



EVROPSKÁ UNIE  
Evropské strukturální a investiční fondy  
Operační program Výzkum, vývoj a vzdělávání



# **Komunikační síť II**

## **(2. část)**

### **Autoři textu:**

Ing. Libor Michalek, Ph.D.

Ing. Petr Machník, Ph.D.

**Ostrava 2020**

# Obsah

1.	Adresování v IPv4 a IPv6.....	3
1.1	IPv4 adresa .....	3
1.2	Třídy IPv4 adres .....	3
1.3	Maska sítě v IPv4 .....	4
1.4	Privátní sítě.....	5
1.5	Automatická konfigurace v IPv4.....	5
1.6	IPv6 adresa .....	6
1.7	Automatická konfigurace v IPv6.....	6
2.	Subnetting .....	8
2.1	VLSM (Variable Length Subnet Mask) .....	8
2.2	Podpora síťových prvků.....	12
3.	Ethernet.....	13
3.1	Ethernet rámec.....	13
3.2	Typy Ethernetu .....	13
3.2.1	Fast Ethernet .....	13
3.2.2	Gigabit Ethernet (1GE) .....	14
3.2.3	10 Gigabit Ethernet (10 GE).....	14
3.2.4	40 a 100 Gigabit Ethernet (40 / 100 GE).....	15
3.2.5	200 a 400 Gigabit Ethernet, Terabit Ethernet .....	15
4.	VLAN .....	16
4.1	Standard IEEE 802.1q .....	16
4.2	Protokol VTP .....	17
4.2.1	Prořezávání (trunking) protokolu VTP .....	17
4.3	Směrování mezi VLAN .....	18
4.4	Protokol Spanning Tree .....	19
5.	Paketové filtry .....	23
5.1	Základní princip filtrace .....	23
5.2	Stavové paketové filtry.....	23
6.	NAT .....	26

# 1. Adresování v IPv4 a IPv6

## 1.1 IPv4 adresa

IPv4 je datově orientovaný protokol, který je používán v sítích s přepojováním paketů (např. Ethernet). Jde o protokol přepravující data bez záruky, tj. negarantuje ani doručení ani zachování pořadí ani vyloučení duplicit. Zajištění těchto záruk je ponecháno na vyšší vrstvě, kterou představuje protokol TCP. Stejně tak je na vyšší vrstvě ponechána kontrola integrity dat, protože IPv4 datagram nese pouze informaci o kontrolním součtu hlavičky datagramu se služebními údaji.

Adresa slouží k identifikaci rozhraní počítače. Má-li počítač více síťových karet, musí mít každá karta vlastní adresu. IP adresy se nacházejí na 3. vrstvě modelu ISO/OSI a jejich primárním cílem je zavést hierarchickou strukturu sítě. IPv4 poskytuje omezený adresní prostor – teoreticky  $2^{32}$  adres (cca  $4 \times 10^9 = 4$  miliardy adres). Příklad struktury IPv4 adresy je uveden na obr. 1.1, z něhož je patrné, že adresa se skládá ze dvou částí a to identifikátoru sítě a identifikátoru stanice. [1,2,3]



Obr.1.1 IPv4 adresa

## 1.2 Třídy IPv4 adres

Původní návrh IPv4 předpokládal rozdělení adresy na síťovou a lokální část fixní, prvních osm bitů adresy určovalo síť, zbytek pak stroj v síti. Síťi tudíž mohlo být nejvýše 256 (v každé však mohlo být přes 16 milionů stanic), s nástupem lokálních sítí tento systém přestal být použitelný. Přijaté řešení spočívalo v zavedení tříd adres, třídy A pro malý počet velkých sítí, třídy B pro střední počet středních sítí a třídy C pro velký počet malých sítí. Dále byla definována třída D pro skupinové vysílání (multicasting) a třída E zůstala jako rezerva.

Úspěšnému žadateli (což byla instituce či firma) o adresu sítě se přidělovala vždy adresa sítě požadované třídy, takže měl pro sebe celý adresní prostor v dané podsíti. Tento mechanismus hospodaření s adresním prostorem se označuje jako třídní (classful).

V tab. 1.1. je uvedeno rozdělení tříd IP adres.

Tab. 1.1 Třídy IP adres

Třída	začátek (bin)	1. bajt	standardní maska	bitů sítě	bitů stanice	sítí	stanic v každé síti
A	0	0–127	255.0.0.0	8	24	$2^7 = 128$	$2^{24} - 2 = 16\,777\,214$
B	10	128–191	255.255.0.0	16	16	$2^{14} = 16384$	$2^{16} - 2 = 65\,534$
C	110	192–223	255.255.255.0	24	8	$2^{21} = 2\,097\,152$	$2^8 - 2 = 254$
D	1110	224–239	<i>multicast</i>				
E	1111	240–255	<i>vyhrazeno jako rezerva</i>				

V každé třídě lze obecně určit počet sítí

$$N_{sítí} = 2^n$$

kde  $n$  je počet bitů identifikátoru sítě.

Rovněž lze v každé třídě obecně určit počet stanic jako

$$N_{stanic} = 2^n - 2$$

V případě počtu stanic je nutno odečíst dvojku proto, že první adresa z každého síťového rozsahu určuje adresu sítě a poslední adresa každého rozsahu tzv. broadcast neboli všesměrovou adresu. Adresu sítě a broadcast nelze použít pro adresaci stanice, a tak je nutné tyto dvě adresy od celkového počtu odečíst. Je-li broadcast použit jako cílová adresa (jako zdrojová adresa použít nelze), je paket doručen všem stanicím v dané síti vymezené maskou sítě.

### 1.3 Maska sítě v IPv4

Maska sítě je číslo, které udává, jak velká je daná síť. Číselná reprezentace udává, kolik bitů zleva má hodnotu 1, např. pro masku 8 to bude 11111111 a zbytek samé nuly. V dekadickém zápisu má maska 8 podobu 255.0.0.0. Jak velká síť bude, lze vypočítat velice jednoduše. Maska určuje, které bity se nesmějí měnit pro danou síť, ty jsou označeny jedničkami, bity, které nesou hodnotu 0, je možné v rámci sítě měnit. Pokud má tedy maska hodnotu 24, tedy maska obsahuje 24 jedniček, pak z IP adresy zbylo pouhých 8 bitů, které lze měnit. Počet adres v dané síti je tedy  $2^{32-24} = 2^8 = 256$ .

Vynásobí-li se binárně bit po bitu adresa stanice s maskou sítě, získá se adresa sítě, viz. tab. 1.2. Převede-li se výsledek logického součinu do dekadického tvaru, získá se hledaná adresa sítě: 192.168.1.0.

Tab. 1.2 Výpočet adresy sítě zdrojové stanice

binárně				
Adresa zdroje	11000000	10101000	00000001	00111000
Maska sítě	11111111	11111111	11111111	00000000
Adresa sítě	11000000	10101000	00000001	00000000

dekadicky				
Adresa zdroje	192	168	1	135
Maska sítě	255	255	255	0
Adresa sítě	192	168	1	0

Identickým postupem zpracovávají směrovače přijaté požadavky o přeposlání paketu na konkrétní cílovou adresu. Pokud má směrovač ve své směrovací tabulce uloženu informaci o cílové síti, do které by měl být paket směrován, provede binární násobení cílové adresy s příslušnou maskou sítě, zkontroluje adresu sítě a podle ní vybere rozhraní, kterým bude paket do cílové sítě přeposlán.

## 1.4 Privátní sítě

Protokol IP je velmi populární, a proto je nasazován i do sítí, které nebyly nebo neměly být připojeny k Internetu. Privátní adresy jsou běžně používány pro domácí, kancelářské a podnikové lokální sítě (LAN), kde veřejné adresy (tj. globálně směrovatelné v Internetu) nejsou žádoucí nebo nejsou dostupné. Privátní adresy jsou označovány jako soukromé, protože nejsou globálně delegované, což znamená, že nejsou přiděleny žádné konkrétní organizaci a jimi adresované IP pakety nemohou být přenášeny přes veřejný internet. Kdokoliv může používat tyto adresy bez schválení od regionálního internetového registru. Pokud takováto privátní síť potřebuje připojení k Internetu, musí používat buď překlad síťových adres (NAT), nebo proxy server.

Pro tyto sítě byly vyhrazeny 3 bloky tzv. privátních adres dle RFC 1918, viz. tab. 1.3

Tab. 1.3 Rozsahy privátních IP adres

Označení RFC 1918	Rozsah IP adres	Počet adres
24bitový blok	10.0.0.0 – 10.255.255.255	16 777 216
20bitový blok	172.16.0.0 – 172.31.255.255	1 048 576
16bitový blok	192.168.0.0 – 192.168.255.255	65 536

## 1.5 Automatická konfigurace v IPv4

DHCP protokol umožňuje prostřednictvím DHCP serveru nastavovat automaticky stanicím v počítačové síti sadu parametrů nutných pro komunikaci pomocí IP protokolu (tj. využívat rodinu protokolů TCP/IP). Typicky DHCP server přiděluje počítačům pomocí DHCP protokolu IP adresu, masku sítě, implicitní bránu a adresu DNS serveru.

## 1.6 IPv6 adresa

IPv6 (internetový protokol verze 6) je v označení nastupujícího protokolu pro komunikaci v současném Internetu (resp. v počítačových sítích, které Internet vytvářejí). IPv6 nahrazuje dosluhující protokol IPv4. Přináší zejména masivní rozšíření adresního prostoru (tj. možnost přidělit všem zařízením jejich vlastní IPv6 adresu) a zdokonalení schopnosti přenášet vysokorychlostně data.

Hlavní změna, kterou přináší IPv6, je daleko větší adresní prostor, což umožňuje větší pružnost při přidělování adres. Velký adresní prostor IPv6 obsahuje celkem  $2^{128}$  (zhruba  $3,4 \times 10^{38}$ ) adres. Je znemožněno použití překladu síťových adres (NAT), který byl zaveden kvůli vyčerpání adresního prostoru IPv4 a kvůli bezpečnosti.

Adresy IPv6 mají 128 bitů, tedy 16 bajtů. Zapisujeme je na rozdíl od IPv4 v hexadecimálním tvaru, vždy po dvojicích bajtů oddělených dvojtečkami, například **FEDC:02A5:0000:002A:00C0:7600:0C12:1C47**

Nevýznamné (úvodní) nuly každé čtveřice můžeme z výše uvedeného zápisu vypustit a adresu zapsat jako **FEDC:2A5:0:2A:C0:7600:C12:1C47**. Jelikož často bývá delší část adresy nulová, můžeme více následných nulových bitů vyjádřit symbolem "::".

Například adresu **FEDC:02A5:0000:0000:00C0:7600:0C12:1C47** můžeme zkrátit na **FEDC:2A5::C0:7600:C12:1C47**. Použili jsme zde navíc možnost vypustit nevýznamné úvodní nuly. Zápis "::" se může vyskytovat i na začátku nebo konci adresy, např. **::FE12:CCD0** nebo **DC80:FD87:A800::**. Z důvodu jednoznačnosti se zápis "::" může v adrese objevit pouze jednou. Např. adresu **FD08:0000:0000:DAC8:0000:0000:0000:DACD** můžeme zapsat jako **FD08::DAC8:0000:0000:0000:DACD** nebo **FD08:0000:0000:DAC8::DACD**. Kdybychom použili (nesprávného) zápisu **FD08::DAC8::DACD**, nebyla by adresa jednoznačná, jelikož by ze zápisu nebylo zřejmé, kolik nulových pozic reprezentuje první a kolik druhý výskyt symbolu "::".

Často potřebujeme vyjádřit také prefix adresy. Počet bitů tvořících prefix zapisujeme za lomítko za hodnotu prefixu, zapsanou podle konvencí zápisu adres IPv6. Jedná se o stejný zápis, na jaký jsme již zvyklí z beztrždního adresování u IPv4 (CIDR). Například 60-bitový prefix zapíšeme jako **AAAA:BBBB:CCCC:DDD0::/60**. [1,2,3]

## 1.7 Automatická konfigurace v IPv6

IPv6 nabízí dvě možnosti – tzv. stavovou a beze-stavovou automatickou konfiguraci. Stavová konfigurace předpokládá zprovoznění speciálního serveru, který parametry připojení na žádost přiděluje. Jde o stejný princip jako u známého protokolu DHCP (Dynamic Host Configuration Protocol, RFC2131), hojně používaného v IPv4. Protokol DHCPv6, určený pro přidělování parametrů v IPv6 má oproti klasickému DHCP některá rozšíření, avšak princip zůstává stejný – stanice nejprve s použitím skupinové adresy vyhledá DHCP server, který ji nabídne na určitou dobu pronájem jisté síťové adresy a spolu s ní další parametry, jako

výchozí bránu a adresu DNS serveru. Stanice si jednu z nabídek vybere a s DHCP serverem, který tuto adresu nabídnul, si požadavek na přidělení vzájemně potvrdí.

U beze-stavové konfigurace se nevyžaduje DHCP server, stanice si svou adresu vytvoří ze své MAC adresy a prefixu lokální sítě, kterou periodicky ohlašuje směrovač na všech rozhraních svých LAN. K tomu se používá zpráva protokolu ICMPv6 Router Advertisement. Směrovač pak rovněž stanici poslouží jako výchozí brána.

## 2. Subnetting

Beztrždní směrování CIDR (*Classless Interdomain Routing*) vzniklo jako řešení při nedostatku veřejných IPv4 adres, ke kterému došlo s masivním rozšiřováním Internetu po celém světě. Beztrždní směrování využívá efektivněji adresního prostoru. Při běžném směrování (Classful Routing) je vždy pro konkrétní třídu IP adresy přiřazena síťová maska, kterou nelze změnit. Příkladem může být IP adresa 10.10.10.1, kde je pevně přiřazena síťová maska 255.0.0.0. Pevně daná délka síťové masky ovšem není ideální a hlavně není efektivní. V určitých případech může být adresní prostor zbytečně rozsáhlý (plýtvání IP adresami) a v jiných naopak nedostatečný (málo IP adres).

Proto vyšla v roce 1993 doporučení RFC 1517 až 1520, která strategii dělení adresního prostoru radikálně upravila. Přestalo se na síť nahlížet „trždně“ a začala se výhradně používat síťová maska. Aby nedošlo k nejednoznačnosti, je třeba oprostit se od trždního náhledu na síť a vždy k adrese doplnit i příslušnou masku sítě. Tato metoda rozdělení sítě na menší části se nazývá podsítování (subnetting). Maska podsítě má tak více jedniček, než maska standardní pro danou třídu. Jedničky lze ale z masky rovněž odebírat (nahrazovat zprava nulami), a vytvářet tak síť větší tzv. supersítě (supernetting). Maska podsítě má potom jedniček méně, než standardní maska třídy. [1,2,3]

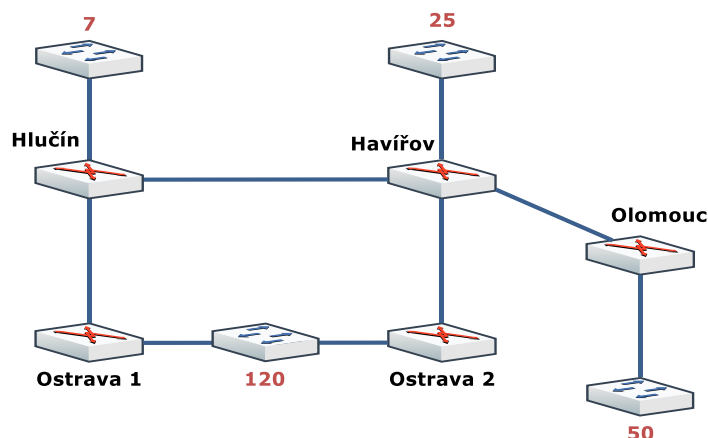
Protože je dekadický zápis velmi dlouhý, zavedl se zjednodušený zápis masky sítě tzv. prefix ve formě celého čísla uváděného za adresou sítě, za lomítkem. Toto číslo je rovno počtu jedniček v masce sítě. Kupříkladu maska sítě z předchozího příkladu 255.0.0.0 by byla zapsána /8 a celý zápis adresy sítě by pak vypadal 10.10.10.1/16.

### 2.1 VLSM (Variable Length Subnet Mask)

VLSM neboli síťová maska s proměnnou délkou řeší potřebu rozdělit síť na různě velké části. Praktické využití je nastíněno v následujícím příkladu:

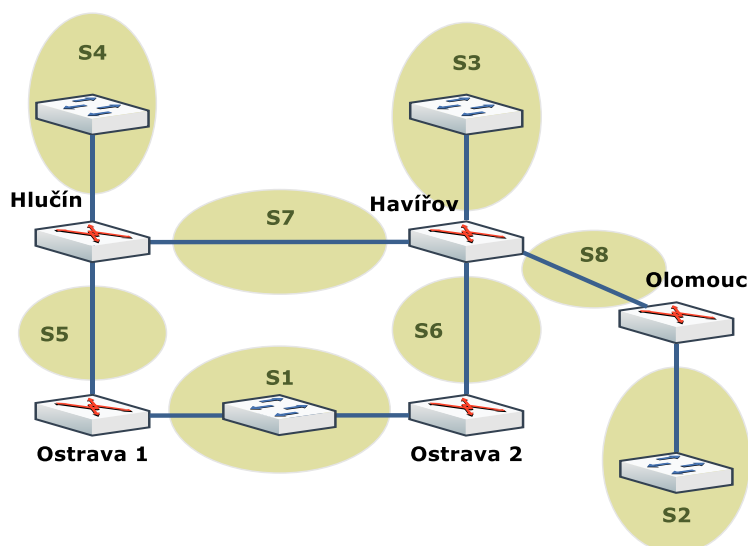
Předpokládejme, že je našim úkolem vytvořit adresování sítě uvedené na Obr. 2.1. Síť je tvořena pěti směrovači v různých městech, propojených spojovacími linkami. V jednotlivých městech jsou připojeny přepínače, na jejichž portech jsou stanice místní lokální sítě. Čísla uvedená u jednotlivých přepínačů určují, s kolika stanicemi připojenými do sítě je v dané lokalitě počítáno.





Obr. 2.1 Architektura sítě

Abychom mohli navrhnout adresování sítě, musíme nejprve stanovit počet IP podsítí a určit minimální počet bitů, který musí být na adresování stanic v každé podsíti vyhrazen. Oblasti tvořící jednotlivé podsítě jsou zobrazeny na Obr. 2.2.



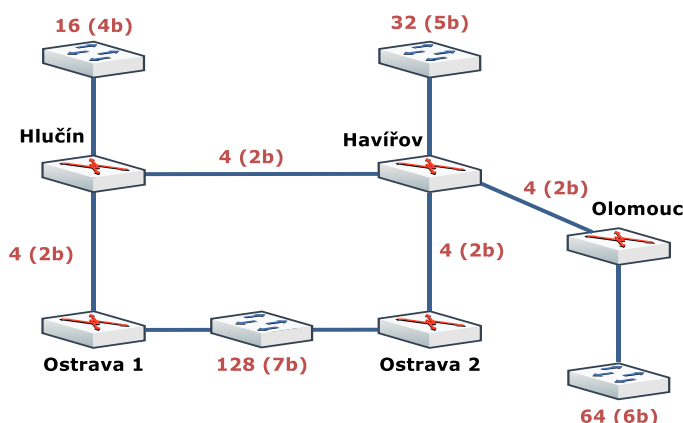
Obr. 2.2 Vyznačení jednotlivých podsítí

Počet bitů, kterými budeme schopni adresovat všechny stanice na dané podsíti, určíme jako nejmenší možný počet bitů, do nichž lze zakódovat číslo odpovídající požadovanému počtu stanic. Jedná se tedy o nejbližší vyšší mocninu dvou větší než požadovaný počet stanic. Při tom však nesmíme zapomínat, že rozhraní směrovače do dané podsítě musí mít také svou IP adresu, takže musíme požadovaný počet stanic zvýšit o jedničku. Mimo to nesmíme zapomenout, že bitová kombinace obsahující samé jedničky je vyhrazena pro broadcast, kombinace samých nul pro označení podsítě jako takové.

Proto pro lokální síť v Hlučíně nepostačí tři bity, jak by se mohlo zdát na první pohled, ale je potřeba čtyři bity. Na Obr. 2.3 jsou počty požadovaných stanic na jednotlivých segmentech upravené na nejbližší mocninu dvou se započtením nepoužitelných vyhrazených adres. Požadované počty adres jsou vyznačeny i u spojovacích (tj. point-to-point) linek.

Všimněte si, že pro adresování stanic spojovací linky je třeba dvou bitů, poskytujících 4 kombinace:

- adresa jednoho routeru,
- adresa druhého routeru,
- označení sítě jako takové,
- broadcast adresa.



Obr. 2.3 Vyznačení počtu požadovaných stanic na jednotlivých segmentech

Celkový počet adres potřebných pro danou síť (se započtením rozhraní routerů i nepoužitelných adres a v jednotlivých podsítích upravený na mocninu dvou) je  $128+64+32+16+4+4+4+4=256$ . Proto poskytovatel Internetu naší síti přidělil prefix o délce 24 bitů (jednu adresu třídy C), čímž pro podsítování zbývá 8 bitů. Přidělená adresa je 213.1.20.0/24.

Pokusme se nyní adresovat síť s použitím konstantní masky podsítě. Podsít s největším množstvím stanic vyžaduje 7 bitů. Z přidělených osmi tak zbývá jeden bit pro určení podsítě, můžeme tedy mít pouze dvě takto velké podsítě. Je zřejmé, že navrhnout adresování s konstantní maskou podsítě nebude při použití 24-bitového prefixu adresy možné. Všimněte si, že i kdyby poskytovatel přidělil více adres (kratší prefix), bylo by v naší síti adresování s konstantní síťovou maskou velice neefektivní: např. spojovacím linkám, vyžadujícím 4 adresy by bylo přiděleno adres 128, čímž by bylo 124 adres neúčinně vyplýtváno.

Proto přistoupíme k adresování s proměnnou maskou podsítě VLSM. Přidělený adresní prostor již nebudeme dělit do bloků o stejné velikosti, jako u konstantní masky podsítě, ale velikost bloků budeme přizpůsobovat počtu stanic na jednotlivých podsítích. Při tom se bloky adres nesmějí překrývat, každé stanici musí být přidělena jednoznačná IP adresa. Počet bitů, jejichž jednoznačnou kombinací jsou určeny bloky adres jednotlivých podsítí, se však bude podle velikosti bloku měnit

S přidělováním adres u VLSM začneme od největší podsítě - S1. Na tu potřebujeme 7 bitů, z bitů použitelných pro podsítování tedy zbývá pro určení podsítě jeden. Síť S1 se rozhodneme přidělit adresy v podsíti, jejíž prefix podsítě bude určen hodnotou 1 v bitu 7,

tedy adresy 213.1.20.128 - 213.1.20.255 (adresa 213.1.20.128 je adresou sítě samotné a 213.1.20.255 je broadcast adresa). Na všechny ostatní sítě tedy zbývá druhá polovina přiděleného rozsahu, tedy adresy 213.1.20.0 - 213.1.20.127.

Pro přehlednost jsou uvedeny v tab.3.1 prefixy jednotlivých sítí a jim odpovídající adresní rozsahy, využít můžeme i přehlednou tabulku na Obr. 2.4.

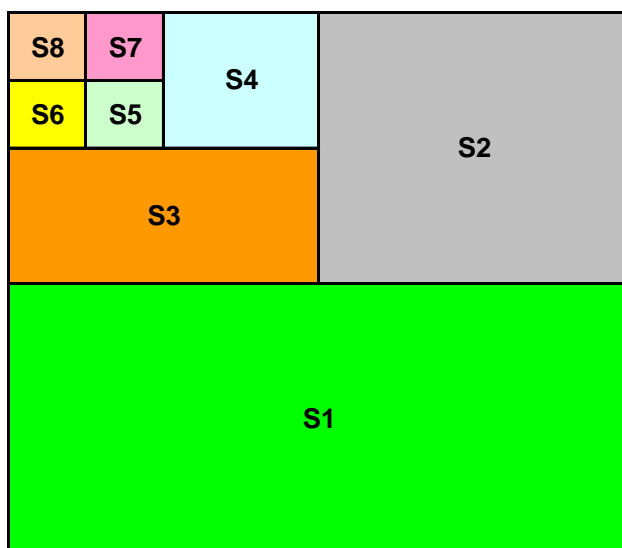
Class C Subnet Table	/24 .0 (00000000) 0 subnets 254 hosts	/25 .128 (10000000) 0 subnet 126 hosts	/26 .192 (11000000) 2 subnets 62 hosts	/27 .224 (11100000) 6 subnets 30 hosts	/28 .240 (11110000) 14 subnets 14 hosts	/29 .248 (11111000) 30 subnets 6 hosts	/30 .252 (11111100) 62 subnets 2 hosts
.0	.0	.0	.0	.0	.0	.0	.0 (.1 -.2)
.4						(.1 -.6)	.4 (.5 -.6)
.8				(.1 -.30)	(.1 -.14)	.8	.8 (.9 -.10)
.12						(.9 -.14)	.12 (.13 -.14)
.16					.16	.16	.16 (.17 -.18)
.20					(.17 -.30)	(.17 -.22)	.20 (.21 -.22)
.24						.24	.24 (.25 -.26)
.28			(.1 -.62)			(.25 -.30)	.28 (.29 -.30)
.32				.32	.32	.32	.32 (.33 -.34)
.36					(.33 -.46)	(.33 -.38)	.36 (.37 -.38)
.40				(.33 -.62)		.40	.40 (.41 -.42)
.44						(.41 -.46)	.44 (.45 -.46)
.48					.48	.48	.48 (.49 -.50)
.52					(.49 -.62)	(.49 -.54)	.52 (.53 -.54)
.56						.56	.56 (.57 -.58)
.60		(.1 -.126)				(.57 -.62)	.60 (.61 -.62)
.64			.64	.64	.64	.64	.64 (.65 -.66)
.68				(.65 -.94)	(.65 -.78)	(.65 -.70)	.68 (.69 -.70)
.72						.72	.72 (.73 -.74)
.76						(.73 -.78)	.76 (.77 -.78)
.80					.80	.80	.80 (.81 -.82)
.84					(.81 -.94)	(.81 -.86)	.84 (.85 -.86)
.88			(.65 -.126)			.88	.88 (.89 -.90)
.92						(.89 -.94)	.92 (.93 -.94)
.96				.96	.96	.96	.96 (.97 -.98)
.100					(.97 -.108)	(.97 -.102)	.100 (.101 -.102)
.104				(.97 -.126)		.104	.104 (.105 -.106)
.108						(.105 -.108)	.108 (.107 -.108)
.112					.112	.112	.112 (.113 -.114)
.116					(.113 -.126)	(.113 -.118)	.116 (.117 -.118)
.120						.120	.120 (.121 -.122)
.124	(.1 -.254)					(.121 -.126)	.124 (.125 -.126)
.128		.128	.128	.128	.128	.128	.128 (.129 -.130)
.132				(.129 -.158)	(.129 -.142)	(.129 -.130)	.132 (.133 -.134)
.136						.136	.136 (.137 -.138)
.140						(.137 -.142)	.140 (.141 -.142)
.144					.144	.144	.144 (.145 -.146)
.148					(.145 -.158)	(.145 -.150)	.148 (.149 -.150)
.152			(.129 -.191)			.152	.152 (.153 -.154)
.156						(.153 -.158)	.156 (.157 -.158)
.160				.160	.160	.160	.160 (.161 -.162)
.164					(.161 -.174)	(.161 -.166)	.164 (.165 -.166)
.168				(.161 -.190)		.168	.168 (.169 -.170)
.172						(.169 -.174)	.172 (.173 -.174)
.176					.176	.176	.176 (.177 -.178)
.180					(.177 -.190)	(.177 -.182)	.180 (.181 -.182)
.184						.184	.184 (.185 -.186)
.188						(.185 -.190)	.188 (.189 -.190)
.192		(.129 -.254)	.192	.192	.192	.192	.192 (.193 -.194)
.196					(.193 -.206)	(.193 -.198)	.196 (.197 -.198)
.200				(.193 -.222)		.200	.200 (.201 -.202)
.204						(.201 -.206)	.204 (.205 -.206)
.208					.208	.208	.208 (.209 -.210)
.212					(.209 -.222)	(.209 -.214)	.212 (.213 -.214)
.216			(.191 -.254)			.216	.216 (.217 -.218)
.220						(.217 -.222)	.220 (.221 -.222)
.224				.224	.224	.224	.224 (.225 -.226)
.228					(.225 -.238)	(.225 -.230)	.228 (.229 -.230)
.232				(.225 -.254)		.232	.232 (.233 -.234)
.236						(.233 -.238)	.236 (.237 -.238)
.240					.240	.240	.240 (.241 -.242)
.244					(.241 -.254)	(.241 -.246)	.244 (.244 -.246)
.248						.248	.248 (.249 -.250)
.252						(.249 -.254)	.252 (.253 -.254)

Obr. 2.4 Přehledná tabulka pro VLSM subnetting

Tab. 2.1 Adresní rozsahy jednotlivých podsítí

lokality - podsít'	označení	adresa sítě	maska podsítě	použitelné adresy	broadcast
Ostrava	S1	213.1.20.128/25	255.255.255.128	213.1.20.129 - 213.1.20.254	213.1.20.255
Olomouc	S2	213.1.20.64/26	255.255.255.192	213.1.20.65 - 213.1.20.126	213.1.20.127
Haviřov	S3	213.1.20.32/27	255.255.255.224	213.1.20.33 - 213.1.20.62	213.1.20.63
Hlučín	S4	213.1.20.16/28	255.255.255.240	213.1.20.17 - 213.1.20.30	213.1.20.31
Spoj Hlučín-Ostrava1	S5	213.1.20.12/30	255.255.255.252	213.1.20.13 - 213.1.20.14	213.1.20.15
Spoj Ostrava2-Haviřov	S6	213.1.20.8/30	255.255.255.252	213.1.20.9 - 213.1.20.10	213.1.20.11
Spoj Hlučín-Haviřov	S7	213.1.20.4/30	255.255.255.252	213.1.20.5 - 213.1.20.6	213.1.20.7
Spoj Haviřov-Olomouc	S8	213.1.20.0/30	255.255.255.252	213.1.20.1 - 213.1.20.2	213.1.20.3

Pro lepší přehlednost můžeme bloky adres postupně přidělované jednotlivým podsítím vyjádřit také graficky. Představme si poskytovatelem přidělený adresní rozsah 256 adres jako čtverec 16 x 16, v jehož levém horním rohu je nejnižší adresa rozsahu, v pravém dolním nejvyšší, viz Obr. 2.4.



Obr. 2.5 Grafická reprezentace jednotlivých podsítí

## 2.2 Podpora síťových prvků

Výskyt různých délek podsítí, tj. síťových masek v rámci jedné sítě ovšem vyžaduje implementaci beztrždního routovacího protokolu (*Classless Routing Protocol*). Podporuje ho směrovací protokol RIPv2, EIGRP, OSPF, ISIS. Směrovací protokoly RIP a IGRP beztrždní směrování nepodporují.

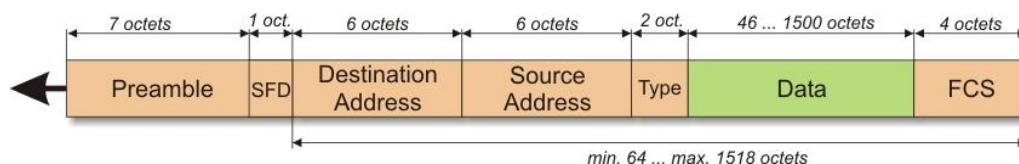
## 3. Ethernet

Ethernet je název souhrnu technologií pro počítačové sítě (LAN, MAN) z větší části standardizovaných jako IEEE 802.3, které používají kabely s kroucenou dvoulinkou, optické kabely (ve starších verzích i koaxiální kabely). Ethernet realizuje fyzickou a linkovou vrstvu referenčního modelu OSI. V současné době je Ethernet nejúspěšnější síťovou technologií. [1,2,3]

### 3.1 Ethernet rámec

Ethernet rámec je protokolová datová jednotka linkové vrstvy v technologii Ethernet. Rámec následně slouží v linkové vrstvě k zapouzdření paketů, předaných ze síťové vrstvy k odvysílání na určitý typ média.

Rámec v síti Ethernet má proměnlivou délku, minimální délka je 64 bytů, maximální 1518 bytů. Formát rámce je uveden na Obr. 3.1.



Obr. 3.1 Ethernet rámec

Význam jednotlivých polí je následující:

- Záhlaví (Preamble) - zahájení rámce, synchronizace
- Cílová adresa (Destination Ethernet Address)- adresa příjemce (8 bytů)
- Zdrojová adresa (Source Ethernet Address) - adresa odesílatele (8 bytů)
- Typ rámce (Length or Type)
- Data - minimálně 46 bytů, maximálně 1500 bytů (maximální délka přenášených označuje jako MTU – Maximum Transmission Unit)
- CRC (Cyclic Redundancy Check) - Kontrolní posloupnost rámce (Frame Check Sequence)

## 3.2 Typy Ethernetu

### 3.2.1 Fast Ethernet

Očividnou změnou oproti Ethernetu klasickému je ukončení podpory koaxiálního kabelu a tím i sběrníkové topologie – tyto již v novém Ethernetu nelze použít. Typickou topologií se stala hvězda.

Specifikace 100Base-TX využívá nestíněnou kroucenou dvoulinku (UTP) kategorie 5 s využitím dvou párů. Umožňuje použít i stíněnou kroucenou dvoulinku (STP) Type 1 a Type 2 (2 páry). Maximální délka segmentu může být 100m, nosná frekvence je 125 MHz a data jsou kódována metodou 4B/5B.

### 3.2.2 Gigabit Ethernet (1GE)

Gigabitový Ethernet je poměrně významným posunem v rychlostech přenosu. Základ jeho fyzické vrstvy byl tvořen podle standardu Fibre Channel. Využívá metodu náhodného přístupu CSMA/CD v podvrstvě MAC a struktura rámce zůstává stejná jako u jeho předchůdců.

Gigabitový Ethernet pracuje rychlostí 1 Gbps. V roce 1998 byl přijat standard IEEE 802.3z zahrnující dvě specifikace (1000Base-SX, 1000Base-LX) pro optické kabely a jednu specifikaci (1000Base-CX) pro metalickou kabeláž (někdy se všechny tyto specifikace označují jako 1000Base-X). Později byl schválen druhý standard IEEE 802.3ab, který zahrnuje specifikaci (1000Base-T) podporující přenos na UTP kabelech kategorie 5.

Je zachována minimální velikost rámce 64B, která je dána standardem IEEE 802.3, čímž je zaručena kompatibilita starších Ethernetů. Gigabitový Ethernet používá sice stejný minimální rámec o velikosti 64 B. Přehled jednotlivých verzí gigabitového Ethernetu je uveden v tab. 3.1.

Tab. 3.1 Přehled jednotlivých verzí gigabitového Ethernetu

Název	Médium	Vzdálenost
1000BASE-CX	Twinaxiální kabel	25 metrů
1000BASE-SX	Vícevidové optické vlákno	220 až 550 metrů, záleží na šířce vlákna a šířce pásma
1000BASE-LX	Vícevidové optické vlákno	500 metrů
1000BASE-LX	Jednovidové optické vlákno	5 km
1000BASE-LX10	Jednovidové optické vlákno využívající vlnovou délku 1,310 nm	10 km
1000BASE-ZX10	Jednovidové optické vlákno využívající vlnovou délku 1,550 nm	~ 70 km
1000BASE-BX10	Jednovidové optické vlákno v jednom směru s vlnovou délkou 1490 nm downstream a 1310 nm upstream	10 km
1000BASE-T	Kroucená dvojlinka (cat-5, cat-5e, cat-6 nebo cat-7)	100 metrů
1000BASE-TX	Kroucená dvojlinka (cat-6, cat-7)	100 metrů

### 3.2.3 10 Gigabit Ethernet (10 GE)

Další varianta technologie Ethernetu - 10Gigabit Ethernet (10GE) dle standardu IEEE 802.3ae byla schválena v roce 2002. Standard IEEE 802.3ae pro 10Gbps Ethernet byl navržen pouze pro plně duplexní provoz. Díky tomu zde neexistuje omezení vzdálenosti mezi uzly, vyplývající z principu přenosové metody. Tato vzdálenost je omezena pouze fyzikálními vlastnostmi přenosového média a optických přenosových prvků. Je odstraněna fyzická i logická sběrnice, neboť standard striktně využívá jen dvoubodové spoje.

10 Gigabit Ethernet je v dnešní době velmi používanou specifikací. Jednotlivé specifikace standardu IEEE 802.3ae jsou uvedeny v tabulce 3.2. Oproti svým předchůdcům se neuplatňuje pouze v lokálních sítích na propojování prvků, ale je využíván i v datových centrech, úložných sítích (SAN), sítích metropolitních a WAN. Bývá kombinován se standardem 40 GBase a nebo 100 GBase. V LAN se používá pro spojování serverových klastrů, kde jsou segmenty s agregací 1000 Base-X nebo 1000 Base-T do jednoho spoje. Využívá se i k propojení mezi servery, přepínači a nebo dvěma přepínači v rámci datového pole.

Tab. 3.2. Přehled verzí 10 GE

Standard	Medium	Dosah [km]
10GBase-SR/SW (850 nm)	mnohovidové vlákno	0,3
10GBase-LR/LW (1350 nm)	jednovidové vlákno	10
10GBase-ER/EW (1550 nm)	jednovidové vlákno	40
10GBase-LX4 (1550 nm)	jednovidové/mnohovidové vlákno	10/0,3

### 3.2.4 40 a 100 Gigabit Ethernet (40 / 100 GE)

Technologie 40/100 GBase Ethernet definuje standard IEEE 802.3ba. Podobně jako 10GBase Ethernet využívá plný duplex. Na rozdíl od jiných standardů má dvě rychlosti 40 Gb/s a 100 Gb/s. Nižší z rychlostí je využívána na propojení datových centrech a 100 Gb/s se používá převážně na spojování přepojovacích prvků. Seznam jednotlivých verzí je uveden v tabulce 3.3.

Tab. 3.3. Přehled verzí 10 GE

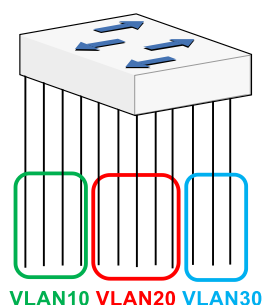
Standard	Dosah [km]
40GBase-SR4	100/150 m
40GBase-FR	2 km
40GBase-LR4	10 km
40GBase-ER4	30 (40) km
40GBase-LM4	140/160 m
100GBase-SR10	100/150 m
100GBase-SR4	70/100 m
100GBase-LR4	10 km
100GBase-ER4	30 (40) km

### 3.2.5 200 a 400 Gigabit Ethernet, Terabit Ethernet

Terabit Ethernet neboli TbE je označení pro Ethernet s přenosovými rychlostmi vyššími než 100 Gbit/s. V roce 2017 byly schváleny standardy 400 Gigabit Ethernet (400G, 400GbE) a 200 Gigabit Ethernet (200G, 200GbE) vyvinuté pracovní skupinou IEEE P802.3bs používající zhruba stejnou technologii jako 100 Gigabit Ethernet. Ethernet Alliance předpokládá, že Ethernet s přenosovou rychlostí 800 Gbit/s a 1,6 Tbit/s bude standardizován po roce 2021.

## 4. VLAN

Virtuální LAN slouží k logickému rozdělení sítě nezávisle na fyzickém uspořádání. Hlavním principem VLAN je segmentace na menší sítě uvnitř fyzické struktury původní sítě. Pomocí VLAN můžeme takovéto dvě sítě vytvořit na jednom nebo několika propojených přepínačích. Mechanismus VLAN dovoluje rozdělit porty přepínače do skupin - virtuálních sítí s tím, že provoz může procházet vždy jen mezi porty téže skupiny, viz. obr.4.1. [1,2,3]

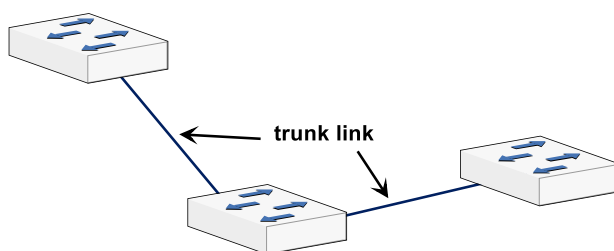


Obr. 4.1 VLAN na přepínači

Pro každou virtuální síť si přepínač vytváří a používá samostatnou přepínací tabulku. Přepínač je tak logicky rozdělen na více nezávislých logických přepínačů, a to čistě na základě konfigurace přepínače, bez potřeby jakéhokoli fyzického přepojování. Tímto získáváme možnost seskupování stanic do vzájemně nezávislých, tzv. virtuálních sítí. Logická struktura sítě se tím stává nezávislou na fyzické topologii.

### 4.1 Standard IEEE 802.1q

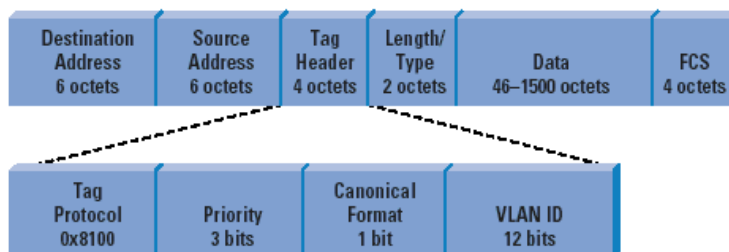
V praxi je nutno propojit virtuální sítě na portech různých přepínačů. Spoje mezi přepínači však v tomto případě musí nést současně provoz z více virtuálních sítí, proto je nutné na spojích mezi přepínači identifikovat příslušnost jednotlivých rámců k virtuální síti, ze které byl vyslán. K tomu se používá speciální identifikace v hlavičce rámce Ethernet, tzv. *tag*. Spoje mezi přepínači, na kterých jsou rámce označovány značkou, jsou často nazývány *trunk*, viz. obr. 4.2.



Obr. 4.2 Trunk linka mezi přepínači



K identifikaci příslušné VLAN dochází ve formátu rámce Ethernet. Používá se hlavička obsahující číslo VLAN v rozsahu 0-4095 vložená do pole *Tag Header*. Mimo čísla VLAN hlavička podle normy 802.1q může nést i prioritu rámce. Původní hodnota pole EtherType pak následuje za hlavičkou 802.1q., viz. obr.4.3.



Obr. 4.3 Rámec Ethernetu s hlavičkou 802.1q s identifikátorem VLAN

Záhlaví podle IEEE 802.1q prodlužuje rámec o 2B. Hardware síťových karet, resp. portů přepínačů tak musí být schopny zpracovávat rámce s maximální délkou (MTU) o 2B vyšší, než definuje klasická norma Ethernetu.

## 4.2 Protokol VTP

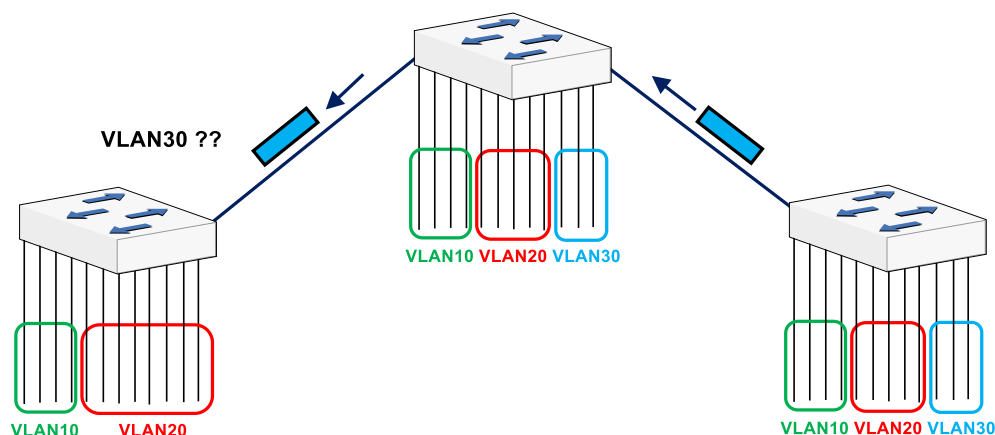
Většinou chceme, aby vytvořené VLANy existovaly v celé síti na všech přepínačích. Pro přenášení informací o těchto VLAN se využívá protokolu VTP (VLAN Trunking Protocol). VTP je L2 protokol, který tedy slouží ke správě (přidávání, mazání, přejmenování) VLAN uvnitř VTP domény. VTP doména je tvořena jedním nebo více síťovými zařízeními, která mají nastaveno stejné jméno domény (volitelně i heslo) a jsou propojeny pomocí *trunk* spoje.

Přepínač může pracovat v režimu:

- server – spravuje seznam všech VLAN, má jej uložen v NVRAM, může vytvářet a mazat VLAN, přijímá a odesílá informace o VLAN přes trunk linky ve VTP doméně
- klient – přijímá konfiguraci ze serveru, udržuje lokální kopii všech VLAN, kterou nelze měnit a nemá ji uloženou v NVRAM, přijímá a odesílá informace o VLAN
- transparentní – neúčastní se VTP, pracuje samostatně, může vytvářet i mazat VLANy, ale změny jsou lokální, přijímá advertisements a ve verzi 2 je i přeposílá (ale nesynchronizuje svoje VLANy, ani je nezveřejňuje), je to jediný mód, kde můžeme vytvářet Extended a Private VLANy, VTP a VLAN konfigurace je uložena v NVRAM

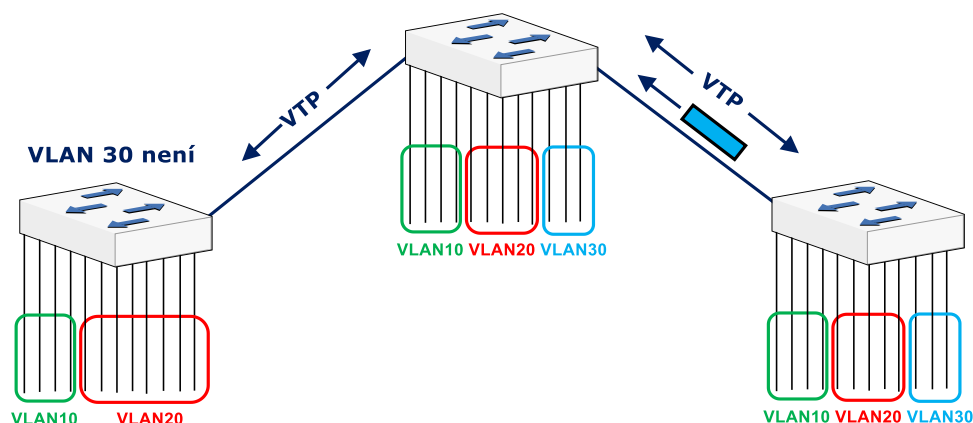
### 4.2.1 Prořezávání (trunking) protokolu VTP

Ethernet rámce vyslané na broadcast adresu jsou vyslány na všechny trunk porty. Zde však může dojít k situaci, kdy přes trunk linky putují rámce k cílovému přepínači i z těch VLAN, u kterých není na cílovém přepínači zařazen žádný port, tj. VLAN na cílovém přepínači není, viz obr. 4.4.



Obr. 4.4 Ethernet rámec s VLAN30 vyslaný na všechny přepínače

Proto přepínače některých výrobců implementují protokol pro tzv. **prořezávání** topologie jednotlivých VLAN, jímž se jednotlivé přímo propojené přepínače informují o číslech VLAN, které mají na jednotlivých portech. Jedná se zpravidla o protokol Cisco VTP, viz. obr. 4.5.



Obr.4.5 Aplikace protokolu VTP

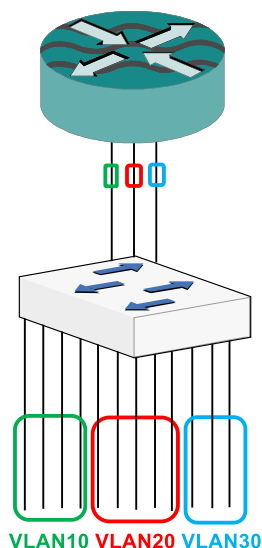
V případě kombinace výrobců zařízení, a tedy nemožnosti použít jednotný protokol pro prořezávání topologie, se pro filtraci jednotlivých VLAN používá statické konfigurace. Tj. danému přepínači se nastaví na příslušném trunk portu které VLAN má propouštět a které nikoliv.

## 4.3 Směrování mezi VLAN

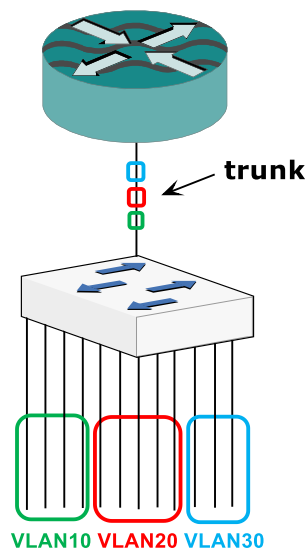
V některých případech bývá užitečné zajistit směrování mezi dvěma VLAN. Toto můžeme realizovat dvěma způsoby:

- pro každý VLAN použijeme fyzický port směrovače, viz. obr. 4.6
- nebo směrování umožníme pomocí jednoho fyzického portu směrovače, tzv. *Router on the stick*, viz. obr. 4.7.

V druhém případě musí mít jeden fyzický port směrovače nastaven tolik tzv. *subinterface*, kolik VLAN je potřebováno směrovat. Nesmíme zapomenout nastavit typ enkapsulace na L2 vrstvě 802.1q a každém *subinterface* přiřadit samostatnou IP adresu. Výhodou tohoto druhého přístupu je ušetření počtu rozhraní směrovače, kdy počet požadovaných rozhraní směrovače neroste s počtem VLAN mezi kterými má směrovač směrovat. Nevýhodou je nižší propustnost, protože při směrování paketu mezi VLAN se využívá jedné fyzické linky.



Obr.4.6 Směrování mezi VLAN pomocí různých portů směrovače

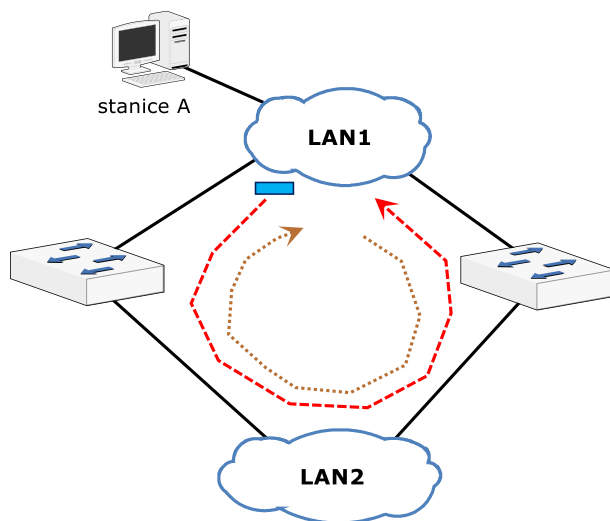


Obr.4.7 Směrování mezi VLAN pomocí *Router on the stick*

## 4.4 Protokol Spanning Tree

Protokol Spanning Tree (STP) je algoritmus k vytvoření logické topologie sítě **bez smyček** v přepínané síti **se smyčkami**. Představme si síť na obr. 4.8. Tato síť je tvořena dvěma přepínači, kde data mohou být směrována ze segmentu A do segmentu B přes dvě cesty

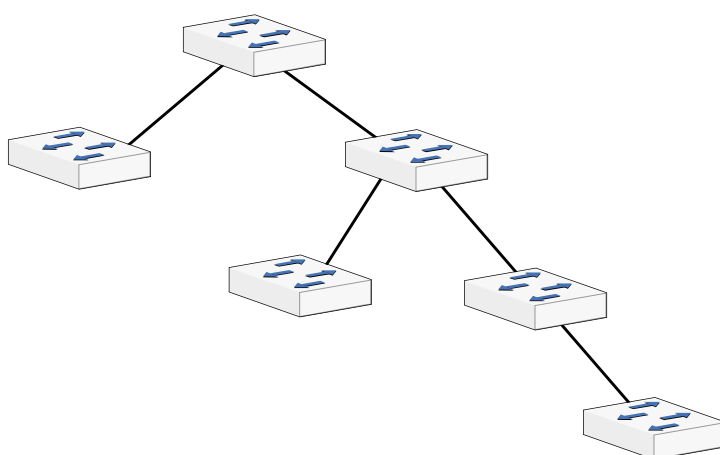
(přes přepínač S1 nebo přes přepínač S2). V této topologii se takto vytváří **smyčka**, ve které by např. *broadcast* rámce cirkulovaly nekonečně.



Obr. 4.8 Cirkulace rámců v topologii se smyčkou

Navíc přítomnost smyčky vede nejen k cirkulaci nebo generování kopií rámců, ale zcela nabourává i samotný princip automatického učení přepínací tabulky přepínače. Cirkulující rámce se zdrojovou adresou stanice **A** jednou přicházejí do přepínačů z LAN1 a jindy z LAN2. Tímto si přepínače neustále střídavě a v polovině případů nesprávně podle portu příchozího rámce aktualizují informaci, za kterým portem je vlastně stanice A připojena.

Z uvedeného případu je zřejmé, že vytvoření smyčky v síťové topologii je **nežádoucí**. Žádoucí ale je mít redundantní linky pro případ zálohy spojení. Proto musíme mít k dispozici mechanismus, který v topologii obsahující smyčky zablokuje některé porty přepínačů tak, aby výsledná topologie byla stromová, např. viz. obr. 4.9.



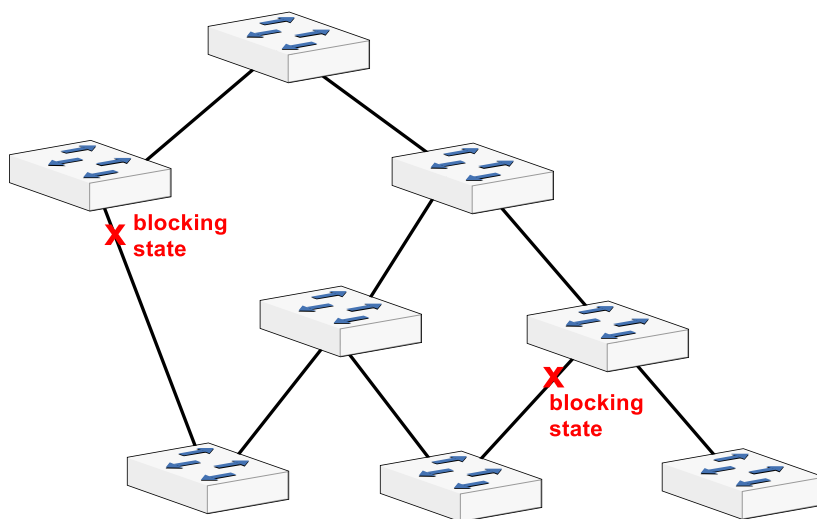
Obr. 4.9 Stromová topologie bez vytvořených smyček

Takovýto mechanismus zajišťuje protokol **Spanning Tree** IEEE 802.1d. Úkolem tohoto protokolu je **neustálé udržování stromové struktury** na danou topologií přepínačů. Důležité vlastnosti Spanning Tree protokolu jsou:

- blokáce linky probíhá vždy jen z jedné strany přepínače, viz. obr.4.10.,
- algoritmus pracuje neustále, v případě výpadku linky nebo portu přepínače se strom automaticky přepočte (odblokuje se některý doposud zablokovaný port).

Vytváření stromu se děje ve dvou krocích, které však probíhají neustále a současně:

1. volba kořenu stromu, tzv. *Root Bridge*
2. vytvoření stromu preferovaných (nejmenší *cost*) cest z každého přepínače k *Root Bridge*.



Obr. 4.10 Blokáce portu při použití Spanning Tree

Redundantní spoje, které nejsou součástí nejkratší cesty stromu, jsou blokovány. Data, která přicházejí na blokované spoje, jsou zahozena. Díky tomu vznikne logická topologie bez smyček. Protokol **Spanning Tree** vyžaduje komunikaci mezi zařízeními, aby detekoval smyčky. Spoje, které vytváří smyčky, jsou dány do blokujícího stavu (*Blocking State*).

Přepínače si posílají zprávy *Bridge Protocol Data Units* (BPDU) pro získání informací o logické topologii bez smyček. Blokované porty přijímají BPDU a to zabezpečí, že když selže aktivní cesta nebo zařízení, vypočítá se nový strom.

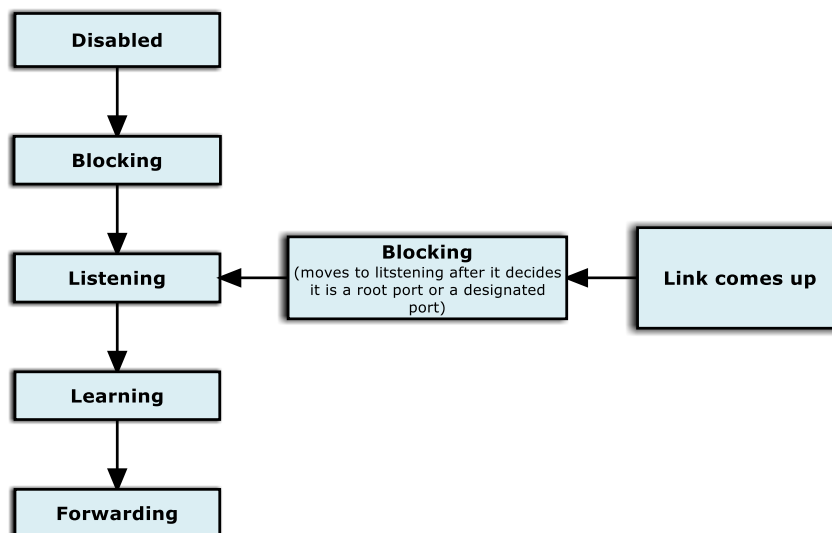
BPDU obsahují informace, pomocí nichž si mohou přepínače:

- vybrat jeden přepínač, který bude fungovat jako *Root Bridge*,
- spočítat nejkratší cestu k *Root Bridge*,
- vybrat jeden ze svých portů jako kořenový port (*Root Port*) pro každý přepínač, který není kořenovým přepínačem. *Root Port* je port s nejlepší cestou k *Root Bridge*.
- Nastavit porty, které jsou součástí spanning tree, tzv. *Designated Ports*.

Z důvodu zabránění smyčkám během konvergence (volba *Root Bridge*) algoritmus definuje přechodné stavy portů, tzv. *Learning a Listening*. Port ve stavu *Listening* nepřeposílá rámce, ale pouze po dobu 15 sekund sleduje okolní provoz, aby mohl rozhodnout, zda se přepne do stavu *Forwarding* (běžný provoz) nebo *Blocking*. Před přechodem do stavu

*Forwarding* se navíc 15 sekund ve stavu *Learning* bez přeposílání rámců pouze učí MAC adresy okolních stanic. Tímto se omezí procento rámců, jejichž příjemce není v přepínací tabulce a musí být tudíž rozeslány na všechny porty.

Jednotlivé fáze portu jsou uvedeny na obr. 4.11.



Obr. 4.11 Fáze portu přepínače protokolu Spanning Tree

- **Blocking state** – porty přijímají pouze BPDUs. Data jsou zahazována a přepínač se na tomto portu neučí MAC adresy. V tomto stavu přetrvává maximálně 20 sekund.
- **Listening state** – přepínač zjišťuje, zda nevede další cesta k root bridge. Cesty, které nemají nejnižší *cost path* k *root bridge* se vrátí do blocking state. Doba naslouchání se nazývá *forward delay* a trvá 15 sekund. Data nejsou přeposílána a MAC adresy se neučí, ale BPDUs se stále zpracovávají.
- **Learning state** – učí se MAC adresy, ale data stále nejsou přeposílána. BPDUs jsou stále zpracovávány. Trvá 15 sekund.
- **Forwarding state** – se přeposílají data, učí se MAC adresy a BPDUs jsou zpracovávány.
- Port může být i v *Disabled State*. V tomto stavu je port, když je administrátorem vypnut nebo má poruchu.

## 5. Paketové filtry

Paketový filtr neboli firewall je síťové zařízení, které slouží k řízení a zabezpečování síťového provozu tím, že blokuje nebo povoluje navazované komunikace na základě předdefinovaných nebo dynamických pravidel a politik. Zjednodušeně se dá říct, že slouží jako kontrolní bod, který definuje pravidla pro komunikaci mezi sítěmi, které od sebe odděluje. Tato pravidla historicky vždy zahrnovala identifikaci zdroje a cíle dat (zdrojovou a cílovou IP adresu) a zdrojový a cílový port. Dnešní moderní firewally rovněž kontrolují stav spojení, dokáží inteligentně filtrovat provoz i na základě protokolů aplikační vrstvy s funkcionalitami podobnými IDS (Intrusion Detection System).

Instalace firewallu navýší celkové zabezpečení IT infrastruktury firmy či domácnosti. Pokud je firewall dobře nastaven, pak jde o jediné vstupní místo, přes které musí projít veškerá komunikace. Firewall pak chrání všechna zařízení před škodlivou příchozí komunikací. [1,2,3]

### 5.1 Základní princip filtrace

Nejjednodušší a nejstarší forma funkce firewallu spočívá v tom, že pravidla přesně uvádějí, z jaké adresy a portu na jakou adresu a port může být doručen procházející paket. Pravidla se tedy aplikují pouze na komunikaci související se síťovou a transportní vrstvou referenčního modelu OSI.

Výhodou tohoto řešení je vysoká rychlost zpracování, proto se ještě i dnes používají na místech, kde není potřebná přesnost nebo důkladnější analýza procházejících dat, ale pouze základní funkce.

Mezi typické představitele paketových filtrů patří ACL (Access Control Lists) na routerech Cisco, viz. obr. 5.1 nebo nástroj iptables v Linuxu (obr. 5.2).

```
SWITCH(config)#access-list 5 deny host 10.5.1.10
SWITCH(config)#access-list 5 permit 10.5.1.10 0.0.0.255
SWITCH(config)#access-list 5 deny any
```

Obr. 5.1 Příklad ACL

```
iptables -A INPUT -p tcp --dport 80 -j DROP
iptables -A INPUT -p tcp --dport 90 -j DROP
iptables -A INPUT -p tcp --dport 100 -j DROP
```

Obr. 5.2 Příklad IP tables

### 5.2 Stavové paketové filtry

Stavové paketové filtry provádějí kontrolu stejně jako jednoduché paketové filtry, navíc si však ukládají informace o probíhajících spojeních protokolu TCP. Tyto informace pak mohou využít při rozhodování, zda procházející pakety patří do již povoleného spojení a mohou být propuštěny, nebo zda musí znovu projít rozhodovacím procesem.

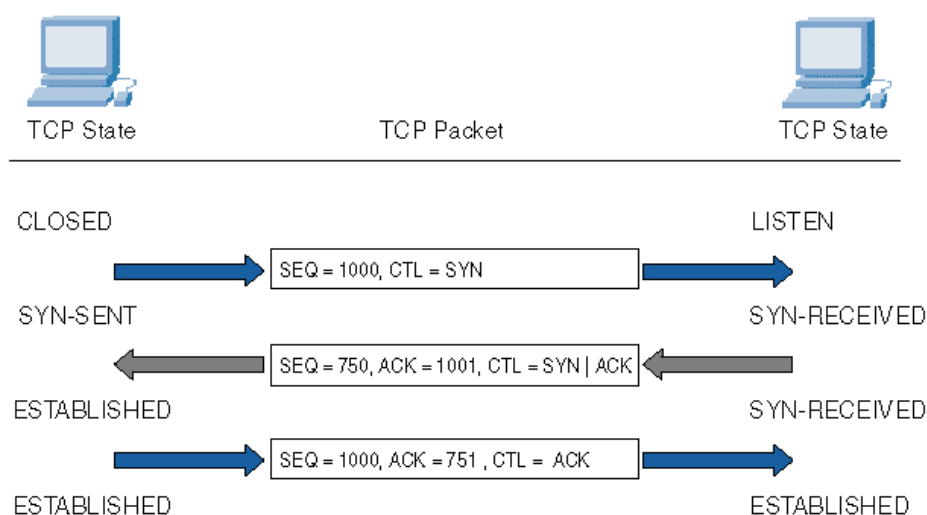
Většina bezstavových paketových filtrů jednoduše povoluje přes firewall všechny porty nad 1024, protože tyto porty se používají pro zpětné sokety z interní sítě. Toto zabezpečení je ale nedostatečné. Nic nebrání např. trojským koním, aby na interní síti vyčkávaly na jakémkoliv portu vyšším než 1024. Bezstavové paketové filtry nemohou tomuto druhu proniknutí zabránit.

Stavové paketové filtry naopak nepropouštějí přes firewall žádné služby, kromě služeb, u nichž mají nastavené povolení, a kromě připojení, která už mají ve svých stavových tabulkách.

Nejnáročnější kontrola se tedy provádí v době sestavení spojení při procesu TCP handshake. Po tomto procesu jsou už všechny pakety (pro danou relaci) zpracovávány rychleji, protože je snadné určit, zda patří do stávající relace nebo ne. Po ukončení spojení se z tabulky daná relace vymaže. Datagram se může nacházet v těchto stavech:

- NEW – datagram otevírá novou komunikaci,
- ESTABLISHED, RELATED – datagram patří do již navázaného spojení.
- INVALID – datagram nepatří do žádného spojení nebo je neidentifikovatelný.

Proces handshake klient zahájí odesláním příznaku SYN=1 v hlavičce paketu. Každý paket, který má nastavený SYN=1 je považován firewallem jako zahajovací paket nového spojení. Pokud je služba požadována klientem na serveru k dispozici, server odpoví odesláním paketu s nastaveným příznakem SYN=1 a ACK=1. Klient pak odpoví paketem, ve kterém je nastaven jen bit ACK=1 a spojení je tímto označeno za ESTABLISHED. Takovým firewallem projdou všechny odchozí pakety, ale příchozí pakety projdou jen takové, které jsou součástí **ESTABLISHED** spojení, viz. obr. 5.3. Toto zabezpečení zabrání hackerům zahájit nežádoucí spojení.



Obr. 5.3 Funkcionalita stavového firewallu

Aby se zabránilo neustálého zaplňování tabulky, je integrován systém, který po určité době spojení z tabulky smaže. Proto mnoho aplikací posílá tzv. „keepalive“ zprávy, aby se firewall nerozhodl spojení ukončit. Za zmínku stojí uvést, že nejčastějším útokem na internetu typu



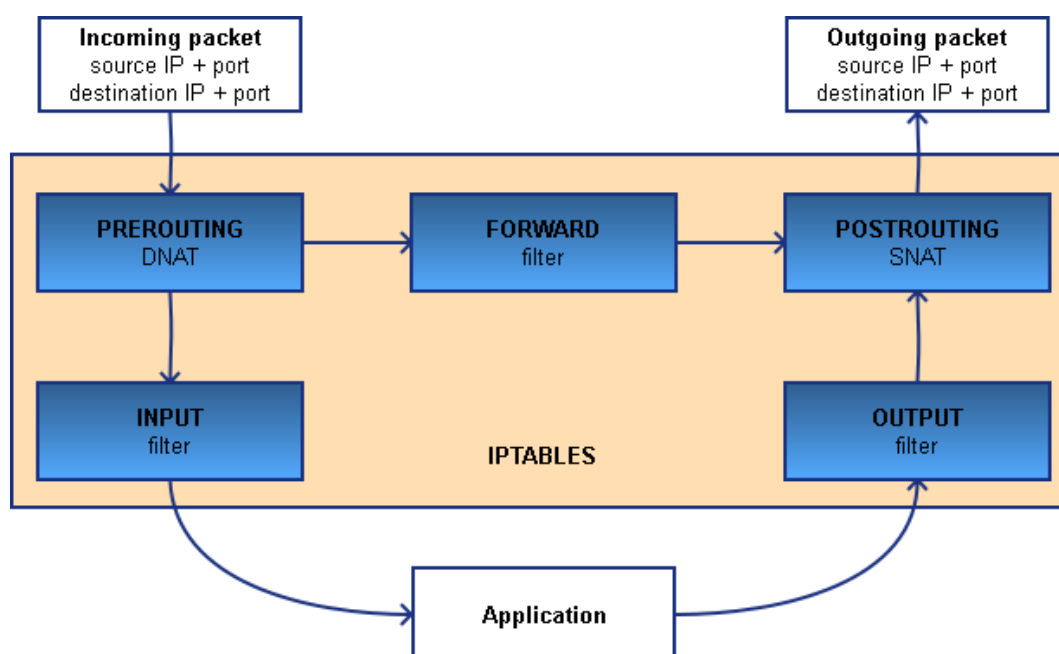
Denial of service je SYN-flood. Tedy útok, kdy útočník odesílá mnoho paketů s příznakem SYN cílovému počítači, ale dále již neodpovídá. To má za následek přeplnění stavové tabulky a mj. zpomalení serveru, ale třeba i zhroucení systému a server pak musí být lokálně restartován. K největším výhodám stavových paketových filtrů patří jejich vysoká rychlost a úroveň zabezpečení a ve srovnání s výše zmíněnými aplikačními branami a jednoduchými paketovými filtry řádově mnohonásobně snazší konfigurace – a díky zjednodušení konfigurace i nižší pravděpodobnost chybného nastavení pravidel obsluhou.

## 6. NAT

NAT (Network Address Translation) je v oblasti počítačových sítí způsob úpravy síťového provozu procházejícího přes router přepisem (tzv. překladu) zdrojové nebo cílové IP adresy, případně i hlaviček protokolů vyšší vrstvy, např. číslo portu u TCP, UDP, apod. NAT může být implementován softwarově na běžném počítači (např. v jádře Linuxu pomocí iptables/netfilter) nebo může být realizován přímo ve firmware či hardware routeru. Pomocí překládání síťových adres (NAT) se nejčastěji převádějí privátní IP adresy v privátní síti na jedinečné veřejné IP adresy, které lze použít v Internetu. Směrovače mají pro překládání síťových adres k dispozici tabulku obsahující interní sokety přiřazené k externím soketům.

Na obr. 12.1 je uveden příklad funkcionality NAT. Oba počítače jsou připojeny k Internetu přes směrovač, kde se provádí překlad adres. Rozlišujeme DNAT, SNAT a speciální případ SNAT – Masquerade. SNAT neboli Source NAT je technika při které se mění zdrojová (source) IP adresa a to po procesu směrování (postrouting). [4]

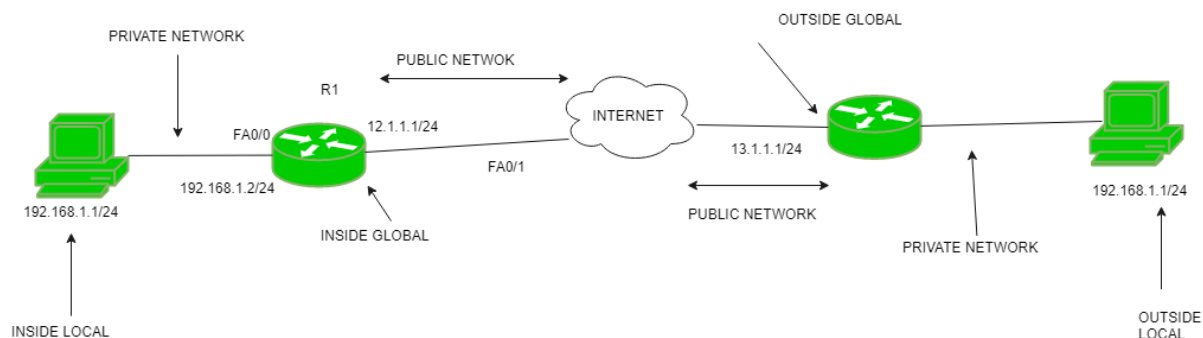
Na obr. 6.1. je znázorněno schéma **IPTABLES**, které se skládá z několika řetězců. PREROUTING je sada pravidel aplikovaných na příchozí pakety před jejich zpracováním v směrovací tabulce. Pokud paket směřuje dovnitř, aplikují se sady pravidel INPUT. Analogicky pro odchozí pakety OUTPUT. FORWARD je použit, pokud server funguje jako router. Pakety, procházející přes FORWARD, neprocházejí sadou pravidel INPUT ani OUTPUT. Pomocí PREROUTING lze modifikovat cílovou adresu Destination NAT (DNAT). Opakem je POSTROUTING – modifikujeme pakety, které již prošly směrovací tabulkou a lze na ně aplikovat pravidla Source NAT (SNAT).



Obr. 6.1 Schéma IPTABLES

Máme-li PC s IP adresou 192.168.1.1 v privátní síti, pak po průchodu paketu směrovačem s funkcí SNAT tento mění na adresu 12.1.1.1. **MASQUERADE** je zvláštní a zároveň asi

nejpoužívanější případ SNAT, kde se může být v privátní síti více zařízení. Jednotlivé IP adresy těchto zařízení jsou překládány na jednu vnější IP adresu, viz. obr. 6.2



Obr. 6.2 Princip NAT

DNAT neboli Destination NAT je naopak technika, při níž se mění cílová IP adresa zařízení. Může se tak realizovat např. port forwarding. DNAT se provádí před směrováním (prerouting).

## Seznam použité literatury

- [1] TANENBAUM, Andrew S. *Computer networks*. 4th ed. New Jersey: Prentice-Hall, c2003. ISBN 9780130661029.
- [2] ODOM, Wendell. *CCNA 200-301 Official Cert Guide Library*. 1. Cisco Press, 2020. ISBN 9780136755449.
- [3] PYLES, James, Jeffrey L. CARRELL. *Guide to TCP/IP: IPv6 and IPv4*. Cengage Learning, 2016. ISBN 9781337020541.
- [4] PETERSEN, Richard. *Ubuntu 20.04 LTS Server: Administration and Reference*. Surfing Turtle Press, 2020. ISBN 9781949857139.



Skriptu Komunikační sítě II (2. část), jejichž autory jsou Libor Michalek a Petr Machník, podléhají licenci [Creative Commons Uveďte původ 4.0 Mezinárodní Licence](https://creativecommons.org/licenses/by/4.0/).