500 i
*> 4.4.4.0/24       10.0.0.2                 0             0 64500 i

ISP#ping 3.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/19/20 ms

ISP#ping 4.4.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/20/24 ms
```

8.2.3 Dual-homed BGP

Druhá BGP konfigurace je otestována na topologii se třemi směrovači (obrázek 8.6). Zákazník je připojen dvěma linkami k AS poskytovatele. Propaguje své sítě, od poskytovatele dostává cesty do všech sítí. Obě linky jsou rovnocenné, ve výchozím nastavení by byla preferována trasa ke směrovači ISP-R1. O použití této trasy by rozhodlo pravidlo o nižším BGP identifikátoru (viz kapitola 4.7 - Volba nejlepší cesty). Zákazník ovšem chce preferovat pravou trasu pro odchozí i příchozí provoz – této preference se docílí modifikací parametrů MED a lokální preference. BGP relace na preferované lince je chráněna heslem.



Obrázek 8.6: Testování konfigurací, dual-homed BGP - topologie

Nastavení BGP sousedů, propagovaných sítí a předpokládaných BGP identifikátorů zobrazuje tabulka 8.6.

Tabulka 8.6: Testování konfigurací - dual-homed BGP - sítě

	ISP-R1	**ISP-R2**	**Customer**
BGP propagované sítě	10.0.0.0/24 1.1.1.0/24	10.0.0.0/24 2.2.2.0/24	3.3.3.0/24 4.4.4.0/24
BGP sousedi	iBGP 10.0.0.2 eBGP 10.0.0.6	iBGP 10.0.0.1 eBGP 10.0.0.10	eBGP 10.0.0.5 eBGP 10.0.0.9
BGP identifikátor	1.1.1.1	2.2.2.2	4.4.4.4

BGP sousedi

Všechny směrovače mají dle následujících výpisů očekávané BGP sousedy.

```
ISP-R1#show ip bgp neighbors
BGP neighbor is 10.0.0.2,  remote AS 64501, internal link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:42:41
...
BGP neighbor is 10.0.0.6,  remote AS 64500, external link
  BGP version 4, remote router ID 4.4.4.4
  BGP state = Established, up for 00:37:04

ISP-R2#show ip bgp neighbors
BGP neighbor is 10.0.0.1,  remote AS 64501, internal link
  BGP version 4, remote router ID 1.1.1.1
  BGP state = Established, up for 00:44:57
...
BGP neighbor is 10.0.0.10,  remote AS 64500, external link
  BGP version 4, remote router ID 4.4.4.4
  BGP state = Established, up for 00:37:32

Customer#show ip bgp neighbors
BGP neighbor is 10.0.0.5,  remote AS 64501, external link
  BGP version 4, remote router ID 1.1.1.1
  BGP state = Established, up for 00:40:52
```

```
...
BGP neighbor is 10.0.0.9,  remote AS 64501, external link
  BGP version 4, remote router ID 2.2.2.2
  BGP state = Established, up for 00:39:09
```

Dostupné a nejlepší cesty

Výpisy potvrzují, že směrovače mají dostupné všechny sítě. Směrovač ISP-R1 – sítě zákazníka jsou směrovány přes směrovač ISP-R2 (vybrány cesty s nižším parametrem MED), směrovač Customer – odchozí provoz je směrován taktéž přes směrovač ISP-R2 (vybrány cesty s vyšší lokální preferencí).

```
ISP-R1#show ip bgp
BGP table version is 20, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       0.0.0.0                  0         32768 i
*>i2.2.2.0/24       10.0.0.2                 0    100      0 i
*>i3.3.3.0/24       10.0.0.10              100    100      0 64500 i
*                   10.0.0.6              1000             0 64500 i
*>i4.4.4.0/24       10.0.0.10              100    100      0 64500 i
*                   10.0.0.6              1000             0 64500 i
*> 10.0.0.0/24      0.0.0.0                  0         32768 i
* i                 10.0.0.2                 0    100      0 i


ISP-R2#sh ip bgp
BGP table version is 18, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
*>i1.1.1.0/24       10.0.0.1                 0    100      0 i
*> 2.2.2.0/24       0.0.0.0                  0         32768 i
*> 3.3.3.0/24       10.0.0.10              100             0 64500 i
*> 4.4.4.0/24       10.0.0.10              100             0 64500 i
* i10.0.0.0/24      10.0.0.1                 0    100      0 i
*>                  0.0.0.0                  0         32768 i
```

```
Customer#sh ip bgp
BGP table version is 6, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       10.0.0.9                 1000      0 64501 i
*                   10.0.0.5                 0    100      0 64501 i
*> 2.2.2.0/24       10.0.0.9                 0    1000      0 64501 i
*                   10.0.0.5                      100      0 64501 i
*> 3.3.3.0/24       0.0.0.0                  0         32768 i
*> 4.4.4.0/24       0.0.0.0                  0         32768 i
*> 10.0.0.0/24      10.0.0.9                 0    1000      0 64501 i
*                   10.0.0.5                 0    100      0 64501 i
```

Ověření preference pravé cesty

Preference pravé cesty je ověřena nástrojem `traceroute` ze směrovačů ISP-R1 a Customer (viz výpis). Příchozí i odchozí provoz do sítě zákazníka je skutečně směrován přes směrovač ISP-R2.

```
ISP-R1#traceroute 3.3.3.3
Type escape sequence to abort.
Tracing the route to 3.3.3.3
  1 10.0.0.2 20 msec 20 msec 20 msec
  2 10.0.0.10 40 msec *  36 msec

Customer#traceroute 1.1.1.1
Type escape sequence to abort.
Tracing the route to 1.1.1.1
  1 10.0.0.9 [AS 64501] 16 msec 16 msec 24 msec
  2 10.0.0.1 [AS 64501] 36 msec *  32 msec
```

ISP-R2+Customer: BGP relace chráněná heslem

Relaci chráněnou heslem bohužel není možné snadno ověřit. Pro ověření je na jedné straně heslo změněno, poté se vypíše chybová hláška. Po opravě hesla se spojení naváže, jak dokazuje výpis.

```
ISP-R2(config-router)#neighbor 10.0.0.10 password spatne
*Mar  1 03:17:42.291: %TCP-6-BADAUTH: Invalid MD5 digest from
10.0.0.10(179) to 10.0.0.9(59945)
ISP-R2(config-router)#neighbor 10.0.0.10 password heslo2
*Mar  1 03:18:58.135: %BGP-5-ADJCHANGE: neighbor 10.0.0.10 Up
```

8.3 Zhodnocení ověření

Ověření funkce aplikace je provedeno ve dvou částech – ověření uživatelských vstupů a ověření funkčnosti vygenerovaných konfigurací.

Uživatelské vstupy aplikace jsou úspěšně ověřeny na několika sadách testovacích vstupů. Při zadání chybného vstupu je uživateli vypsána chybová hláška a příslušná vstupní pole jsou ve formuláři vyznačena.

Funkčnost vygenerovaných konfigurací je ověřena na několika testovacích topologiích v reálném operačním systému Cisco IOS. První topologie se zaměřuje na zabezpečení zařízení, další dvě na konfiguraci směrovacího protokolu BGP. V topologii pro ověření zabezpečení je úspěšně otestováno připojení na konzole (jak fyzickou, tak virtuální), zabezpečení heslem, ověření uživatelským účtem a služby běžící na zařízeních. Dvě topologie věnované protokolu BGP úspěšně ověřují navázaná sousedství, propagované sítě, propagovanou výchozí cestu a v případě druhé topologie i preferenci jedné linky pomocí změněných parametrů MED a lokální preference.

9 Závěr

Tématem diplomové práce bylo vytvoření nástroje pro konfiguraci aktivních síťových prvků, který bude snadno použitelný pro studenty, nebude omezený potřebou vlastnit fyzické zařízení či jinými licenčními požadavky a bude snadno rozšiřitelný.

Pro splnění cílů bylo nutné prozkoumat několik vybraných nástrojů pro konfiguraci síťových prvků a nalézt jejich slabiny. Jediný z testovaných nástrojů, Juniper J-Web pro konfiguraci síťových prvků značky Juniper, umožňuje v grafickém uživatelském rozhraní nastavit většinu podstatných parametrů síťového zařízení. U nástrojů pro konfiguraci síťových zařízení značky Cisco toto neplatí – směrovací protokoly a zabezpečení je možné nastavit pouze omezeně. Směrovací protokol BGP pak úplně chybí. Pokročilé nastavení stále spoléhá na příkazovou řádku a s tím spojenou nutnost znát zadávané příkazy. Pro počáteční seznámení studentů, kteří s konfigurací síťových prvků nemají praktické zkušenosti, nemusí být příkazová řádka nejvhodnějším nástrojem.

V této práci proto byla navržena a realizována aplikace, která umožňuje nejen vytvoření virtuální topologie, vložení a propojení síťových zařízení, ale hlavně nastavení parametrů a protokolů z grafického uživatelského rozhraní bez nutnosti znát syntaxi příkladů – parametry jednotlivých zařízení se zadávají v přehledných dialozích. Vytvořená aplikace kontroluje všechny vstupy uživatele, nedovolí mu zadat nesprávná data. Přispívá výuce také tím, že se veškeré provedené změny okamžitě projevují ve vygenerované konfiguraci. Aplikace byla navržena s ohledem na budoucí rozšiřitelnost.

Jako správné rozhodnutí se ukázalo použití několika open-source knihoven. Jejich použití značně urychlilo vývoj aplikace. Zejména framework JUNG použitý pro implementaci síťové topologie řeší elegantně vše od abstraktních datových struktur až po výslednou vizualizaci.

Funkčnost aplikace byla ověřena nastavením několika různých topologií, vygenerováním konfigurací a otestováním na reálném operačním systému Cisco IOS.

Byl vytvořen nástroj, který může významně pomoci výuce studentů, kteří se teprve seznamují s konfigurací síťových prvků. Nástroj je snadno rozšiřitelný, umožňuje snadno přidat další konfigurovatelné parametry zařízení nebo směrovací protokoly.

Seznam zkratek

ACL	Access List
AS	Autonomous System
ASN	Autonomous System Number
BGP	Border Gateway Protocol
CCP	Cisco Configuration Professional
CDP	Cisco Discovery Protocol
CIDR	Classless Inter-Domain Routing
CLI	Command Line Interface
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
eBGP	External Border Gateway Protocol
EGP	Exterior Gateway Protocol
EIGRP	Enhanced Interior Gateway Routing Protocol
GUI	Graphical User Interface
iBGP	Internal Border Gateway Protocol
IGP	Interior Gateway Protocol
IOS	Internetwork Operating System
JUNG	Java Universal Network/Graph Framework
MED	Multiple Exit Discriminator
NAT	Network Address Translation
NTP	Network Time Protocol
NVRAM	Non-Volatile Random Access Memory
OSPF	Open Shortest Path First
RADIUS	Remote Authentication Dial In User Service

RIP	Routing Internet Protocol
SDM	Cisco Router and Security Device Manager
SSH	Secure Shell
TACACS+	Terminal Access Controller Access-Control System
TFTP	Trivial File Transfer Protocol
VPN	Virtual Private Network
XML	eXtensible Markup Language

Seznam použité literatury

[1] *Cisco Packet Tracer* [online]. [2014?] [cit. 2014-06-19]. Dostupné z: <https://www.netacad.com/web/about-us/cisco-packet-tracer>

[2] COMER, Douglas. *Internetworking with TCP/IP (Volume 1): Principles, protocols and architectures.* Upper Saddle River, New Jersey, USA: Prentice Hall, 2000. 4th edition. ISBN 0-13-018380-6.

[3] BEASLEY, Jeffrey – NILKAEW, Piyasat. *A Practical Guide to Advanced Networking.* Indianapolis, Indiana, USA: Pearson, 2013. 3rd edition. ISBN 0-7897-4904-1.

[4] PUŽMANOVÁ, Rita. *Moderní komunikační sítě od A do Z.* Praha: Computer Press, 1998. ISBN 80-7226-098-7.

[5] *Route-Maps for IP Routing Protocol Redistribution Configuration* [online]. 2005-08-10 [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/td/docs/security/asa/asa84/configuration85/guide/asa_cfg_cli_85/route_maps.html>

[6] *Cisco Router and Security Device Manager* [online]. [2009?] [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/products/collateral/cloud-systems-management/router-security-device-manager/product_data_sheet0900aecd800fd118.html>

[7] *Cisco Configuration Professional* [online]. [2014?] [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/products/cloud-systems-management/configuration-professional/index.html>

[8] *NTP FAQ* [online]. c2006 [cit. 2014-06-19]. Dostupné z: <http://www.ntp.org/ntpfaq/NTP-s-def.htm>

[9] STALLINGS, William. *Protocol Basics: Secure Shell Protocol* [online]. [2006?] [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_12-4/124_ssh.html>

[10] *Juniper J-Web* [online]. c2014 [cit. 2014-06-19]. Dostupné z: <http://www.juniper.net/us/en/products-services/security/jweb/>

[11] *Cisco Internetwork Operating System (Cisco IOS)* [online]. 2006-02-02 [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/support/docs/ios-nx-os-software/ios-software-releases-110/13178-15.html>

[12] *Using the Command-Line Interface in Cisco IOS Software* [online]. 2010-02-24 [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/td/docs/ios/iproute_bfd/configuration/guide/15_1/irb_15_1_book/usingios.html>

[13] *Cabling Guide for Console and AUX Ports* [online]. 2006-09-03 [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/support/docs/routers/7000-series-routers/12223-14.html>

[14] *Back up and Restore Configuration Files* [online]. 2006-08-03 [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/support/docs/ios-nx-os-software/ios-software-releases-122-mainline/46741-backup-config.html>

[15] MARTINEZ, Julio. *Basic router security* [online]. 2013-10-16 [cit. 2014-06-19]. Dostupné z: *<https://supportforums.cisco.com/document/11936661/basic-router-security>*

[16] LE ROY, Ted. *How to secure a Cisco router* [online]. 2010-03-16 [cit. 2014-06-19]. Dostupné z: <http://www.infosecisland.com/blogview/3309-How-to-secure-a-Cisco-router.html>

[17] BALCHUNAS, Aaron. *Basic Router Security* [online]. Verze 1.11. c2007 [cit. 2014-06-19]. Dostupné z: <http://www.routeralley.com/ra/docs/basic_router_security.pdf>

[18] SINGH, Shashank. *Cisco Guide to Harden Cisco IOS Devices* [online]. 2014-06-03 [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/support/docs/ip/access-lists/13608-21.html>

[19] *Cisco IOS Password Encryption Facts* [online]. 2008-07-22 [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/support/docs/security-vpn/remote-authentication-dial-user-service-radius/107614-64.html>

[20] *TCP and UDP Small Servers* [online]. 2006-08-01 [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/support/docs/ios-nx-os-software/ios-software-releases-110/12815-23.html>

[21] *Configuring Cisco Discovery Protocol* [online]. [2004?] [cit. 2014-06-19]. Dostupné z: <http://www.cisco.com/c/en/us/td/docs/ios/12_2/configfun/configuration/guide/ffun_c/fcf015.html>

[22] GRYGÁREK, Petr. *Přednášky předmětu SPS: Směrovací protokol BGP* [online]. c2005 [cit. 2014-06-19]. Dostupné z: <http://www.cs.vsb.cz/grygarek/SPS/lect/BGP/BGP.html>

[23] HLAVÁČEK, Tomáš – PETR, Emanuel. *Směrovací protokol BGP* [prezentace]. Verze 1.7. 2013-02-27.

[24] *Autonomous System (AS) Numbers* [online]. 2014-06-03 [cit. 2014-06-19]. Dostupné z: <http://www.iana.org/assignments/as-numbers/as-numbers.xhtml>

[25] *MiG Layout Quick Start Guide* [online]. Verze 1.2.2. c2009 [cit. 2014-06-19]. Dostupné z: <http://www.miglayout.com/QuickStart.pdf>

[26] *JUNG: Java Universal Network/Graph Framework* [online]. [2010?] [cit. 2014-06-19]. Dostupné z: <http://jung.sourceforge.net/>

[27] GHAIDA, Saddam – TUFAR, Nicolai. *Java Library for IP/Subnet Calculator* [online]. 2011-10-02 [cit. 2014-06-19]. Dostupné z: <http://www.sghaida.com/java-library-for-ipsubnet-calculator>

Seznam obrázků

Seznam tabulek

Přílohy

Příloha A: Vzorový vstupní/výstupní soubor

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!-- kořenový element - topologie -->
<topology>
 <!-- devices - konfigurace zařízení -->
 <devices>
  <!-- konfigurace jednotlivého zařízení  -->
  <device hostname="R0" id="0" type="router" x="140.0" y="167.0">
   <!-- konfigurace zabezpečení  -->
   <security>
    <!-- běžící služby -->
    <service cdp="true" domain-lookup="true"
             http-secure-server="false" http-server="false"
             password-encryption="false" service-config="true"
             small-servers="false"/>
    <!-- fyzická konzole  -->
    <line-con logging-synchronous="false" login-local="false"/>
    <!-- virtuální konzole -->
    <line-vty logging-synchronous="false" login-local="false"
              ssh="false" telnet="false"/>
   </security>
   <!-- interfaces - konfigurace síťových rozhraní -->
   <interfaces>
    <!-- konfigurace jednotlivého síťového rozhraní -->
    <interface description="R0-&gt;R1" ip="192.168.1.1/24"
               name="Ethernet 0/0" shutdown="false"
               type="ethernet"/>
    <interface name="Ethernet 0/1" shutdown="true"
               type="ethernet"/>
    <interface name="Ethernet 0/2" shutdown="true"
               type="ethernet"/>
    <interface name="Ethernet 0/3" shutdown="true"
               type="ethernet"/>
    <interface name="Serial 1/0" shutdown="true" type="serial"/>
    <interface name="Serial 1/1" shutdown="true" type="serial"/>
    <interface name="Serial 1/2" shutdown="true" type="serial"/>
    <interface name="Serial 1/3" shutdown="true" type="serial"/>
   </interfaces>
```

```
  <!-- loopbacks - konfigurace virtuálních síťových rozhraní -->
  <loopbacks>
   <!-- konfigurace jednotlivého virtuálního rozhraní -->
   <loopback name="Loopback 0" shutdown="true" type="loopback"/>
   <loopback name="Loopback 1" shutdown="true" type="loopback"/>
   <loopback name="Loopback 2" shutdown="true" type="loopback"/>
   <loopback name="Loopback 3" shutdown="true" type="loopback"/>
  </loopbacks>
  <!-- statické směrování -->
  <static-routes>
   <!-- jednotlivé statické trasy -->
   <route destination="0.0.0.0/0" next-hop="Null 0"
          next-hop-type="null"/>
  </static-routes>
  <!-- dynamické směrování -->
  <router>
   <!-- konfigurace protokolu BGP -->
   <router-bgp as="64500" default-information="true">
    <!-- propagované sítě -->
    <networks>
     <network address="0.0.0.0/0"/>
    </networks>
    <!-- BGP sousedi -->
    <neighbors>
     <neighbor ip="192.168.1.2" remote-as="64501"/>
    </neighbors>
   </router-bgp>
  </router>
 </device>
 <!-- konfigurace dalšího zařízení -->
 <device hostname="R1" id="1" type="router" x="472.0" y="168.0">
  <security>
   <service cdp="true" domain-lookup="true"
            http-secure-server="false" http-server="false"
            password-encryption="false" service-config="true"
            small-servers="false"/>
   <line-con logging-synchronous="false" login-local="false"/>
   <line-vty logging-synchronous="false" login-local="false"
             ssh="false" telnet="false"/>
  </security>
```

```
   <interfaces>
    <interface description="R1-&gt;R0" ip="192.168.1.2/24"
               name="Ethernet 0/0" shutdown="false"
               type="ethernet"/>
    <interface name="Ethernet 0/1" shutdown="true"
               type="ethernet"/>
    <interface name="Ethernet 0/2" shutdown="true"
               type="ethernet"/>
    <interface name="Ethernet 0/3" shutdown="true"
               type="ethernet"/>
    <interface name="Serial 1/0" shutdown="true" type="serial"/>
    <interface name="Serial 1/1" shutdown="true" type="serial"/>
    <interface name="Serial 1/2" shutdown="true" type="serial"/>
    <interface name="Serial 1/3" shutdown="true" type="serial"/>
   </interfaces>
   <loopbacks>
    <loopback description="Connected network" ip="1.1.1.1/24"
              name="Loopback 0" shutdown="false" type="loopback"/>
    <loopback name="Loopback 1" shutdown="true" type="loopback"/>
    <loopback name="Loopback 2" shutdown="true" type="loopback"/>
    <loopback name="Loopback 3" shutdown="true" type="loopback"/>
   </loopbacks>
   <router>
    <router-bgp as="64501" default-information="false">
     <networks>
      <network address="192.168.1.0/24"/>
      <network address="1.1.1.0/24"/>
     </networks>
     <neighbors>
      <neighbor ip="192.168.1.1" remote-as="64500"/>
     </neighbors>
    </router-bgp>
   </router>
  </device>
 </devices>
 <!-- spojení mezi zařízeními -->
 <connections>
  <connection dst-id="1" dst-interface="Ethernet 0/0"
              src-id="0" src-interface="Ethernet 0/0"/>
 </connections>
</topology>
```

Příloha B: Vygenerované konfigurace

Dual-homed BGP – ISP-R1

```
!
version 12.4
!
hostname ISP-R1
!
boot-start-marker
boot-end-marker
!
!
service password-encryption
!
!
username isp privilege 15 secret isp
!
!
no cdp run
!
no service tcp-small-servers
no service udp-small-servers
!
no service config
!
no ip domain-lookup
no ip http server
no ip http secure-server
!
line con 0
 login local
 logging synchronous
line vty 0 4
 login local
 logging synchronous
 transport input ssh
!
ip domain-name domain.local
crypto key generate rsa modulus 1024
ip ssh version 2
!
!
!
interface Ethernet 0/0
 no ip address
 shutdown
!
interface Ethernet 0/1
```

```
 no ip address
 shutdown
!
interface Ethernet 0/2
 no ip address
 shutdown
!
interface Ethernet 0/3
 no ip address
 shutdown
!
interface Serial 1/0
 description ISP-R1->ISP-R2
 ip address 10.0.0.1 255.255.255.252
 no shutdown
!
interface Serial 1/1
 description ISP-R1->Customer
 ip address 10.0.0.5 255.255.255.252
 no shutdown
!
interface Serial 1/2
 no ip address
 shutdown
!
interface Serial 1/3
 no ip address
 shutdown
!
interface Loopback 0
 description ISP-R1 NET
 ip address 1.1.1.1 255.255.255.0
 no shutdown
!
interface Loopback 1
 no ip address
 shutdown
!
interface Loopback 2
 no ip address
 shutdown
!
interface Loopback 3
 no ip address
 shutdown
!
!
!
!
ip route 10.0.0.0 255.255.255.0 Null 0
```

```
ip route 10.0.0.8 255.255.255.252 Serial 1/0
!
!
!
!
router bgp 64501
 no synchronization
 bgp log-neighbor-changes
 network 1.1.1.0 mask 255.255.255.0
 network 10.0.0.0 mask 255.255.255.0
 neighbor 10.0.0.2 remote-as 64501
 neighbor 10.0.0.6 remote-as 64500
 no auto-summary
!
!
exception data-corruption buffer truncate
end
```

Dual-homed BGP – ISP-R2

```
!
version 12.4
!
hostname ISP-R2
!
boot-start-marker
boot-end-marker
!
!
service password-encryption
!
!
username isp privilege 15 secret isp
!
!
no cdp run
!
no service tcp-small-servers
no service udp-small-servers
!
no service config
!
no ip domain-lookup
no ip http server
no ip http secure-server
!
line con 0
 login local
 logging synchronous
line vty 0 4
 login local
 logging synchronous
 transport input ssh
!
ip domain-name domain.local
crypto key generate rsa modulus 1024
ip ssh version 2
!
!
!
interface Ethernet 0/0
 no ip address
 shutdown
!
interface Ethernet 0/1
 no ip address
 shutdown
```

```
!
interface Ethernet 0/2
 no ip address
 shutdown
!
interface Ethernet 0/3
 no ip address
 shutdown
!
interface Serial 1/0
 description ISP-R2->ISP-R1
 ip address 10.0.0.2 255.255.255.252
 no shutdown
!
interface Serial 1/1
 no ip address
 shutdown
!
interface Serial 1/2
 description ISP-R2->Customer
 ip address 10.0.0.9 255.255.255.252
 no shutdown
!
interface Serial 1/3
 no ip address
 shutdown
!
interface Loopback 0
 description ISP-R2 NET
 ip address 2.2.2.2 255.255.255.0
 no shutdown
!
interface Loopback 1
 no ip address
 shutdown
!
interface Loopback 2
 no ip address
 shutdown
!
interface Loopback 3
 no ip address
 shutdown
!
!
!
!
ip route 10.0.0.0 255.255.255.0 Null 0
ip route 10.0.0.4 255.255.255.252 Serial 1/0
!
```

```
!
!
!
router bgp 64501
 no synchronization
 bgp log-neighbor-changes
 network 2.2.2.0 mask 255.255.255.0
 network 10.0.0.0 mask 255.255.255.0
 neighbor 10.0.0.1 remote-as 64501
 neighbor 10.0.0.10 remote-as 64500
 neighbor 10.0.0.10 password heslo2
 no auto-summary
!
!
exception data-corruption buffer truncate
end
```

Dual-homed BGP – Customer

```
!
version 12.4
!
hostname Customer
!
boot-start-marker
boot-end-marker
!
!
service password-encryption
!
!
username customer privilege 15 secret customer
!
!
no cdp run
!
no service tcp-small-servers
no service udp-small-servers
!
no service config
!
no ip domain-lookup
no ip http server
no ip http secure-server
!
line con 0
 login local
 logging synchronous
line vty 0 4
 login local
 logging synchronous
 transport input ssh
!
ip domain-name domain.local
crypto key generate rsa modulus 1024
ip ssh version 2
!
!
!
interface Ethernet 0/0
 no ip address
 shutdown
!
interface Ethernet 0/1
 no ip address
 shutdown
```

```
!
interface Ethernet 0/2
 no ip address
 shutdown
!
interface Ethernet 0/3
 no ip address
 shutdown
!
interface Serial 1/0
 no ip address
 shutdown
!
interface Serial 1/1
 description Customer->ISP-R1
 ip address 10.0.0.6 255.255.255.252
 no shutdown
!
interface Serial 1/2
 description Customer->ISP-R2
 ip address 10.0.0.10 255.255.255.252
 no shutdown
!
interface Serial 1/3
 no ip address
 shutdown
!
interface Loopback 0
 description Customer NET1
 ip address 3.3.3.3 255.255.255.0
 no shutdown
!
interface Loopback 1
 description Customer NET2
 ip address 4.4.4.4 255.255.255.0
 no shutdown
!
interface Loopback 2
 no ip address
 shutdown
!
interface Loopback 3
 no ip address
 shutdown
!
!
!
!
!
!
```

```
!
route-map 10.0.0.5-MED
 set metric 1000
route-map 10.0.0.5-LP
 set local-preference 100
route-map 10.0.0.9-MED
 set metric 100
route-map 10.0.0.9-LP
 set local-preference 1000
!
router bgp 64500
 no synchronization
 bgp log-neighbor-changes
 network 3.3.3.0 mask 255.255.255.0
 network 4.4.4.0 mask 255.255.255.0
 neighbor 10.0.0.5 remote-as 64501
 neighbor 10.0.0.5 route-map 10.0.0.5-MED out
 neighbor 10.0.0.5 route-map 10.0.0.5-LP in
 neighbor 10.0.0.9 remote-as 64501
 neighbor 10.0.0.9 route-map 10.0.0.9-MED out
 neighbor 10.0.0.9 route-map 10.0.0.9-LP in
 neighbor 10.0.0.9 password heslo2
 no auto-summary
!
!
exception data-corruption buffer truncate
end
```

Příloha C: Uživatelská příručka

Překlad programu

K překladu programu je nutné mít nainstalovaný nástroj pro automatizovaný překlad Apache Maven (testováno na verzi 2.2.1) a Java Development Kit (testováno na verzi 1.7.0 update 60).

Pro úspěšný překlad programu je třeba vstoupit do adresáře projektu (kde se nachází konfigurační soubor `pom.xml`):

```
malda@asterix:~$ cd CiscoConf/
malda@asterix:~/CiscoConf$ ls -l
-rw-rw-r-- 1 malda malda 2003 čen  8 13:13 nbactions.xml
-rw-r--r-- 1 malda malda 1051 dub 25 11:42 nb-configuration.xml
-rw-rw-r-- 1 malda malda 3473 čen  8 13:12 pom.xml
drwxrwxr-x 4 malda malda 4096 čen 22 13:18 src
```

Vlastní překlad a balíčkování se spustí příkazem `mvn package`:

```
malda@asterix:~/CiscoConf$ mvn package
...
[INFO] Building CiscoConf
...
[INFO] Building zip: /home/malda/CiscoConf/target/CiscoConf.zip
...
[INFO] BUILD SUCCESSFUL
```

Výstupem je balíček `CiscoConf.zip` v adresáři `target`. Balíček obsahuje následující soubory:

- `CiscoConf.jar` – balíček JAR s aplikací
- `run.bat` – skript pro spuštění na platformě Microsoft Windows
- `run.sh` – skript pro spuštění na platformě Linux

Spuštění programu

Pro úspěšné spuštění aplikace je nutné mít nainstalovanou Javu (testováno na verzi 1.7.0 update 60).

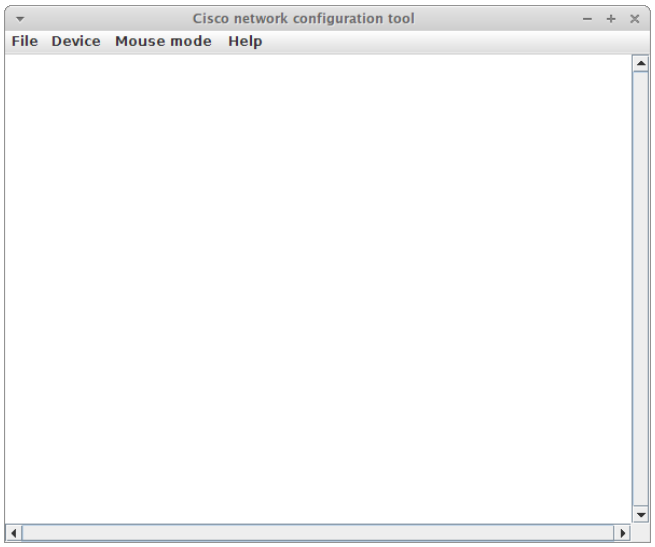
Nejprve je nutné rozbalit distribuční balíček (příkladem je použití nástroje `unzip` na Linuxu, na Windows lze použít jakýkoli ekvivalent):

```
malda@asterix:~/CiscoConf/target$ unzip CiscoConf.zip
Archive:  CiscoConf.zip
   creating: CiscoConf-0.1/
  inflating: CiscoConf-0.1/run.sh
  inflating: CiscoConf-0.1/run.bat
  inflating: CiscoConf-0.1/CiscoConf.jar
```

Program se spustí jedním ze skriptů, podle platformy:

```
malda@asterix:~/CiscoConf/target$ cd CiscoConf-0.1/
malda@asterix:~/CiscoConf/target/CiscoConf-0.1$ ./run.sh
```

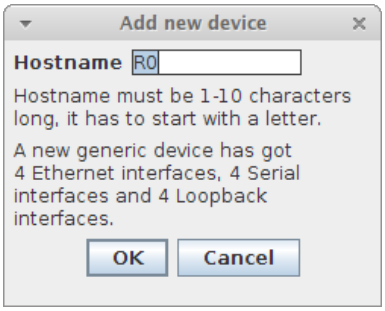
Po spuštění je zobrazeno hlavní okno aplikace s prázdným pracovním prostředím (obrázek C.1).



Obrázek C.1: Uživatelská příručka - hlavní okno programu

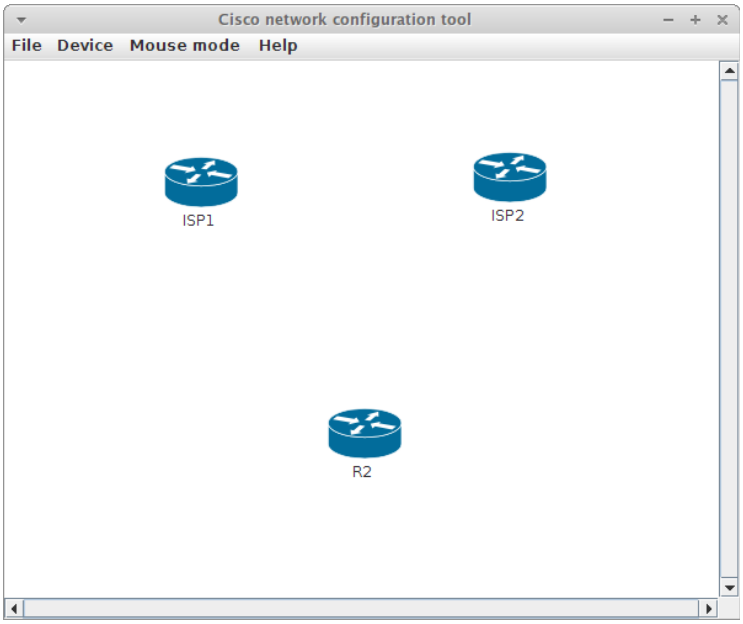
Přidání směrovače

Směrovače se přidávají z hlavního okna aplikace buď zvolením položky v hlavním menu (Device → New router) nebo stisknutím klávesové zkratky `Ctrl+N`. Zobrazí se dialog pro přidání nového zařízení (obrázek C.2).



Obrázek C.2: Uživatelská příručka – přidání nového zařízení

Po vyplnění hostname a potvrzení tlačítkem OK se zařízení přidá do topologie (obrázek C.3). Směrovače je možné tažením myší přemístit na jakékoliv místo v pracovním prostředí.



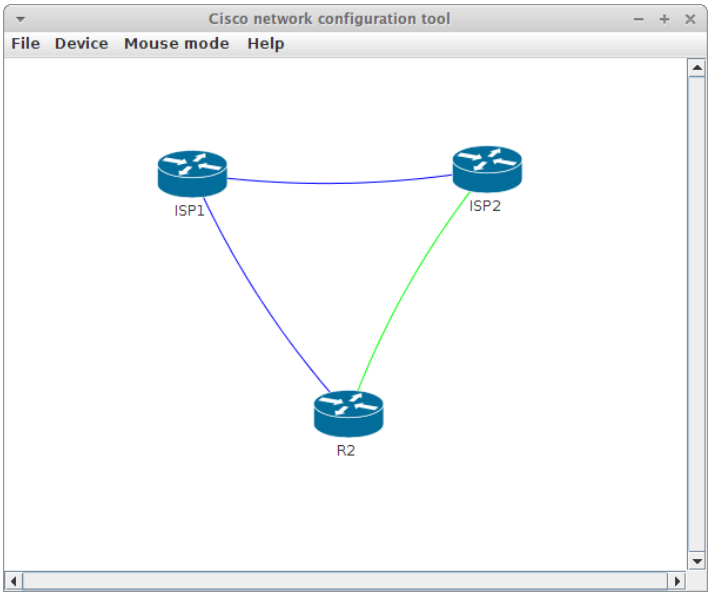
Obrázek C.3: Uživatelská příručka - nová zařízení

Přidání spojení

Spojení se přidá zvolením položky „Add connection“ z kontextového menu zařízení. Kontextové menu zařízení se vyvolá kliknutím pravým tlačítkem myši na zařízení. Zobrazí se dialog pro přidání spojení (obrázek C.4). Nejprve je třeba vybrat zdrojové síťové rozhraní, na základě výběru se přidají možná cílová zařízení (taková, která mají volné síťové rozhraní stejného typu jako je zdrojové). Po výběru cílového zařízení se přidají jeho konkrétní volná síťová zařízení.

Obrázek C.4: Uživatelská příručka - přidání nového spojení

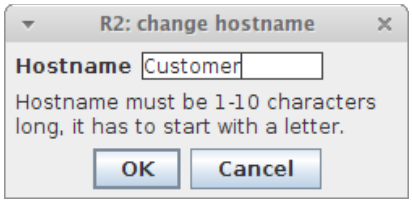
Vložená spojení jsou odlišena barvou dle typu – sériová linka je zobrazena modře, Ethernet zeleně (obrázek C.5).



Obrázek C.5: Uživatelská příručka - nová spojení

Změna hostname zařízení

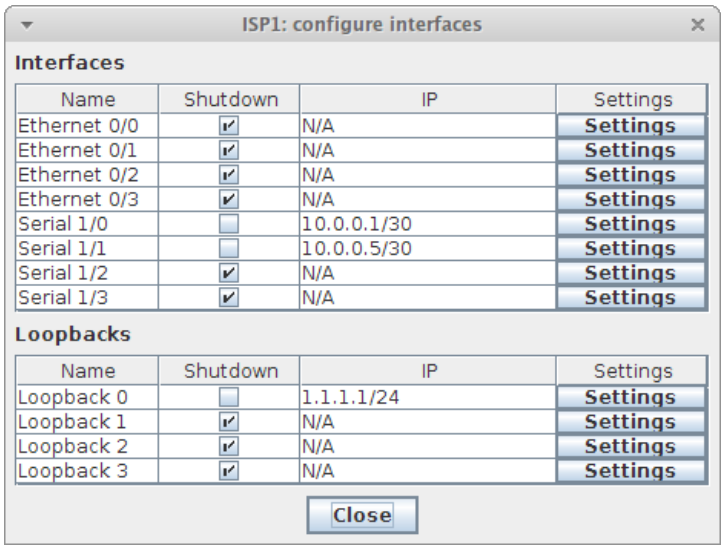
Změnit jméno zařízení lze v dialogu, který se zobrazí po vybrání položky „Change hostname“ z kontextového menu zařízení (obrázek C.6).



Obrázek C.6: Uživatelská příručka - změna hostname

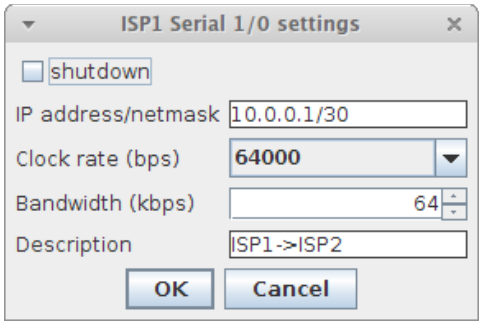
Konfigurace síťových rozhraní

Přehled všech síťových rozhraní zařízení se zobrazí po zvolení položky „Configure interfaces“ z kontextového menu zařízení (obrázek C.7). V tabulkách jsou zobrazeny jak fyzická rozhraní (Serial, Ethernet), tak virtuální rozhraní (Loopback). U každého rozhraní je zobrazen příznak aktivace rozhraní a IP adresa (pokud je nastavena). Podrobná konfigurace je možná po kliknutí na tlačítko „Settings“ u příslušného rozhraní.



Obrázek C.7: Uživatelská příručka - síťová rozhraní

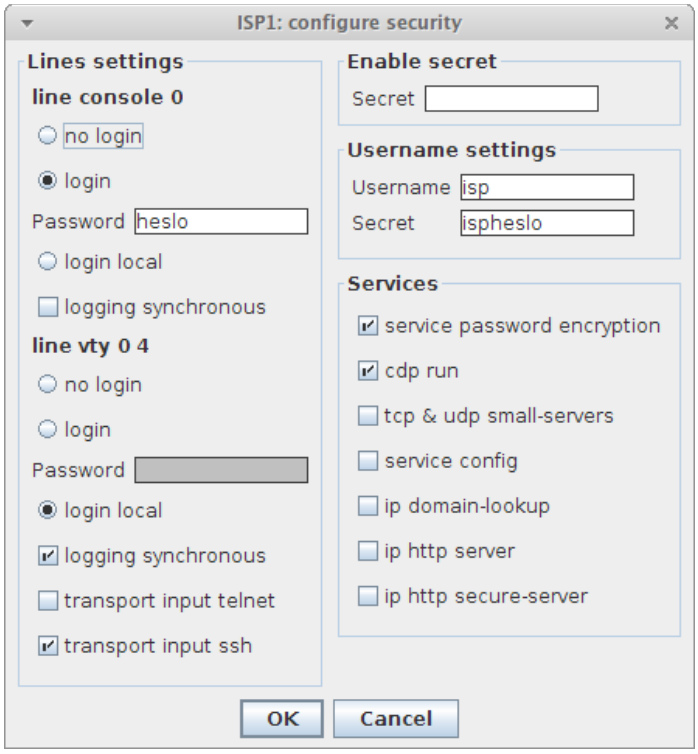
Dialog na obrázku C.8 slouží k nastavení parametrů síťových rozhraní. U všech rozhraní je možné nastavit příznak aktivace, IP adresu a popisek, u sériového rozhraní navíc parametry clock rate a bandwidth.



Obrázek C.8: Uživatelská příručka - nastavení síťového rozhraní

Konfigurace zabezpečení

Konfigurace zabezpečení zařízení se provádí v dialogu zobrazeném po vybrání položky „Configure security“ z kontextového menu zařízení (obrázek C.9).

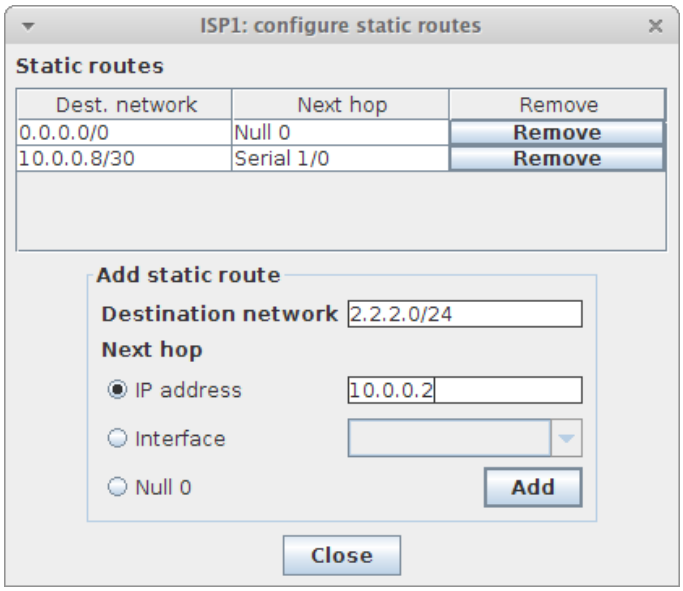


Obrázek C.9: Uživatelská příručka - nastavení zabezpečení

Dialog umožňuje nastavení přístupových kanálů – fyzické konzole (heslo a jeho vyžadování, synchronizace logování) a virtuální konzole (heslo a jeho vyžadování, synchronizace logování, služby povolené pro vzdálený přístup). Lze nastavit heslo do privilegovaného režimu a uživatelský účet. V poslední sekci se nastavují běžící služby.

Konfigurace statického směrování

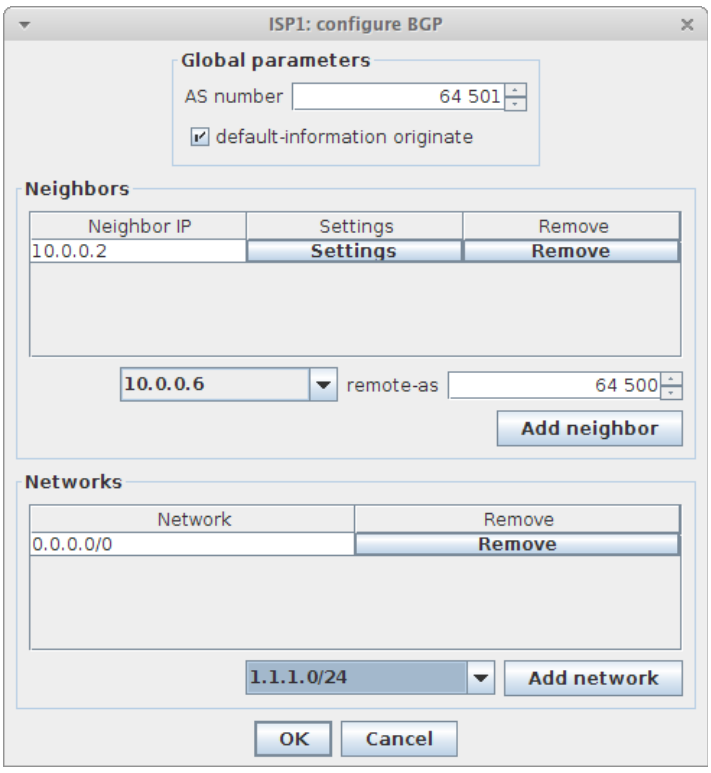
Nastavení statického směrování je možné po zvolení položky „Configure static routes“ z kontextového menu zařízení (obrázek C.10). V tabulce jsou zobrazeny stávající statické trasy, je možné je vymazat tlačítkem „Remove“. Pod tabulkou se nachází formulář pro přidání nové statické trasy. Po zadání adresy cílové sítě se přepínačem zvolí typ následujícího skoku a vyplní jeho hodnota (IP adresa či síťové rozhraní vybrané ze seznamu). Přidání se provede tlačítkem „Add“.



Obrázek C.10: Uživatelská příručka - statické směrování

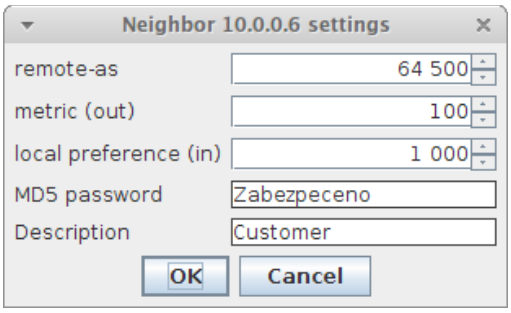
Konfigurace BGP

Dialog pro konfiguraci směrovacího protokolu BGP se zobrazí po výběru položky „Configure BGP“ z kontextového menu zařízení (obrázek C.11). V dialogu se nastavují základní parametry protokolu (lokální ASN a příznak propagace výchozí cesty), BGP sousedi a propagované sítě. Uprostřed dialogu je tabulka s výpisem BGP sousedů. Příslušným tlačítkem lze vyvolat dialog s podrobným nastavením parametrů BGP souseda, či souseda smazat. BGP souseda lze také přidat – v seznamu jsou IP adresy všech přímých sousedů. Je nutné nastavit vzdálené ASN tohoto souseda, přidá se tlačítkem „Add neighbor“. Ve spodní části formuláře jsou v tabulce vypsány propagované sítě. Nově propagovaná síť se přidá zvolením sítě ze seznamu (v seznamu jsou lokální sítě a statické trasy) a stisknutím tlačítka „Add network“.



Obrázek C.11: Uživatelská příručka - BGP

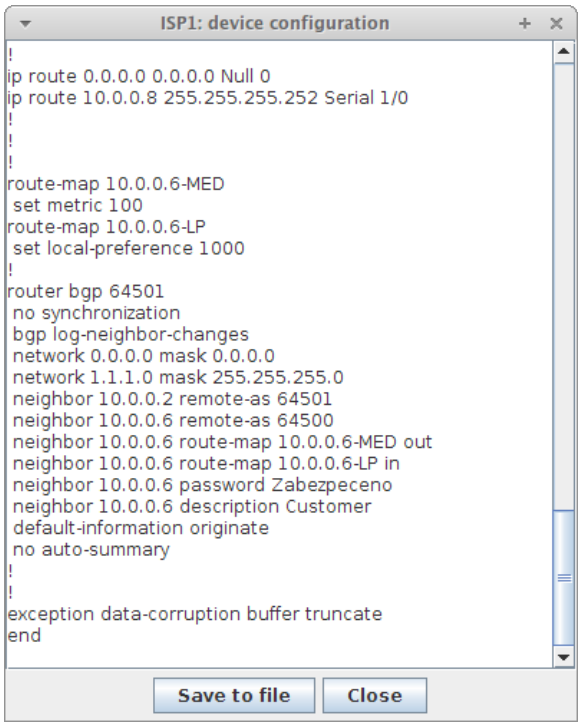
Dialog nastavení BGP souseda (obrázek C.12) umožňuje nastavení vzdáleného ASN, atributu MED (pro všechny cesty odesílané sousedovi), lokální preference (pro všechny cesty přijaté od souseda), zabezpečení BGP relace heslem a popis.



Obrázek C.12: Uživatelská příručka - BGP soused

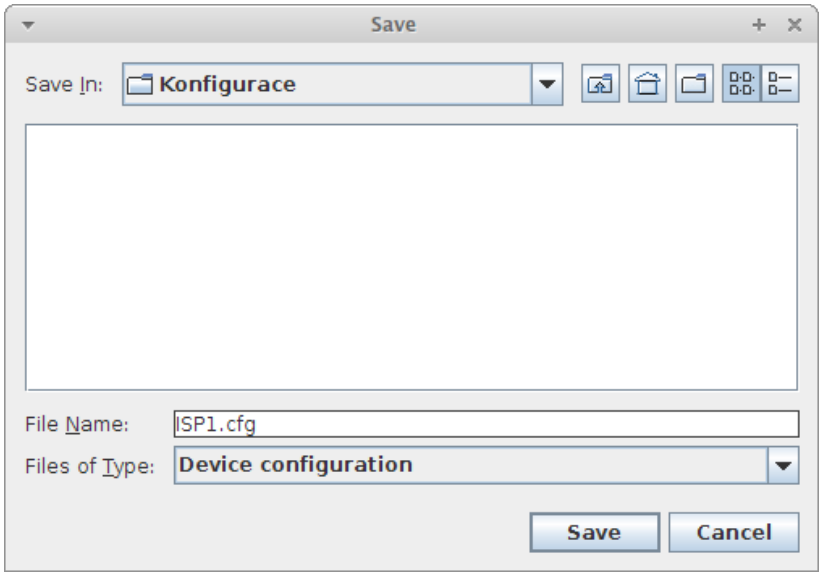
Zobrazení textové konfigurace zařízení

Zobrazit aktuální textovou konfiguraci zařízení je možné vybráním položky „Show device configuration“ kontextového menu zařízení (obrázek C.13).



Obrázek C.13: Uživatelská příručka - zobrazení konfigurace

Konfiguraci lze přímo zkopírovat a vložit do zařízení či uložit do souboru. Uložení do souboru se provádí kliknutím na tlačítko „Save to file“. Zobrazí se standardní formulář pro uložení souboru (obrázek C.14). Výchozí název souboru je nastaven na `hostname.cfg`, název je samozřejmě možné změnit.



Obrázek C.14: Uživatelská příručka - uložení konfigurace

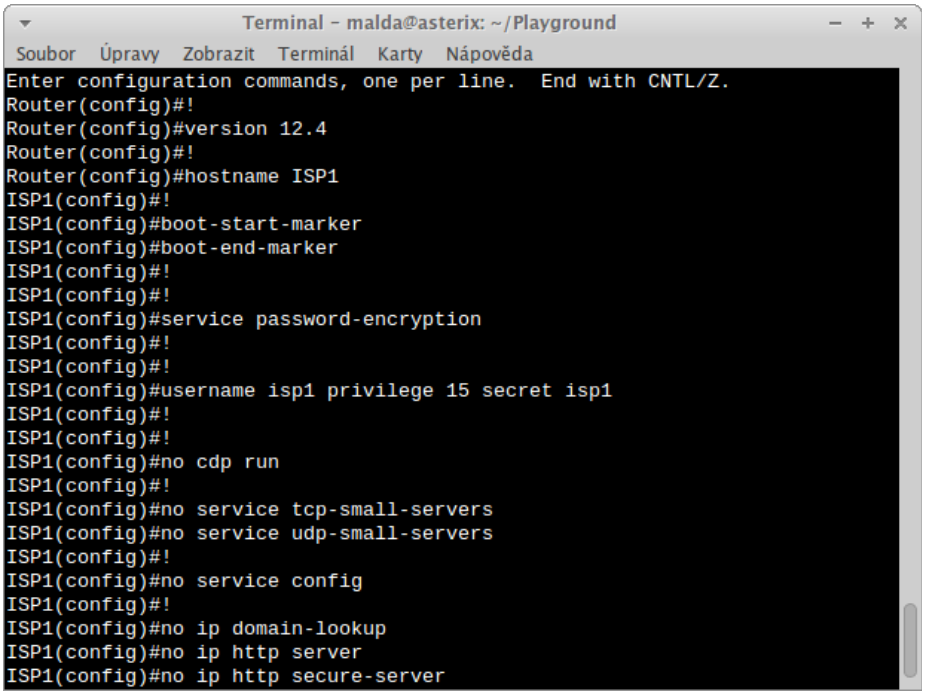
Vložení textové konfigurace do zařízení

Nejrychlejším způsobem, jak vložit textovou konfiguraci do zařízení, je zkopírovat ji přímo do terminálu připojeného na CLI zařízení.

Je nutné zařízení přepnout do globálního konfiguračního módu:

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
```

Poté je možné vložit celou zkopírovanou konfiguraci. Konfigurace se ihned aplikuje, viz obrázek C.15.



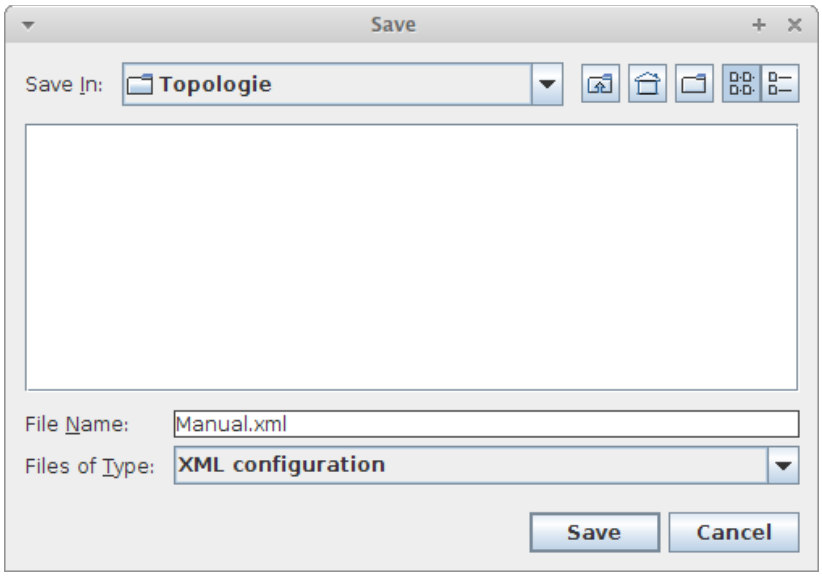
Obrázek C.15: Uživatelská příručka - vložení konfigurace do zařízení

Smazání zařízení/spojení

Zařízení nebo spojení lze smazat výběrem volby „Delete device“ či „Delete connection“ z příslušného kontextového menu. Vždy je nutné smazání potvrdit.

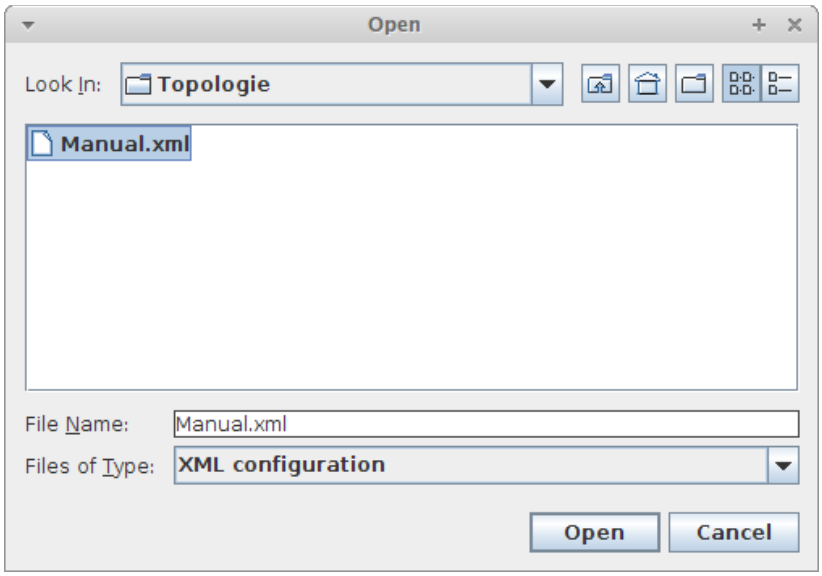
Uložení a načtení topologie

Uložení celé topologie je možné z hlavní obrazovky programu, z menu File → Save topology, nebo použitím klávesové zkratky `Ctrl+S`. Pokud nejsou v topologii žádná vložená zařízení, uložení bude odepřeno. V opačném případě se zobrazí standardní formulář pro uložení souboru s přednastaveným názvem `configuration.xml` (obrázek C.16). Název je samozřejmě možné změnit.



Obrázek C.16: Uživatelská příručka - uložení topologie

Načíst topologii lze volbou položky hlavního menu File → Load topology, nebo klávesovou zkratkou `Ctrl+O`. Pokud existují nějaká zařízení v aktuální topologii, zobrazí se dotaz, vyžadující potvrzení akce – načtením topologie se celá aktuální topologie vymaže. V případě souhlasu se opět zobrazí standardní formulář pro otevření souboru (obrázek C.17).



Obrázek C.17: Uživatelská příručka - načtení topologie

Příloha D: DVD

K diplomové práci je přiloženo DVD obsahující:

- adresář `application` – balík se spustitelnou aplikací
- adresář `examples` – ukázkové topologie a konfigurace z kapitoly 8
  - podadresář `1-security` – kapitola 8.2.1 (Zabezpečení)
  - podadresář `2-single-homed` – kapitola 8.2.2 (Single-homed BGP)
  - podadresář `3-dual-homed` – kapitola 8.2.3 (Dual-homed BGP)
- adresář `project` - Maven projekt (kompletní zdrojové kódy aplikace)
- adresář `text` – text diplomové práce ve formátu PDF a OpenDocument