

# Příkazy CISCO IOS

## Obsah

Úvod.....	5
Vysvětlivky.....	6
Závorky v příkazech.....	6
Lomené závorky < a >.....	6
Složené závorky { a }.....	6
Styl příkazů.....	6
Tučné příkazy (hned pod nadpisem).....	6
Příkazy (netučné).....	6
Znaky a závorky v nadpisech.....	6
Hranaté závorky [ a ].....	6
Pajpa (!) a zarovnání vpravo.....	6
Přehled módů .....	7
Klávesové zkratky .....	8
Uživatelský mód (User Mode).....	9
> enable.....	9
> ping.....	9
> traceroute.....	9
> exit.....	9
> logout.....	9
> show [sh].....	9
> show flash.....	9
> show ip interface.....	9
> show ip route.....	9
> show version.....	10
> show clock.....	10
> show history.....	10
> show interfaces .....	10
> show session.....	10
> telnet.....	10
> disconnect.....	10
> terminal history.....	10
Privilegovaný mód (Privileged Mode).....	12
# disable.....	12
# setup.....	12
# clock set.....	12
# debug.....	12
# debug ip routing.....	12
# debug ip rip.....	12
# debug ip igrp.....	12
# debug all.....	12
# debug ppp negotiation.....	12
# clear.....	12
# clear arp-cache.....	12
# clear ip route.....	13
# configure terminal.....	13
# copy.....	13
# copy flash.....	13

# copy ftp.....	13
# copy runing-config.....	13
# copy startup-config.....	13
# copy tftp.....	13
# reload.....	13
# show.....	14
# show controllers.....	14
# show interfaces.....	14
# show interfaces status.....	14
# show interfaces summary.....	14
# show interfaces trunk.....	14
# show interfaces stats.....	14
# show ip.....	14
# show ip interface.....	14
# show ip route.....	14
#show ip protocols.....	15
#show ip ospf.....	15
# show runing-config [#sh run].....	15
# show cdp.....	15
# show cdp entry.....	15
# show cdp interface.....	15
# show cdp neighbors .....	15
# show processes.....	16
# show protocols.....	16
# show runing-config.....	16
# show startup-config.....	16
# show version.....	16
# show logging.....	16
# show sessions.....	16
# show ssh.....	16
# show users.....	16
# show line.....	16
Konfigurační mód (Configuration mod).....	17
(config)# end.....	17
(config)# banner.....	17
(config)# banner motd.....	17
(config)# banner login.....	17
(config)# hostname.....	17
(config)# line.....	17
(config)# line console.....	17
(config)# line vty.....	17
(config)# enable.....	18
(config)# enable password.....	18
(config)# enable secret.....	18
(config)# logging.....	18
(config)# interface <rozhraní>.....	18
(config)# ip <příkaz>.....	18
(config)# ip route.....	19
(config)# router.....	19
Nastavení interfaců.....	20
(config-if)# description.....	20
(config-if)# ip.....	20

(config-if)# ip address.....	21
(config-if)# ip ospf.....	21
(config-if)#ip ospf authentication.....	21
(config-if)#ip ospf authentication-key.....	21
(config-if)#ip ospf cost.....	21
(config-if)#ip ospf dead-interval.....	22
(config-if)#ip ospf hello-interval.....	22
(config-if)#ip ospf message-digest-key.....	22
(config-if)#ip ospf network.....	22
(config-if)#ip ospf priority.....	22
(config-if)# ipv6.....	23
(config-if)# ipv6 address.....	23
(config-if)# clock rate.....	23
(config-if)# shutdown.....	23
(config-if)# bandwidth.....	23
Nastavení RIP.....	25
(conf-router)# version.....	25
(conf-router)# auto-summary.....	25
(conf-router)# network.....	25
(conf-router)# default-information.....	26
(conf-router)# distance.....	26
(conf-router)# passive-interface.....	26
(conf-router)# redistribute.....	26
(conf-router)# timers.....	26
# debug ip rip.....	27
Nastavení EIGRP (vektor vzdálenosti).....	28
(config-router)# network.....	28
Cena metriky.....	28
(config-if)# bandwidth.....	28
(config-router)# metric.....	28
# show ip protocols.....	29
#show interface.....	29
(config-if)# ip bandwidth-percent eigrp.....	30
Automatická a manuální sumarizace.....	30
(conf-router)# auto-summary   Automatická sumarizace.....	30
(config-if)# ip summary-address eigrp   Ruční sumarizace.....	30
(config-router)# variance.....	31
Autentizace.....	31
(config-if)# ip authentication mode eigrp   Hashování.....	31
(config-if)# ip authentication key-chain eigrp   Zapnutí autentizace.....	31
(config)# key chain   Skupina klíčů.....	31
(config-keychain)# key   ID klíče.....	31
(config-keychain-key)# key-string   Řetězec klíče.....	32
(config-keychain-key)# accept-lifetime starttime.....	32
(config-keychain-key)# send-lifetime starttime.....	32
Výpisy a kontrola stavu provozu.....	32
# show ip eigrp neighbors   Výpis sousedů.....	32
# show ip eigrp interfaces   Informace rozhraní.....	32
# show ip eigrp topology   Tabulka topologie.....	32
# show ip eigrp traffic   Traffic.....	33
# show ip route eigrp   EIGRP routovací tabulka.....	33
Odstraňování závad – Debug.....	33

# debug eigrp fsm.....	33
# debug eigrp packets.....	33
# debug eigrp neighbor.....	33
# debug ip eigrp neighbor.....	33
# debug ip eigrp notifications.....	33
Nastavení OSPF (stav linky).....	34
(config-router)# network.....	34
(config-router)# log-adjacency-changes.....	34
(config)# interface loopback.....	34
(config-if)# ip address.....	34
(config-router)# router-id.....	35
DR a BDR.....	35
Router(config-if)#ip ospf priority .....	35
Cena metriky (cost).....	35
(config-if)# bandwidth.....	35
(config-if)# ip ospf cost.....	35
Autentizace.....	35
(config-router)# area.....	35
(config-if)# ip ospf authentication-key.....	36
(config-if)# ip ospf message-digest-key.....	36
Intervaly.....	36
(config-if)# ip ospf hello-interval.....	36
(config-if)# ip ospf dead-interval.....	36
Propagace implicitní cesty.....	36
(config)# ip route.....	36
(config-router)# default-information originate.....	36
# debug ip ospf events.....	36
# show ip protocols.....	37
(config-router)# area.....	37

## Úvod

**Cisco IOS** (původně Internetwork Operating System) je operační systém používaný na směrovačích a prepínačích firmy Cisco Systems.

Tento materiál postupně zpracovávám v rámci studia CISCO CCNA pro mé budoucí použití jako příručka. Zpřístupňuji jej však i pro ostatní zájemce.

Použil jsem internetové zdroje, které jsou uvedené jako prameny na konci a výpisky z hodin CCNA a curriculli.

## Vysvětlivky

### *Závorky v příkazech*

#### **Lomené závorky < a >**

Povinný/povinné příkaz(y). Musí být zapsán jeden z uvedených příkazů nebo parametrů.

#### **Složené závorky { a }**

Nepovinný/nepovinné příkazy. Příkazy nebo hodnoty, které se mohou zapsat a tím doplnit funkcionalitu.

### *Styl příkazů*

#### **Tučné příkazy (hned pod nadpisem)**

Syntaxe příkazů.

#### **Příkazy (netučné)**

Ukázka příkazů.

### *Znaky a závorky v nadpisech*

#### **Hranaté závorky [ a ]**

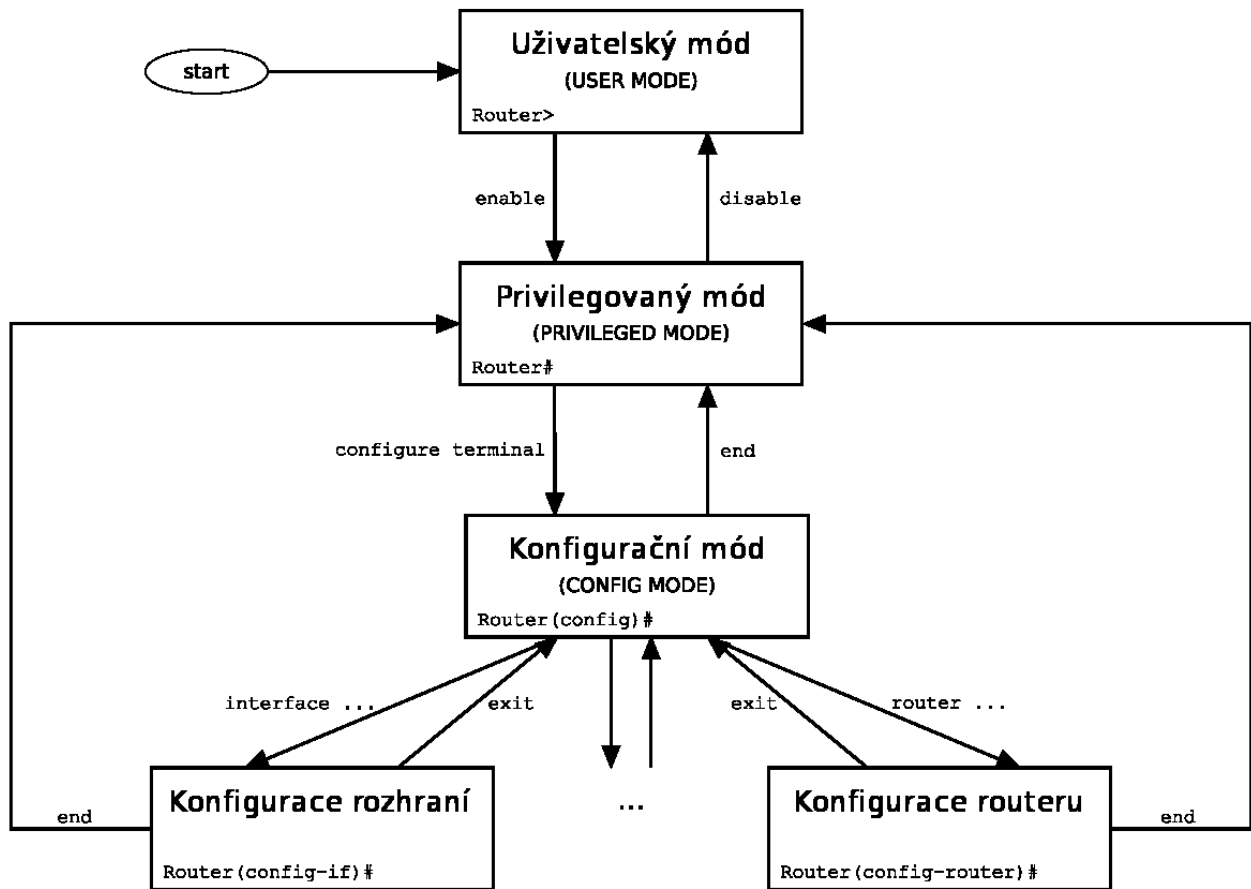
Zkrácený příkaz. „Alternativa“ příkazu ve zkrácené podobě.

#### **Pajpa | a zarovnání vpravo**

Alternativní nadpis příkazu.

## Přehled módů

Základní módy a příkazy, kterými lze mezi těmito módy přecházet:



## Klávesové zkratky

- Šipka nahoru (↑) nebo Ctrl+P - zobrazí předchozí příkaz
- Šipka dolů (↓) nebo Ctrl+N - zobrazí následující příkaz
- Šipka doprava (→) nebo Ctrl+F - posun o jeden znak vpřed
- Šipka doleva (←) nebo Ctrl+B - posun o jeden znak zpět
- Tab - automatické doplnění příkazu
- ? - zobrazí nabídku dostupných příkazů (podle doposud napsaných znaků)
- příkaz „do“ se dá použít před příkazem které jsou použitelné v privilegovaném módu, ale přitom se můžeme nacházet například v konfiguračním módu. Spuštěním příkazu za pomoci „do“ stále zůstaneme v aktuálním módu. Např:

```
(config)# do ping 192.168.1.1
```

- Příkaz „no“ před příkazy zruší daný příkaz. Viz ukázka zrušení ip adresy:  
#no ip address
- Všechny příkazy se dají téměř libovolně zkracovat až na tvar, který musí být pro IOS zřejmý.  
Příklad:

- Následující příkazy:

```
>enable
#config terminal
(config)#interface fastEthernet 0/0
(config-if)#description goR2
(config-if)#ip address 192.168.1.1 255.255.255.252
(config-if)#no shutdown
```

- Lze následně zkrátit:

```
>en
#conf t
(config)#int fa0/0
(config-if)#des goR2
(config-if)#ip add 192.168.1.1 255.255.255.252
(config-if)#no sh
```

- V této příručce se budou používat plné názvy příkazu, ale pokud bude nějaký dlouhý příkaz tak se uvede v hranatých závorkách „[]“ (v nadpisu), která zkratka se běžně používá. V běžném provozu se však používají zkrácené příkazy, protože při konfiguraci ušetří čas.



## Uživatelský mód (User Mode)

Mód, ve kterém se nacházíte po navázání relace s routerem/switchem. Neumožňuje měnit konfiguraci.

### > *enable*

Vstup do privilegovaného módu.

### > *ping*

> **ping** <ip adresa>

>ping 192.168.1.1

Příkaz ping zjišťuje dostupnost cíle pomocí protokolu ICMP.

### > *traceroute*

> **traceroute** <ip adresa>

>traceroute 192.168.1.1

Trasování cesty k cíli pomocí protokolu ICMP

### > *exit*

Ukonči relaci

### > *logout*

Ukonči relaci

### > *show [sh]*

Příkaz zobrazí nejružnější informace

### > *show flash*

Příkaz zobrazí informace o vnitřní paměti FLASH.

### > *show ip interface*

> **show ip interface** {rozhraní}

>show ip interface seriál1/0

Příkaz zobrazí informace o fyzických rozhraních , na kterých je nakonfigurována IP adresa.

### > *show ip route*

> **show ip route** {příkaz}

Příkaz zobrazí routovací tabulku.

Příkazy:

- **ip adresa** síť

#show ip router 192.168.1.0

- **bgp:** ...
- **connected:** ...
- **eigrp:** výpis routovací tabulky protokolu EIGRP
- **ospf:** výpis routovací tabulky protokolu OSPF
- **rip:** výpis routovací tabulky protokolu RIP
- **static:** výpis statické routovací tabulky

**> show version**

Příkaz zobrazí informace o hardwaru, IOS, rozhraních, paměti a konfiguračním registru.

**> show clock**

Příkaz zobrazí nastavený datum a čas.

**> show history**

Příkaz zobrazí seznam naposledy použitých příkazů.

**> show interfaces**

```
> show interfaces {rozhraní}
```

```
>show interfaces eth 0/0/0
```

Příkaz zobrazuje všechny informace o rozhraních.

**> show session**

Zobrazí seznam navázaných spojení.

**> telnet**

```
> telnet <ip address|hostname>
```

```
>telnet 192.168.1.1
```

Telnet klient (pro připojení ke vzdáleným systémům).

**> disconnect**

```
> disconnect <číslo (1 - 16) aktivované sítě >
```

```
>disconnect 1
```

Odpojí navázanou relaci.

**> terminal history**

```
> terminal history <velikost bufferu>
```

```
>terminal history 20
```

Příkaz nastaví velikost bufferu pro uchovávání zadaných příkazů.

## Privilegovaný mód (Privileged Mode)

Mód, ve kterém je možno provádět změnu konfigurace.

### # disable

Návrat do uživatelského módu.

### # setup

Spustí konfigurační utilitu (postupně se formou jednoduchých otázek zeptá na základní konfiguraci).

### # clock set

```
# clock set <hh:mm:ss>
```

```
#clock set 12:35:00
```

Příkaz nastaví čas.

### # debug

Zapne různé ladící výpisy.

#### # debug ip routing

Zapne ladící výpisy pro statické routování

#### # debug ip rip

Zapne ladící výpisy pro routovací protokol RIP.

#### # debug ip igrp

Zapne ladící výpisy pro routovací protokol IGRP.

#### # debug all

Zapne všechny ladící výpisky.

#### # debug ppp negotiation

Zapne ladící výpisy pro ověřování v protokolu PPP.

### # clear

Resetovací funkce

#### # clear arp-cache

Vymaže ARP cache (tabulka obsahující k IP adresám jejich MAC adresy).

#### # clear ip route

```
# clear ip route<* nebo ip adresa>
```

```
#clear ip route 192.168.1.1
```

Vymaže všechny (\*) nebo určitý záznam z routovací tabulky.

### # configure terminal

Vstup do konfiguračního terminálu.

### # copy

Zkopíruje ...

#### # copy flash

```
# copy flash <ftp | tftp | running-config | startup-config>
```

```
#copy flash running-config
```

Zkopíruje z flash do ftp, tftp, running-config nebo startup-config.

**# copy ftp**

```
# copy ftp <flash | running-config | startup-config>
```

```
#copy ftp flash
```

Zkopíruje z FTP do flash, running-config nebo startup-config.

**# copy runing-config**

```
# copy runing-config <flash | ftp | startup-config | tftp>
```

```
#copy runing-config startup-config
```

Zkopíruje z runing-config do flash, ftp, startup-config nebo tftp

**# copy startup-config**

```
# copy startup-config <flash | ftp | runing-config | tftp>
```

```
#copy startu-config flash
```

Zkopíruje z startup-config na flash nebo nahradí aktuální systémovou konfiguraci v ftp, runing-config nebo tftp.

**# copy tftp**

```
# copy tftp <flash | running-config | startup-config>
```

```
#copy tftp: 192.168.1.100 flash
```

Zkopíruje z TFTP do flash, running-config nebo startup-config.

**# reload**

Restartuje IOS

**# show**

Zobrazí informace:

**# show controllers**

```
# show controllers {rozhraní}
```

```
#show controllers seriál 0/0/0
```

Show controllers je vhodné použít při hledání chyb, když fyzická kontrola spojení není možná. Příkaz zobrazí typ kabelu připojeného u kontroléru, což je velmi účinný nástroj pro nalezení špatných typů kabelů v propojení, vadných kabelů nebo nalezení sériových rozhraní bez kabelů.

**# show interfaces**

```
# show interfaces {rozhraní}
```

```
#show interfaces ethernet 0/0
```

Detailní výpis o jednom/všech interfacích.

**# show interfaces status**

Vypsání informací fyzických interfaců s popisem a stavem v přehledné formě.

**# show interfaces summary**

Stručný seznam všech interfaců se statistikou přenosů.

**# show interfaces trunk**

Informace o existujících troncích.

**# show interfaces stats**

Seznam statistik pro interfacý.

**# show ip**

```
# show ip <další příkaz(y)>
```

```
#show ip route
```

Zobrazení informací, které souvisí s IP.

### **# show ip interface**

```
# show ip interface {interface apod.}
```

```
#show ip interface brief
```

```
#show ip interface seriál 0/0
```

Zobrazení informací o ACL a routování na interface .

### **# show ip route**

```
# show ip route {bližší specifikování}
```

```
#show ip route rip
```

Zobrazení směrovací tabulky

### **#show ip protocols**

```
# show ip protocols
```

```
#show ip protocols
```

Vypíše informace o použitém routovacím protokolu a statistiku (např.: administrativní vzdálenost).

### **#show ip ospf**

```
#show ip ospf {bližší specifikování}
```

```
#show ip ospf neighbor
```

Zobrazení informací ohledně OSPF protokolu.

Bližší specifikování mohou být:

- **<1 – 65535>**: ID procesu
- **border-routers**: Informace o koncových směrovačích
- **database**: Shrnutí databáze.
- **interface**: Informace o interfacích (typ sítě, časovače, cena linky, vznik sousedství v příslušném směru apod.).
- **neighbor**: Seznam sousedů a informace o nich.
- **virtual-links**: Informace o virtuálních linkách.

### **# show runing-config [#sh run]**

Výpis kompletní konfigurace.

### **# show cdp**

```
# show cdp {entry | interfaces | neighbors}
```

```
#show cdp neighbors detail
```

Zobrazí informace o CDP.

### **# show cdp entry**

```
# show cdp entry <* | word ...>
```

```
#show cdp entry *
```

Zobrazí informace o daném sousedovi.

### **# show cdp interface**

```
# show cdp interface {rozhraní}
```

```
#show cdp interface ethernet 0/0/1
```

Zobrazí informace o rozhraních na kterých běží CDP.

### **# show cdp neighbors**

**# show cdp neighbors {detail}**

#show cdp neighbors

Cdp neighbours poskytuje informace o zařízeních, které jsou přímí „sousedé“ za pomoci CDP protokolu. Výpis zobrazuje informace o aktivních zařízeních, ID portu, MAC adresu, IP adresu a název zařízení. Příkaz slouží ke zjištění funkčních spojení mezi vedlejšími zařízeními.

### **# show processes**

Informace o využití procesoru a běžících procesech.

### **# show protocols**

Zobrazí, které protokoly jsou aktuálně použité (nakonfigurované).

### **# show running-config**

Vypsání běžící konfigurace.

### **# show startup-config**

Vypsání startovací konfigurace.

### **# show version**

Informace o zařízení a verzi IOS.

### **# show logging**

Informace o loggování a poslední záznamy.

### **# show sessions**

Informace o aktuálních telnetových spojeních.

### **# show ssh**

Informace o aktuálních SSH spojeních.

### **# show users**

Informace o přihlášených uživateli.

### **# show line**

Informace o linkách.

## Konfigurační mód (Configuration mod)

Konfigurace zařízení.

**(config)# end**

Návrat do privilegovaného módu.

**(config)# banner**

Popisky

**(config)# banner motd**

**(config)# banner motd <informace>**

(config)#banner motd %Hello World%

Nastavení popisu/baneru/informace zařízení (switch, router) začínat a končit řetězec musí stejným jakýmkoliv znakem, který se nesmí objevit uprostřed daného řetězce.

**(config)# banner login**

**(config)# banner login <informace>**

(config)#banner login %Cekame na vase prihlaseni%

Nastavení popisu/baneru/informace zařízení při loginu (řetězec viz banner motd).

**(config)# hostname**

**(config)# hostname <informace>**

(config)#hostname R1

Popisek zařízení pro lepší přehlednost. Popisek typu „Router(config)#“ se změní na „R1(config)#“.

**(config)# line**

Nastavuje způsob přihlašování ke konzoli a virtuálním terminálům.

**(config)# line console**

**(config)# line console 0**

**(config-line)# password <heslo>**

**(config-line)# login**

(config)#line console 0

(config-line)#password vrz123buch

(config-line)#login

???. U switchů se login nepoužívá, u routerů je to nutné!

**(config)# line vty**

**(config)# line vty <id prvního spojení> <id posledního spojení>**

**(config-line)# password <heslo>**

**(config-line)# login**

(config)#line vty 0 3

(config-line)#password vrz123buch

(config-line)#login

Telnetové připojení k zařízení. Telnetové připojení není dost bezpečné. V tomto případě je nastavené heslo „vrz123buch“ pro 4 spojení s ID 0 až 3 (tzv virtuální terminály).

**(config)# enable**

Nastavení zabezpečení (hesla) do privilegovaného módu (enable) – bude požadováno heslo.

**(config)# enable password**

**(config)# enable password <heslo>**

```
(config)#enable password vrz123buch
```

Nastavení hesla do privilegovaného módu.

### **(config)# enable secret**

```
(config)# enable secret <heslo>
```

```
(config)#enable secret vrz123buch
```

Nastavení hesla do privilegovaného módu. Heslo na rozdíl od předchozí verze uloží zašifrovaně.

### **(config)# logging**

```
(config)# logging <Ip adresa | buffered | console | host | on | userinfo  
| synchronous>
```

```
(config)#logging console warnings
```

```
(config)#logging console
```

```
(config)#no logging console
```

Výpis debug informací do terminálu.

### **(config)# interface <rozhraní>**

```
(config)# interface <rozhraní>
```

```
(config)#interface ethernet 0/0/0
```

```
(config-if)#ip address 192.168.1.1 255.255.255.0
```

```
(config-if)#description R2
```

```
(config-if)#no shutdown
```

Zvolení rozhraní, které se dále bude konfigurovat.

### **(config)# ip <příkaz>**

```
(config)# ip <příkaz>
```

```
(config)#ip route 10.0.0.0 255.255.255.0 192.168.1.1
```

Příkazy:

- **access-list:**
- **default-network:**
- **dhcp:** Nastavení DHCP serveru
- **domain:**
- **domain-lookup:**
- **domain-name:** Nastavení defaultního domain-name
- **forward-protocol:**
- **ftp:** Nastavení FTP
- **host:**
- **inspect:**
- **ips:** Systémy pro prevenci průniku
- **local:** Zadání lokálních nastavení
- **name-server:** Zadání IP adresy DNS serveru, který se má použít.
- **nat:** Konfigurační příkazy NAT
- **route:** Nastavit statických spojení



- **ssh**: Konfigurace SSH
- **tcp**: Globální TCP parametry

### **(config)# ip route**

```
(config)# ip route <cílový subnet> <maska> <ip/typ rozhrani>
```

```
(config)#ip route 10.0.0.0 255.255.255.0 serial0/0
```

```
(config)#ip route 0.0.0.0 0.0.0.0 75.34.21.10
```

Vytvoření statických spojení – routovacích záznamů.

### **(config)# router**

```
(config)# router <rip | bgp | eigrp | ospf>
```

```
(config)#router rip
```

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

```
(config-router)#network 10.0.0.0
```

Nastavení automatického vytvoření routovací tabulky za pomoci různých protokolů.

## Nastavení interfaců

Konfigurace interfaců. Do nastavení interfaců se dá dostat následovně:

```
> enable
# config terminal
(config)# interface <interface>
(config)#interface serial 0/0
```

Interfacy mohou být tyto:

- Ethernet
- FastEthernet
- GigabitEthernet
- Serial
- Loopback (virtuální interface)
- Tunnel
- Vlan
- Virtual-Template
- Range
- Dot11Radio

### **(config-if)# description**

```
(config-if)# description <popisek>
(config-if)#description R2
```

Popisek daného interfacu.

### **(config-if)# ip**

```
(config-if)# ip <příkaz>
(config-if)# ip address 10.0.1.1 255.255.255.0
```

Příkazy:

- address
- acces-group
- hello-interval
- helper-address
- inspect
- ips
- mtu
- nat
- ospf
- split-horizon

- summary-address
- virtual-reassembly

### **(config-if)# ip address**

```
(config-if)# ip address <ip interfacu> <maska>
```

```
(config-if)# ip address 10.0.1.1 255.255.255.252
```

Přidání IP adresy k danému interfacu.

### **(config-if)# ip ospf**

```
(config-if)# ip ospf <příkaz>
```

```
(config-if)# ip ospf cost 390
```

Přidání parametrů OSPF k danému interfacu.

Příkazy:

- authentication
- authentication-key
- cost
- dead-interval
- hello-interval
- message-digest-key
- network
- priority

### **(config-if)#ip ospf authentication**

```
(config-if)# ip ospf authentication <>
```

```
(config-if)# ip ospf authentication ...
```

XXX

### **(config-if)#ip ospf authentication-key**

```
(config-if)# ip ospf authentication-key <password>
```

```
(config-if)# ip ospf authentication-key cisco
```

Nastavení klíče (hesla). Maximální délka hesla je 8 znaků. Toto heslo musí mít všechny sousedící routery stejné, jinak si nebudou moci vyměňovat OSPF informace.

### **(config-if)#ip ospf cost**

```
(config-if)# ip ospf cost <1 - 65535>
```

```
(config-if)# ip ospf cost 390
```

Konfigurace metriky přímo nastavením ceny (cost). Cena linky je určena vydělením referenční šířky pásma šířkou pásma tohoto rozhraní.

Šířka pásma je číslo 1 až 10000000 [KB]. Cena je číslo 1 až 65535.

### **(config-if)#ip ospf dead-interval**

```
(config-if)# ip ospf dead-interval <1 - 65535>
```

```
(config-if)# ip ospf dead-interval 40
```

Zbývající čas (v sekundách), který bude směrovač čekat na přijetí kontaktního hello packetu od souseda (neighbor) před tím než se prohlásí sousedství za zrušené/mrtvé (dead). Většinou 4x delší

než hello-interval. Musí být stejný jako dead interval u sousedů jinak se nestanou sousedy.

### **(config-if)#ip ospf hello-interval**

```
(config-if)# ip ospf hello-interval <1 - 65535>
```

```
(config-if)# ip ospf hello-interval 160
```

Interval kdy se má posílat Hello packet sousedovi. Musí být stejný jako hello interval u sousedů jinak se nestanou sousedy.

### **(config-if)#ip ospf message-digest-key**

```
(config-if)# ip ospf message-digest-key <ID klíče> md5 <password>
```

```
(config-if)# ip ospf message-digest-key 1 md5 cisco
```

Nastavení identifikačního klíče pro OSPF.

**ID klíče** je identifikátor klíče a může nabývat 1 – 255.

**Heslo** může nabývat maximálně 16 znaků, které bude následně zahashované (MD5).

ID klíče i heslo musí být na sousedících routerech stejné jinak si nebudou vyměňovat OSPF informace.

Heslo bude v konfiguraci (NVRAM) routeru uloženo jako čistý text (plain text) pokud se nepoužije příkaz:

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

### **(config-if)#ip ospf network**

```
(config-if)# ip ospf network <broadcast | point-to-point>
```

```
(config-if)# ip ospf network broadcast
```

??????????????????.

### **(config-if)#ip ospf priority**

```
(config-if)# ip ospf priority <0 - 255>
```

```
(config-if)# ip ospf priority 1
```

Nastavení priority rozhraní.

Priorita 0 – router je nezpůsobilý stát se pověřeným směrovačem (DR) ani záložní (BDR). Čím vyšší číslo tím vyšší priorita a možnost, aby se router stal pověřeným routerem (s nejvyšší prioritou) nebo záložním (druhá nejvyšší priorita). Jsou-li dvě (nebo více) nejvyšší priority se stejnou hodnotou tak se následně pověřuje podle nejvyššího ID routeru.

### **(config-if)# ipv6**

```
(config-if)# ipv6 <příkaz>
```

```
(config-if)#ipv6 address 2002:C0A8:2101::1/128
```

Příkazy:

- address
- dhcp
- eigrp
- enable
- hello-interval
- mtu
- nat

- nd
- ospf
- rip
- summary-address
- traffic-filter

### **(config-if)# ipv6 address**

```
(config-if)# ipv6 address <ipv6>
```

```
(config-if)#ipv6 address 2002:C0A8:2101::1/128
```

Přidání IPv6 k danému interfacu.

### **(config-if)# clock rate**

```
(config-if)# clock rate <frekvence>
```

```
(config-if)#clock rate 64000
```

Nastavení časování na interfacu s označením DCE

### **(config-if)# shutdown**

```
(config-if)# shutdown
```

```
(config-if)#shutdown
```

Vypnutí daného interfacu.

Pro opětovné zapnutí (nebo první zapnutí) slouží příkaz:

```
(config-if)#no shutdown
```

### **(config-if)# bandwidth**

```
(config-if)# bandwidth <1 - 10000000>
```

```
(config-if)#bandwidth 64
```

Konfigurace metriky pomocí šířky pásma (rychlost/šířka pásma) v kilobitech. Používá se například u protokolu OSPF.

## Nastavení RIP

Info:

- IGP (Interior Gateway Protocol) – **Vnitřní protokol**
- DV (Distance Vektor) – **Vektor vzdálenosti** – (typ algoritmu)
- Classful – **Třídní** (RIPv1); Classless – **Beztrídní** (RIPv2)
- AD (Administrative Distance) – Administrativní vzdálenost – **120**
- Kód – **R**

Spuštění konfigurace protokolu RIP se provede následovně:

```
> enable
# config terminal
(conf)# router rip

(conf-router)# version
(conf-router)# version <1 | 2>
(conf-router)#version 2
```

Nastavení verze protokolu RIP (verze 1 se již téměř vůbec nepoužívá).

Při použití následujícího příkazu:

```
(conf-router)#no version 2
```

Přepne zpět do verze 1, ale přijímat bude aktualizace jak verze 1 tak verze 2. Vysílat však bude jen ve verzi 1.

```
(conf-router)# auto-summary
(conf-router)# auto-summary
```

Použití u RIP verze 2. Zapne se funkce automatické sumarizace, které je stejné jak u RIP verze 1. Sumarizace na hranici plné třídy. Tato funkce se ve verzi 2 nachází díky zpětné kompatibilitě k verzi 1.

U RIP verze 2 se z pravidla používá spíše následující příkaz, který vypne sumarizaci v rámci plné třídy tzv. classless (beztrídní):

```
(conf-router)#no auto-summary
```

```
(conf-router)# network
(conf-router)# network <subnet>
```

```
(conf-router)#network 192.168.0.0
```

Zpřístupní směrování na ip 192.168.0.0 a jím přiřazeným hostům. V daném případě budou zpřístupněny IP adresy od 192.168.0.0 až po 192.168.255.255.

```
(conf-router)#network 10.0.0.0
```

Nyní bude rozsah zpřístupnění pro routování od 10.0.0.0 až po 10.255.255.255.

```
(conf-router)#network 10.0.1.0
```

Nyní bude rozsah zpřístupnění pro routování od 10.0.1.0 až po 10.0.1.255 u RIP verze 2 jinak opět 10.0.0.0 až po 10.255.255.255.

**Nejvhodnější** je použít IP adresy sítě jaké jsou uvedené v routovací tabulce za pomoci příkazu (RIP

verze 2):

```
(conf-router)#do show ip route
```

Nyní stačí jen zapsat ip adresy které jsou s **příznakem C (conected)**. Viz obrázek:

```

172.31.0.0/30 is subnetted, 3 subnets
C      172.31.7.16 is directly connected, Serial0/1/1
C      172.31.7.20 is directly connected, Serial0/1/0
C      172.31.7.32 is directly connected, FastEthernet0/0
R0(conf-router)#network

```

**Tip:** IP adresa se nemusí ručně opisovat stačí jen zatahnout a **kliknout prostředním tlačítkem** – fungční **jen GNU Linux**.

### Ukázka rozdílu RIP verze 1 a 2 v příkazu network:

Máme podsít' (subnet) 10.0.1.0. Ve verzi 2 se zapíše:

```
(conf-router)#network 10.0.1.0
```

U verze 1 se musí zapsat však celá třída:

```
(conf-router)#network 10.0.0.0
```

Samozřejmě pokud se u verze 1 zadá nějaká podsít' jako ve verzi 2, tak se to zkonvertuje na třídní IP adresu (10.0.1.0 → 10.0.0.0)

### **(conf-router)# default-information**

```
(conf-router)# default-information originate
```

```
????????????????????
```

### **(conf-router)# distance**

```
(conf-router)# distance <1 - 255>
```

```
(conf-router)# distance 3
```

Nastavení administrativní vzdálenosti.

### **(conf-router)# passive-interface**

```
(conf-router)# passive-interface <interface>
```

```
(conf-router)#passive-interface FastEthernet 0/0/1
```

Rozhraní na kterém se má potlačit vysílání update. Naslouchat však stále bude.

### **(conf-router)# redistribute**

```
(conf-router)# redistribute <směrovací protokol>
```

```
(conf-router)#redistribute ospf 1
```

```
????????
```

### **(conf-router)# timers**

```
(conf-router)# timers basic <Interval 1> <Interval 2> <Interval 3>
<Interval 4>
```

```
(conf-router)#timers basic 30 180 180 240
```

- **Interval 1 (update):** interval mezi updaty.
- **Interval 2 (invalid):** tento interval slouží, jak dlouho se má čekat při neobdržení směrovací tabulky před tím, než se označí za neplatnou.
- **Interval 3 (holddown):** pokud tento interval vyprší je daná cesta označena jako neplatná.

- **Interval 4 (flush):** pokud tento interval vyprší je uchovaná neplatná cesta vymazána.

### **# debug ip rip**

**# debug ip rip**

(conf-router)#do debug ip rip

Zapnutí vypisování aktivit RIP v reálném čase.

**Pozor:** používat pouze při ladění, protože se jinak zbytečně zatěžuje router.

Vypnout se dá následovně:

#undebug all

vypne veškeré ladění. Nebo použít příkaz:

#undebug ip rip

pro vypnutí ladění RIP.



## Nastavení EIGRP (vektor vzdálenosti)

Info:

- IGP (Interior Gateway Protocol) – **Vnitřní protokol**
- DV (Distance Vektor) – **Vektor vzdálenosti** – (typ algoritmu)
- Classless – **Beztrždní**
- AD (Administrative Distance) – Administrativní vzdálenost – **90**
- Kód – **D**

EIGRP (Enhanced Interior Gateway Routing Protocol) je směrovací protokol typu vektor vzdálenosti (vylepšený), beztrždní směrovací protokol. EIGRP, je vylepšení protokolu Cisco IGRP (Interior Gateway Routing Protocol).

EIGRP obsahuje několik funkcí, které se běžně nevyskytují v jiných směrovacích protokolech typu vektor vzdálenosti jako jsou RIP nebo IGRP. Mezi ně patří:

- spolehlivý transportní (L4) protokol RTP (Reliable Transport Protocol),
- omezené aktualizace,
- konvergentní algoritmus DUAL (Diffusing Update Algorithm),
- vytváření vztahů sousedství (adjacencies),
- tabulky Sousedů (neighbor) a Topologickou (topology).

Spuštění konfigurace protokolu OSPF se provede následovně:

```
> enable
# config terminal
(conf)# router eigrp <číslo AS>
```

Číslo AS (Autonomous system – autonomní systém) může nabývat hodnot 1 až 65535. Číslo AS odlišuje jeden AS od jiného AS v síti – Odlišné AS primárně mezi sebou nekomunikují pokud se nepoužije redistribuce. AS je v podstatě i zároveň id procesu EIGRP

Vypnout směrovací protokol EIGRP (pro daný AS) se dá následovně:

```
(conf)# no router eigrpv <číslo AS>
```

### **(config-router)# network**

```
(config-router)# network <subnet> {pseudo maska}
(config-router)# network 192.168.1.0 0.0.0.255
```

Pokud se **neuvede** inverzní maska (pseudo/wildcard mask), tak se bere classfull mask.

Pseudo maska je vlastně invertovaná neboli negovaná maska. V tomto případě je rozsah od 192.168.1.0 až 192.168.1.255. Pokud bude pseudo maska například 0.0.0.3 tak je rozsah od 192.168.1.0 až 192.168.1.3. Skutečně to může být jinak. Když se zadá například IP adresa 10.0.0.0 a wildcard 255.255.255.255 tak se budou brát všechny podsítě i s jejich maskami jako je například 10.0.1.0/27, 10.0.2./24, atd. Ale je to méně škálovatelné než když se zadají všechny podsítě ručně!

### **Cena metriky**

#### **(config-if)# bandwidth**

```
(config-if)# bandwidth <1 - 10000000>
(config-if)# bandwidth 64
```

Konfigurace **ceny metriky (cost)** pomocí šířky pásma (rychlost/šířka pásma) v kilobitech. Slouží k výpočtu metriky.

### **(config-router)# metric**

```
(config-router)# metric weights <tos> <k1> <k2> <k3> <k4> <k5>
```

```
(config-router)#metric weights 0 1 0 1 0 0
```

Nastavení metriky:

- tos (type of service) může nabývat hodnot od 0 do 8. Defaultně je nastavena 0.  
***Pozn.:** tos je odkaz na původní protokol IGRP, zamýšlející směrování podle typu služby. Protože to ale nebylo nikdy zavedeno do praxe, je pole tos v tomto příkaze vždy nastaveno na 0.*
- K1 (bandwidth) může nabývat hodnot od 0 do 256. Defaultně je nastavena 1.
- K2 (load) může nabývat hodnot od 0 do 256. Defaultně je nastavena 0.
- K3 (delay) může nabývat hodnot od 0 do 256. Defaultně je nastavena 1.
- K4 (reliability) může nabývat hodnot od 0 do 256. Defaultně je nastavena 0.
- K5 (reliability) může nabývat hodnot od 0 do 256. Defaultně je nastavena 0.

S implicitním nastavením je metrika EIGRP redukována na nejpomalejší šířku pásma plus součet všech zpoždění odchozích rozhraní z lokálního směrovače do cílové sítě.

Aby mohly dva směrovače zformovat vztah sousedství v EIGRP, musí jim vzájemně souhlasit hodnoty K1 až K5.

**Vzorce pro výpočet metriky:**

- Defaultní metrika (implicitní formule):  
 $K1 * bandwidth + K3 * delay$
- Kompletní metrika (kompletní formule):  
$$(K1 * bandwidth + \frac{K2 * bandwidth}{256 - load} + K3 * delay) * \frac{K5}{reliability + K4}$$

Při výpočtu metriky se používají následující hodnoty:

- bandwidth (referenční šířka pásma):  
$$256 * \frac{10\,000\,000}{\text{nejnižší šířka pásma na trase do cíle}}$$
- delay (zpoždění):  
$$\frac{256 * (\text{součet zpoždění na cestě do cíle})}{10}$$

Nejlepší cesta (s nejmenší metrikou, feasible distance) je ta s největší šířkou pásma a s nejmenším zpožděním.

### **# show ip protocols**

```
#show ip protocols
```

Pro zobrazení aktuální (případně defaultní) nastavené metriky (i jiných informací) K1 až K5. Metrika se nachází (u EIGRP) na řádku: **EIGRP metric weight**.

### **#show interface**

```
#show interface {interface}
```

```
#show interface ethernet 0/0
```

Získání aktuálních hodnot vah metriky na daném rozhraní:

- **BW** (bandwidth/přenosová rychlost): je zobrazená v Kb (Kbit)
- **DLY** (delay/zpoždění): doba potřebná pro cestu paketu k cíli v  $\mu$ s (usec – mikro sekundách).
- **rely** (reliability/spolehlivost): měřítko pravděpodobností, že linka selže respektive jak často se vyskytují chyby na lince (1 [1/255] – nespolehlivá linka, 255 [255/255] – spolehlivá linka) během aktuálních posledních 5 minut.
- **load** (load/zatížení): aktuální využití (zatížení) linky (0 [0/255] – nevyužívaná linka, 1 [1/255] – minimálně využívaná linka, 255 [255/255] – plně zatížená linka).

### **(config-if)# ip bandwidth-percent eigrp**

```
(config-if)# ip bandwidth-percent eigrp <AS> <procento>
```

```
(config-if)#ip bandwidth-percent eigrp 1 150
```

Tento příkaz slouží k nastavení skutečného rozsahu přenosové kapacity (bandwidth - šířky pásma), které může být EIGRP použito na daném rozhraní pro výměnu směrovacích informací.

**AS** – Autonomní systém

**procento** – jaké procento (z příkazu „(config-if)#bandwidth“ ) se má použít k nastavení skutečného rozsahu bandwidth.

Procenta mohou být i převyšující čísla 100 (100%) z důvodu nastavení nižšího bandwidth než je reálná (resp teoretická) hodnota. Defaultně je u EIGRP nastaveno na 50%. Takto nastavený bandwidth (na nižší hodnotu) je užitečné pro určité práce s metrikou (například: aby linka byla úmyslně použita jako záložní a přitom chceme použít její plnou přenosovou kapacitu).

### **Automatická a manuální sumarizace**

#### **(conf-router)# auto-summary**

```
(conf-router)# auto-summary
```

#### **| Automatická sumarizace**

Zapne se funkce automatické sumarizace. Sumarizace na hranici plné třídy (classful).

U EIGRP se z pravidla používá spíše následující příkaz, který vypne sumarizaci v rámci plné třídy tzv. classless (beztřídní):

```
(conf-router)#no auto-summary
```

Od verze IOS 12.2 je automatická sumarizace implicitně vypnutá.

Špatně navržená síť s nesouvislými podsítěmi může mít problémy s konektivitou, jestliže je funkce automatické sumarizace zapnutá. Například: jestliže by dva směrovače inzerovaly stejnou síť 172.16.0.0/16, když by ve skutečnosti bylo třeba, aby inzerovaly dvě různé sítě 172.16.10.0/24 a 172.16.20.0/24. Pro tyto případy se používá manuální sumarizace.

#### **(config-if)# ip summary-address eigrp**

#### **| Ruční sumarizace**

```
(config-if)# ip summary-address eigrp <číslo AS> <subnet> <maska> {AD}
```

```
(config-if)#ip summary-address eigrp 1 10.10.0.0 255.255.0.0 75
```

Zapnutí manuální sumarizace na konkrétním rozhraní (interface)

- **číslo AS**: číslo autonomního systému.
- **subnet**: podsíť, která se má sumarizovat.
- **maska**: maska sumarizované sítě.
- **AD**: administrativní vzdálenost (Administrative Distance) nabývá hodnot od 1 do 255.

Defaultní hodnota je 5.

### **(config-router)# variance**

```
(config-router)# variance <1 - 128>
```

```
(config-router)#variance 50
```

Příkaz variance (variance) slouží k vyvažování zátěže - nestejněmorné vyvažování zátěže (unequal cost path load balancing). Standardně se zvolí cesta s nejnižší metrikou (FD), pokud je však těchto cest více (mají stejné FD), tak se použije stejnoměrné vyvažování zátěže. Pokud se použije tato funkce, tak se použijí všechny cesty, které mají metriku menší nebo rovnou  $\text{variance} * \text{FD}$  tudíž platí tento vztah:  $\text{metrika} \leq \text{variance} * \text{FD}$ .

Použijí se samozřejmě cesty, které jsou FS (feasible successor) – splňující podmínku  $\text{RD} < \text{FD}$  (RD - Inzerovaná vzdálenost [Reported Distance]; FD - Přípustná vzdálenost [Feasible Distance]). To znamená, že cesta od souseda do cíle je menší než nejkratší cesta.

### **Autentizace**

??? V tomto se moc nevyznám – neměl jsem možnost vyzkoušet (packettracer nepodporuje)!!!

#### **(config-if)# ip authentication mode eigrp**

**| Hashování**

```
(config-if)# ip authentication mode eigrp <AS> md5
```

```
(config-if)#ip authentication mode eigrp 1 md5
```

Tento příkaz zapne MD5 hashovací funkci k autentizaci EIGRP paketů na daném rozhraní pro daný AS (autonomní systém).

#### **(config-if)# ip authentication key-chain eigrp**

**| Zapnutí autentizace**

```
(config-if)# ip authentication key-chain eigrp <AS> <název skupiny>
```

```
(config-if)#ip authentication key-chain eigrp 1 michalangelo
```

Tento příkaz zapne autentizaci EIGRP paketů na daném rozhraní pro daný AS (autonomní systém) za pomoci skupiny klíčů.

**název skupiny** – skupina klíčů (key chain): jméno skupiny klíčů, které se mají použít k autentizaci.

#### **(config)# key chain**

**| Skupina klíčů**

```
(config)# key chain <název skupiny>
```

```
(config)#key chain michalangelo
```

Určuje pojmenovanou skupinu klíčů (key chain). Název skupiny musí souhlasit s názvem, které je uvedeno příkazem „ip authentication key-chain eigrp“ na rozhraní.

#### **(config-keychain)# key**

**| ID klíče**

```
(config-keychain)# key <ID klíče>
```

```
(config-keychain)#key 1
```

Určení čísla klíče (ID klíče). Rozsah klíčů je od 1 do 2147483647. Identifikační čísla klíčů nemusí být na sebe navazující. V řetězci musí být definován nejméně jeden klíč.

#### **(config-keychain-key)# key-string**

**| Řetězec klíče**

```
(config-keychain-key)# key-string <řetězec>
```

```
(config-keychain-key)#key-string tajneheslo
```

Uřčení hesla (řetězce) klíče. Řetězec může obsahovat od 1 do 80 alfanumerických znaků (malá i velká písmena), s výjimkou prvního znaku, který nemůže být číslice.

#### **(config-keychain-key)# accept-lifetime starttime**

```
(config-keychain-key)# accept-lifetime start-time {infinite | end-time | duration | seconds}
```

```
(config-keychain-key)#accept-lifetime ???
???
```

**(config-keychain-key)# send-lifetime starttime**

```
(config-keychain-key)# accept-lifetime start-time {infinite | end-time |
duration | seconds}
```

```
(config-keychain-key)#accept-lifetime ???
???
```

## Výpisy a kontrola stavu provozu

### # show ip eigrp neighbors

| Výpis sousedů

```
# show ip eigrp neighbors {detail | AS}
```

```
#show ip eigrp neighbors detail
```

Vypíše tabulku sousedů. Pokud se zapíše i **detail**, tak se zobrazí detailní tabulka, kde se ověřuje zda je soused nastaven jako hraniční směrovač (stub router). Nebo zobrazí jen pro určitý AS (autonomní systém).

### # show ip eigrp interfaces

| Informace rozhraní

```
# show ip eigrp interfaces {rozhraní | AS}
```

```
#show ip eigrp neighbors serial 0/0
```

Vypíše seznam informací o každém rozhraní (bez parametru), určitém **rozhraní** (parametr např.: seriál 0/0), nebo AS (číslo autonomního systému, např.: 1).

### # show ip eigrp topology

| Tabulka topologie

```
# show ip eigrp topology {IP | AS | all-links}
```

```
#show ip eigrp topology
```

```
#show ip eigrp topology all-links
```

Vypíše tabulku topologie – kde se nacházejí přípustní následníci (feasible successors).

**IP – s**

**AS – s**

**all-links** – Zobrazí tabulku topologie včetně cest, které nesplňují podmínku přípustnosti (feasibility condition). Zobrazuje všechny možné cesty do cílové sítě.

### # show ip eigrp traffic

| Traffic

```
# show ip eigrp traffic {AS}
```

```
#show ip eigrp traffic
```

Vypíše počet a typ vyslaných a přijatých paketů. Případně jen pro určitý AS (autonomní systém).

### # show ip route eigrp

| EIGRP routovací tabulka

```
# show ip route eigrp
```

Vypíše seznam směrovací tabulky pouze se záznamy protokolu EIGRP.

## Odstraňování závad – Debug

### # debug eigrp fsm

```
# debug eigrp fsm
```

Živý výpis událostí a akcí prováděné algoritmem DUAL související s EIGRP metrikou přípustných následníků – FSM (Feasible Successor Metrics).

### # debug eigrp packets

```
# debug eigrp packets
```

Živý výpis události a akce související s EIGRP pakety. Typy informací o packetu: UPDATE, REQUEST, QUERY, REPLY, HELLO a ACK.

### **# debug eigrp neighbor**

```
# debug eigrp neighbor
```

Živý výpis událostí související s EIGRP sousedy.

### **# debug ip eigrp neighbor**

```
# debug ip eigrp neighbor
```

Živý výpis událostí související s EIGRP sousedy pro protokol IP.

### **# debug ip eigrp notifications**

```
# debug ip eigrp notifications
```

Živý výpis oznámení události EIGRP.

## Nastavení OSPF (stav linky)

Info:

- IGP (Interior Gateway Protocol) – **Vnitřní protokol**
- LS (Link State) – **Stav linky** – (typ algoritmu)
- Classless – **Beztrídni**
- AD (Administrative Distance) – Administrativní vzdálenost – **110**
- Kód – **O**

Spuštění konfigurace protokolu OSPF se provede následovně:

```
> enable
# config terminal
(conf)# router ospf <ID procesu>
```

Identifikace procesu může nabývat hodnot 1 až 65535. Identifikátor procesu pouze odlišuje jeden proces od jiného na jednom zařízení (nesouvisí s OSPF oblastí – OSPF area).

Některé příkazy jsou uvedené i v nastavení **interface** v části **(config-if)# ip ospf**.

### **(config-router)# network**

```
(config-router)# network <subnet> <pseudo maska> area <0 - 255>
(config-router)#network 192.168.1.0 0.0.0.255 area 0
```

IP adresa podsítě (subnet) s inverzní maskou (pseudo/wildcard mask).

Pseudo maska je vlastně invertovaná neboli negovaná maska. V tomto případě je rozsah od 192.168.1.0 až 192.168.1.255. Pokud bude pseudo maska například 0.0.0.3 tak je rozsah od 192.168.1.0 až 192.168.1.3. Skutečně to může být jinak. Když se zadá například IP adresa 10.0.0.0 a wildcard 255.255.255.255 tak se budou brát všechny podsítě i s jejími maskami jako je například 10.0.1.0/27, 10.0.2./24, atd. Ale je to méně škálovatelné než když se zadají všechny podsítě ručně!

Area je identifikace oblasti OSPF.

### **(config-router)# log-adjacency-changes**

```
(config-router)# log-adjacency-changes {detail}
(config-router)#log-adjacency-changes detail
```

Nastaví směrovač tak, aby posílal systémové logovací zprávy (syslog message), když nastane změna stavu mezi OSPF sousedy. Defaultně je zapnuto:

```
(config-router)#log-adjacency-changes
```

bez příkazu „detail“. Tudíž vypisuje jen událost zapnuto/vypnuto.

### **(config)# interface loopback**

```
(config)# interface loopback <0 - 2147483647>
(config)#interface loopback 0
```

Vytvoří se virtuální rozhraní.

### **(config-if)# ip address**

```
(config-if)# ip address <IP> <Maska>
(config-if)#ip address 192.168.100.1 255.255.255.255
```

Přiřazení právě jedné IP adresy (IP adresa sítě) do virtuálního rozhraní.

Identifikace routeru (OSPF router ID) tak zvaná zpětná smyčka.

### **(config-router)# router-id**

**(config-router)# router-id <IP>**

```
(config-router)#router-id 10.1.1.1
```

Nastavení identifikátoru routeru (Router ID).

Pokud je již proces aktivní, tak nový identifikátor routeru bude použit až po restartu OSPF nebo restartu systému (IOS).

Statický identifikátor směrovače lze smazat následovně:

```
(config-router)#no router-id 10.1.1.1
```

Pokud opět je proces aktivní, tak předchozí nastavení ID routeru bude použito po restartu IOS nebo OSPF.

Restart (reload) IOS se provede příkazem:

```
(config-router)#do reload
```

### **DR a BDR**

**DR (Designated Router)** pověřený router. **BDR (Backup Designated Router)** záložní pověřený router. Např v místech kde je více routeru propojeno switchem (router → router to nemá smysl).

### **Router(config-if)#ip ospf priority**

Viz Nastavení interface → (config-if)# ip ospf priority.

### **Cena metriky (cost)**

#### **(config-if)# bandwidth**

**(config-if)# bandwidth <1 - 10000000>**

```
(config-if)#bandwidth 64
```

Konfigurace **ceny metriky (cost)** pomocí šířky pásma (rychlost/šířka pásma) v kilobitech.

Alternativou je příkaz (např):

```
Router(config-if)#ip ospf cost 1564
```

### **(config-if)# ip ospf cost**

Viz Nastavení interface → (config-if)# ip ospf cost.

### **Autentizace**

#### **(config-router)# area**

**(config-router)# area <oblast> authentication {message-digest}**

```
(config-router)#area 0 authentication
```

Bez příkazu „message-digest“ je tzv. jednoduchá autentizace – nešifrovaná – heslo bude posíláno nešifrovaně (plain text).

S příkazem „message-digest“ je šifrovaná autentizace kdy bude heslo zahashovano pomocí MD5.

### **(config-if)# ip ospf authentication-key**

Jednoduchá autentizace.

Viz Nastavení interface → (config-if)# ip ospf authentication-key.

### **(config-if)# ip ospf message-digest-key**

Šifrovaná autentizace.

Viz Nastavení interface → (config-if)# ip ospf message-digest-key.



## Intervaly

### (config-if)# ip ospf hello-interval

Viz Nastavení interface → (config-if)# ip ospf hello-interval.

### (config-if)# ip ospf dead-interval

Viz Nastavení interface → (config-if)# ip ospf dead-interval.

## Propagace implicitní cesty

### (config)# ip route

```
(config)# ip route 0.0.0.0 0.0.0.0 <interface | IP>
```

```
(config)# ip route 0.0.0.0 0.0.0.0 serial 0/1/1
```

Nastaví se statický routovací záznam s IP 0.0.0.0 a maskou 0.0.0.0 pro daný interface.

Následně se musí nastavit: „(config-router)#default-information originate“ viz dále.

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

```
(config-router)# default-information originate {always}
```

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

**Bez příkazu always:** Nastaví, aby byla implicitní cesta propagována na všechny routery OSPF.

**S příkazem always:** Nastaví, aby byla implicitní cesta propagována na všechny routery OSPF i když není žádná nastavená (některé IOS nemusí obsahovat always – např. Packettracer 5.3.2.0027).

Obvykle nastaveno na vstupním nebo bránovém (gateway) routeru (ASBR – Autonomous System Boundary Router).

## # debug ip ospf events

```
# debug ip ospf <adj | events>
```

```
#(config-router)#do debug ip ospf events
```

Zapnutí vypisování aktivit OSPF v reálném čase.

**Pozor:** používat pouze při ladění, protože se jinak zbytečně zatěžuje router.

Vypnout se dá následovně:

```
#undebug all
```

vypne veškeré ladění. Nebo použít příkaz:

```
#undebug ip ospf events
```

pro vypnutí ladění OSPF events.

## # show ip protocols

Tímto výpisem se dá zjistit administrativní vzdálenost (Distance)

### (config-router)# area

```
(config-router)# area <oblast> <příkaz>
```

```
(config-router)#area 0 authentication
```

**Oblast** nabývá hodnot 0 – 4294967295 nebo ID oblasti ve formátu IP adresy.

**Příkazy** mohou být:

- Authentication: zpřístupní jednoduchou autentizaci (heslo bude posíláno v čistém textu)
- Default cost: ???
- Nssa: specifikuje NSSA oblast

- Stub: specifikuje stub oblast
- Virtual-link: definuje virtuální linky a jejich parametry

Využito na zpracování:

**Seminární práce do předmětu CC3**

Hledání chyb na směrovačích

Autoři:

Pavel Pop (xpoppa00)

Petr Jurča (xjurca04)

Datum obhajoby: 9.4.2007

[http://cs.wikibooks.org/wiki/P%C5%99%C3%ADkazy\\_Cisco\\_IOS](http://cs.wikibooks.org/wiki/P%C5%99%C3%ADkazy_Cisco_IOS)

<http://www.samuraj-cz.com>