PREDNASKA • POCITACOVE SIETE

Referenčný model TCP/IP a úvod do CISCO zariadení

Ako naozaj funguje sieť — od kabla az po prehliadac

Gabriel Bugár
KPS TUKE

O com si dnes povieme

01 **Uvod a historia**

Co je TCP/IP, ako vznikol a preco vytlacil OSI model

02 **4 vrstvy modelu**

Sietova pristupova, Internetova, Transportna, Aplikacna

03 **Kluc ove protokoly**

IP, TCP, UDP, HTTP, DNS — a ich ulohy

04 **Prakticke prikazy**

ping, tracert, ipconfig, netstat, nslookup, curl

05 **Realne priklady**

Otvorenie webovej stranky, email, video streaming

06 **Wireshark a cvicenia**

Analyza skutocnych paketov a ulohy pre vas

Co je pocitacova siet a preco vrstvovy model?

DEFINICIA

Pocitacova siet

Skupina prepojenych zariadeni (pocitacov, telefonov, serverov, IoT), ktore si dokazu vymienat data prostrednictvom prenosoveho media — kabla, optiky alebo radioveho signalu.

PRECO MODEL?

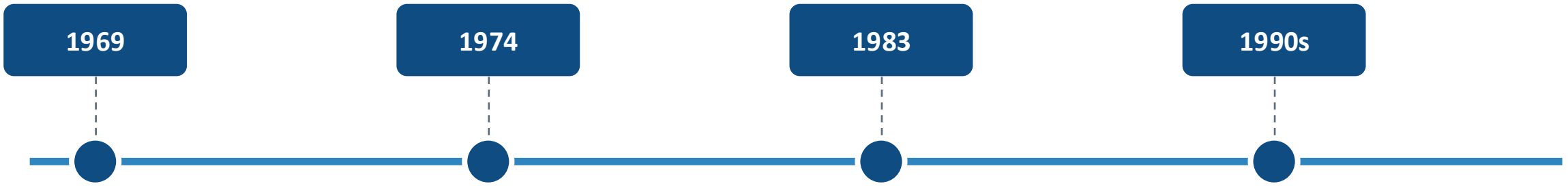
- Rozdeli zlozity problem na mensie casti
- Kazda vrstva ma jasnu ulohu (separation of concerns)
- Vyrobcovia mozu nezavisle vyvijat hardware a software
- Uzivatel nemusi vediet, co sa deje pod kapotou

ANALOGIA: POSTOVA SLUZBA

Vrstvy ako roznym ludom v procese:

Aplikacia	Pisete list (obsah)
Transport	Davate list do obalky a oznacujete ID
Internet	Pisete adresu kam to ma ist
Sietov pris.	Postar to fyzicky preneše do auta

Kratka historia TCP/IP



ARPANET

Prvy paketovo prepinany protokol financovany US Dept. of Defense

Vinton Cerf + Bob Kahn

Prvy navrh protokolu TCP ("A Protocol for Packet Network Intercommunication")

TCP/IP standard

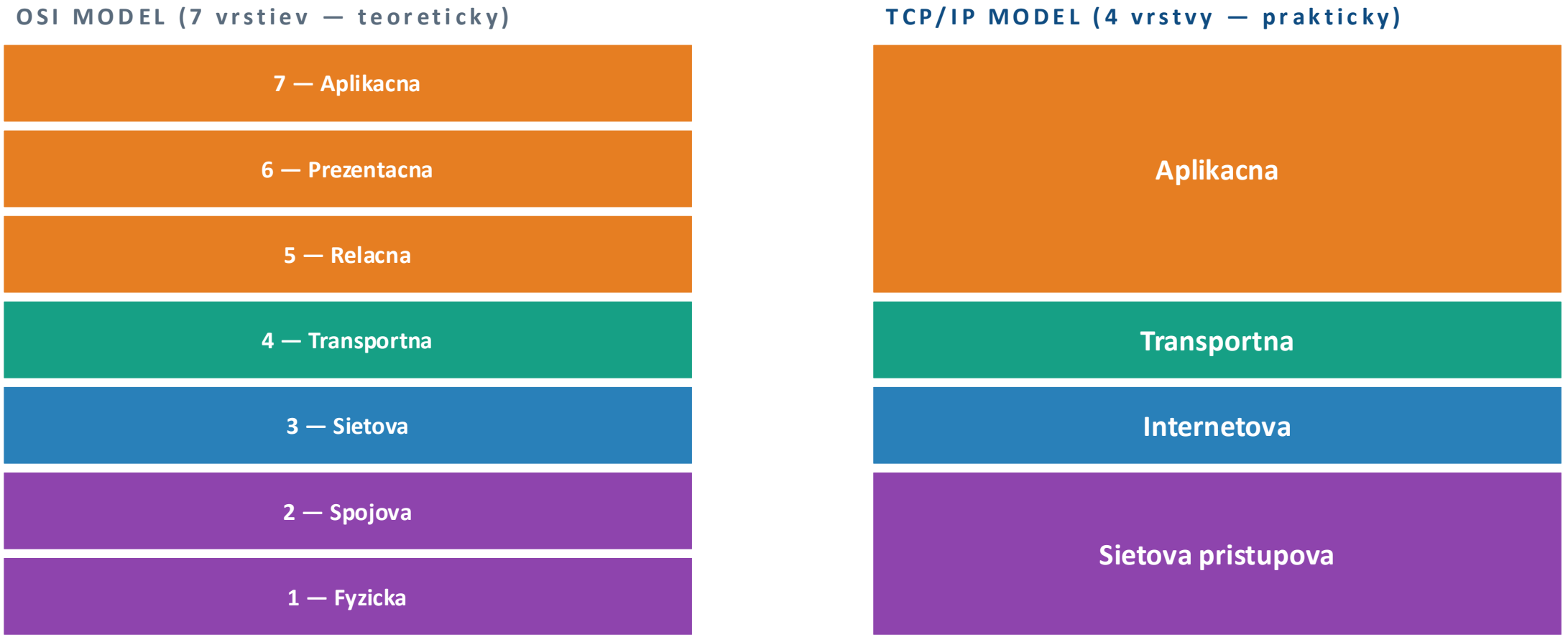
ARPANET prechadza z NCP na TCP/IP — oficialny zaciatok modernoeho internetu

Komercny internet

WWW (Tim Berners-Lee), HTTP, prehliadace TCP/IP sa stava globalnym standardom

***ZAUJIMAVOST**: TCP/IP nezacal ako akademicky model — vznikol z praktickej potreby prepojit roznorode vojenske siete tak, aby ich nedokazal vyradit jeden zasah.*

TCP/IP vs OSI — dva pohlady na to iste



V praxi pouzivame TCP/IP — je jednoduchsi a presne odraza realne implementacie. OSI sluzi hlavne ako vyukovy a referencny model.

Styri vrstvy TCP/IP — celkovy prehlad

4	**Aplikacna** *Sluzby pre uzivatela*	PROTOKOLY HTTP, HTTPS, DNS, SMTP, FTP, SSH	DATOVA JEDNOTKA **Data**
3	**Transportna** *Spolahlivy prenos medzi procesmi*	PROTOKOLY TCP, UDP	DATOVA JEDNOTKA **Segment / Datagram**
2	**Internetova** *Logicke adresovanie a smerovanie*	PROTOKOLY IP, ICMP, ARP	DATOVA JEDNOTKA **Paket**
1	**Sietova pristupova** *Fyzicky prenos po medii*	PROTOKOLY Ethernet, Wi-Fi (802.11), PPP	DATOVA JEDNOTKA **Ramec (frame)**

Vrstva 1: Sietova pristupova

UCEL

Fyzicky prenos dat po medii

Tato vrstva sa stara o to, ako sa data dostanu z jedneho zariadenia do druheho cez fyzicke spojenie. **Definuje signaly, kabelaz, MAC adresy a pristup k mediu**.

KLUC OVE OBLASTI

- **Fyzicke medium —** kabel, optika, radiove vlny
- **Kodovanie signalu —** ako sa 0 a 1 prenasaju ako napatie/svetlo
- **MAC adresy —** jedinecne hardware adresy zariadeni
- **Ramcovanie (frames) —** balenie dat do ramcov so suctom kontrol

Protokoly L1 a MAC adresy

NAJCASTEJSIE PROTOKOLY

Ethernet (IEEE 802.3)

Drotove pripojenie cez UTP/optiku.
Rychlosti: 10 Mb/s — 400 Gb/s.

Wi-Fi (IEEE 802.11)

Bezdrotove siete (a/b/g/n/ac/ax).
Dnes Wi-Fi 6 az 9.6 Gb/s.

PPP

Point-to-Point Protocol — modemove a seriove linky, VPN tunelovanie.

MAC ADRESA

Media Access Control

48-bitovy jedinecny identifikator sietoveho rozhrania, vypaleny vyrobcom v hardvery.

00:1B:44:11:3A:B7

00:1B:44	11:3A:B7
OUI — vyrobca	*seriove cislo*

- Funguje len v ramci jednej lokalnej siete (LAN)
- Switche pouzivaju MAC tabulku na presmerovanie ramcov
- Da sa zmenit softverovo (MAC spoofing)

Praktika: ipconfig (Windows) / ifconfig (Linux, Mac)

CO TO ROBI?

Zobrazi nastavenia sietovych rozhrani — IP adresu, MAC adresu, masku, branu (gateway).

PRIKAZ

```
ipconfig /all
```

```
ipconfig /release
```

```
ipconfig /renew
```

CO HLADAME

- **Physical Address (MAC)**
- **IPv4 Address (IP)**
- Subnet Mask (rozsah siete)
- Default Gateway (smerovac)
- DNS Servers (preklad mien)

cmd.exe

```
C:\Users\student> ipconfig /all

Windows IP Configuration

Ethernet adapter Ethernet:

   Connection-specific DNS Suffix . : local
   Description . . . . . . . . . . . : Intel(R) I219-V
   Physical Address. . . . . . . . . : 00-1B-44-11-3A-B7
   DHCP Enabled. . . . . . . . . . . : Yes
   IPv4 Address. . . . . . . . . . . : 192.168.1.42(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.1.1
   DNS Servers . . . . . . . . . . . : 8.8.8.8
                                       1.1.1.1

Wireless LAN adapter Wi-Fi:
   Media State . . . . . . . . . . . : Media disconnected
```

Vrstva 2: Internetova

UCEL

Logicke adresovanie a smerovanie

Tato vrstva sa stara o to, aby sa data dostali zo zariadenia A do zariadenia B aj cez viacero sieti. **Pridava IP adresy a rozhoduje o najlepsej ceste paketu**.

KLUC OVE ULOHY

- **Adresovanie —** kazde zariadenie ma jedinecnu IP adresu
- **Smerovanie (routing) —** ako sa paket dostane k cielu cez routery
- **Fragmentacia —** delenie velkych paketov, ked je linka mensia
- **Diagnostika —** ICMP — chybove a kontrolne spravy

IP adresy: IPv4 vs IPv6

IPv4

32 bitov • ~4.3 mld adries

```
192.168.1.42
```

4 oktety po 8 bitov, oddelene bodkami

- Najpouzivanejsi format dnes
- Adresny priestor sa vycerpava
- Verejne vs. privatne adresy (NAT)
- Priklad: 10.0.0.0, 172.16.0.0, 192.168.0.0 — privatne

IPv6

128 bitov • ~340 sextilionov adries

```
2001:0db8:85a3::8a2e:0370:7334
```

8 skupin po 16 bitov v hexadecimal, oddelene :

- Buducnost — nastupca IPv4
- Obrovsky adresny priestor
- Vstavana bezpecnost (IPsec)
- Bez potreby NAT

Protokoly internetovej vrstvy

IP

Internet Protocol

Hlavny protokol — zabezpecuje doruc enie paketov medzi sietami pomocou IP adries.

- Connectionless (bez spojenia)
- Best-effort (negarantuje dorucenie)
- Spravuje fragmentaciu

ICMP

Internet Control Message Protocol

Diagnosticky protokol — chybove hlasenia a stavove informacie.

- Pouziva sa pri ping (echo request/reply)
- Hlasenia 'Destination unreachable'
- TTL exceeded — pre tracert

ARP

Address Resolution Protocol

Preklada IP adresu na MAC adresu v lokalnej sieti.

- Funguje len v ramci LAN
- Pouziva broadcast otazky
- Cache udrziava nedavne preklady

Praktika: ping — overovanie dostupnosti

AKO TO FUNGUJE

Posle ICMP Echo Request a caka na Echo Reply. Meria cas a stratovost paketov.

```
ping google.com
```

CO POVIE VYSTUP

- Cas odpovede (latencia v ms)
- TTL — Time To Live
- Velkost paketu (bytes)
- Stratovost paketov
- DNS preklad domeny na IP

bash

```
$ ping google.com

PING google.com (142.250.184.46): 56 data bytes
64 bytes from 142.250.184.46: icmp_seq=0 ttl=117 time=12.4 ms
64 bytes from 142.250.184.46: icmp_seq=1 ttl=117 time=11.8 ms
64 bytes from 142.250.184.46: icmp_seq=2 ttl=117 time=14.2 ms
64 bytes from 142.250.184.46: icmp_seq=3 ttl=117 time=12.1 ms

--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss
round-trip min/avg/max/stddev = 11.8/12.6/14.2/0.9 ms

# Co tu vidime:
# - 142.250.184.46 = IP adresa Google
# - ttl=117 = pocet 'skokov' co este moze paket urobit
# - time=12.4 ms = ako rychlo sa paket vratil
```

Praktika: tracert — cesta paketu

AKO TO FUNGUJE

Posiela pakety s rastucim TTL (1, 2, 3...). Kazdy router cestou vrati 'TTL exceeded' — tak vidime mapu cesty.

```
tracert google.com
```

(Linux/Mac: traceroute google.com)

KEDY POMAHA

- Diagnostika problemov so spojenim
- Identifikacia, kde data 'visne'
- Zobrazi pripadne firewally a NAT

bash

```
$ traceroute google.com

traceroute to google.com (142.250.184.46), 30 hops max
 1  router.local (192.168.1.1)        1.2 ms  1.1 ms
 2  10.0.0.1                          5.4 ms  5.6 ms
 3  isp-gw.provider.sk (85.135.4.1)   8.1 ms  8.0 ms
 4  ba-core1.provider.sk              9.5 ms  9.3 ms
 5  prague-ix.peering.net            14.2 ms 14.0 ms
 6  google-peer.frankfurt            18.7 ms 18.5 ms
 7  72.14.232.1                      21.4 ms 21.2 ms
 8  142.250.184.46                   22.1 ms 22.0 ms

# Vidime kazdy router (hop) na ceste k cielu
# Stupajuce hodnoty time = vacsia geograficka vzdialenost
# 8 skokov stacilo — paket presiel 3 krajinami
```

Vrstva 3: Transportna

UCEL

Komunikacia medzi procesmi

Spaja konkretnu aplikaciu na jednom pocitaci s konkretnou aplikaciou na druhom. **Pridava porty a podla protokolu zaisti spolahlivost alebo rychlost**.

KLUC OVE FUNKCIE

- **Porty —** identifikuju konkretnu aplikaciu (HTTP=80, HTTPS=443)
- **Segmentacia —** delenie velkych dat na zvladnutelne kuski
- **Spolahlivost (TCP) —** potvrdzovanie, retransmisia, kontrola chyb
- **Rychlost (UDP) —** minimalne reziie pre real-time komunikaciu

TCP vs UDP — dva pristupy k transportu

TCP

Transmission Control Protocol

"Spolahlivy listonos"

Spojenie	ano (3-way handshake)
Spolahlivost	garantuje dorucenie
Poradie	zachovava poradie
Rychlost	pomalsie (vacsia rezia)
Pouzitie	web, email, SSH, file transfer

UDP

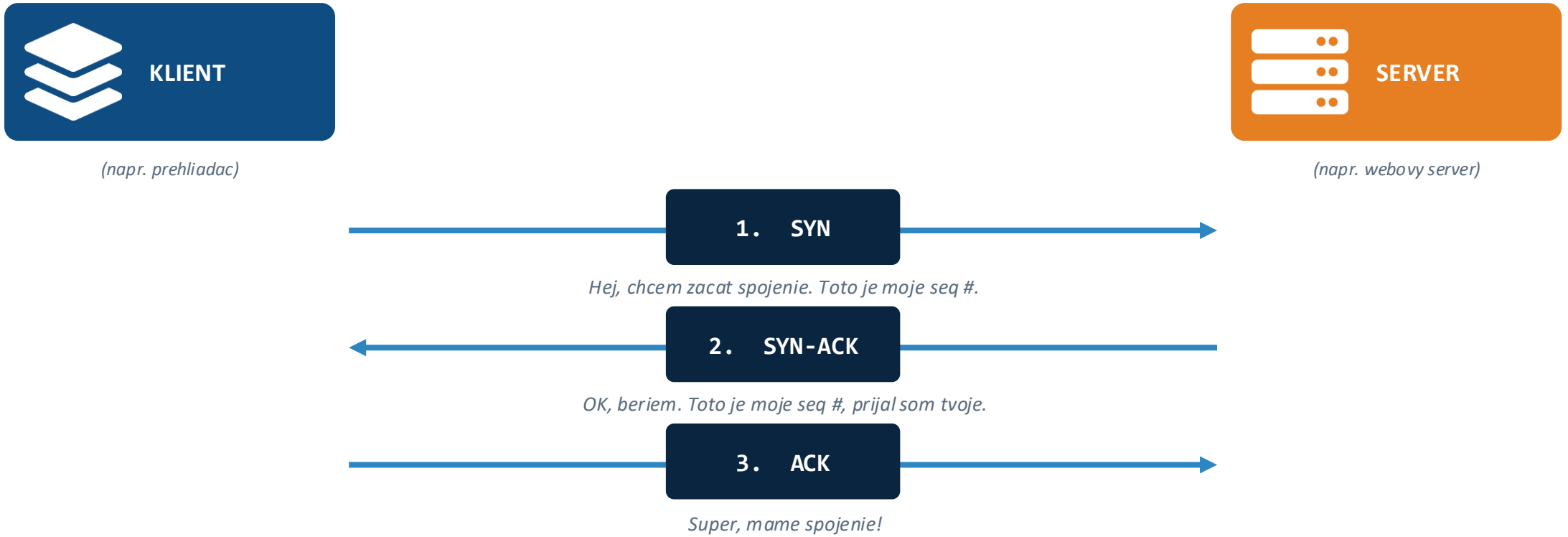
User Datagram Protocol

"Rychly leták"

Spojenie	nie (connectionless)
Spolahlivost	negarantuje dorucenie
Poradie	nezachovava
Rychlost	rychle (mala rezia)
Pouzitie	DNS, video stream, hry, VoIP

TCP three-way handshake

Skor ako sa zacnu prenasat data, klient a server si vymenia 3 spravy aby si potvrdili spojenie:



Po handshakou nasleduje spolahlivy prenos dat; spojenie sa potom uzatvara cez FIN/ACK (4-way close).

Porty — adresy aplikacii

CO JE PORT?

16-bitove cislo, ktore identifikuje konkretnu aplikaciu na zariadeni.

ROZSAH

- 0–1023 well-known
- 1024–49151 registered
- 49152–65535 dynamic

ANALOGIA

IP adresa je dom. Port je byt v dome — kazda aplikacia ma svoj.

PORT	PROTOKOL	POPIS
20/21	FTP	Prenos suborov
22	SSH	Bezpecny vzdialeny terminal
25	SMTP	Odosielanie emailov
53	DNS	Preklad mien
80	HTTP	Web (nesifrovany)
110	POP3	Stahovanie emailov
143	IMAP	Synchronizacia mailov
443	HTTPS	Web (sifrovany)
3389	RDP	Vzdialena plocha (Windows)

Praktika: netstat — kto je pripojeny?

CO TO ROBI

Vypise vsetky aktivne sietov e spojenia, otvorene porty a stav (LISTEN, ESTABLISHED, TIME_WAIT).

```
netstat -an
```

STAVY

- LISTEN — server caka na klienta
- ESTABLISHED — aktivne spojenie
- TIME_WAIT — spojenie sa zatvara
- CLOSED — uzatvorene

bash

```
$ netstat -an | head -15

Proto Local Address          Foreign Address        State
TCP   0.0.0.0:22             0.0.0.0:0              LISTEN
TCP   0.0.0.0:80             0.0.0.0:0              LISTEN
TCP   0.0.0.0:443            0.0.0.0:0              LISTEN
TCP   192.168.1.42:51234     142.250.184.46:443     ESTABLISHED
TCP   192.168.1.42:51235     140.82.112.4:443       ESTABLISHED
TCP   192.168.1.42:51236     52.84.150.39:443       ESTABLISHED
TCP   192.168.1.42:51201     34.102.136.180:443     TIME_WAIT
UDP   0.0.0.0:53             *:*
UDP   0.0.0.0:5353           *:*

# Local Address — moja IP a port
# Foreign Address — IP a port druhej strany
# Mam otvorene spojenia s Google, GitHub, AWS...
```

Vrstva 4: Aplikacna

UCEL

Sluzby pre uzivatela

Vrstva, s ktorou priamo pracuju aplikacie a uzivatel. **Definuje, akym jazykom sa rozpravaju klient a server (HTTP, SMTP, DNS...).**

CO SEM PATRI

- **Web —** HTTP / HTTPS — komunikacia prehliadaca a serveru
- **Email —** SMTP (odoslanie), POP3 / IMAP (prijem)
- **DNS —** preklad domenovych mien na IP adresy
- **Subory a vzdialeny pristup —** FTP, SSH, SCP, RDP

Aplikacne protokoly v praxi

HTTP

port 80

HyperText Transfer

Zaklad webu — ziadosti GET/POST

HTTPS

port 443

HTTP + TLS

Sifrovany web (TLS/SSL)

DNS

port 53

Domain Name System

Preklad google.com → IP

SMTP

port 25

Simple Mail Transfer

Odosielanie e-mailov

FTP

port 21

File Transfer

Stary protokol na prenos suborov

SSH

port 22

Secure Shell

Bezpecna vzdialena konzola

Praktika: nslookup a curl

nslookup — preklad mien na IP

bash — DNS

```
$ nslookup google.com

Server:    8.8.8.8
Address:   8.8.8.8#53

Non-authoritative answer:
Name:    google.com
Address: 142.250.184.46
Address: 142.250.184.78

$ nslookup -type=MX google.com
google.com  mail exchanger = 10 smtp.google.com

# Pred kazdym requestom DNS najprv prelozi
# domenu na IP — bez tohto by web nefungoval
```

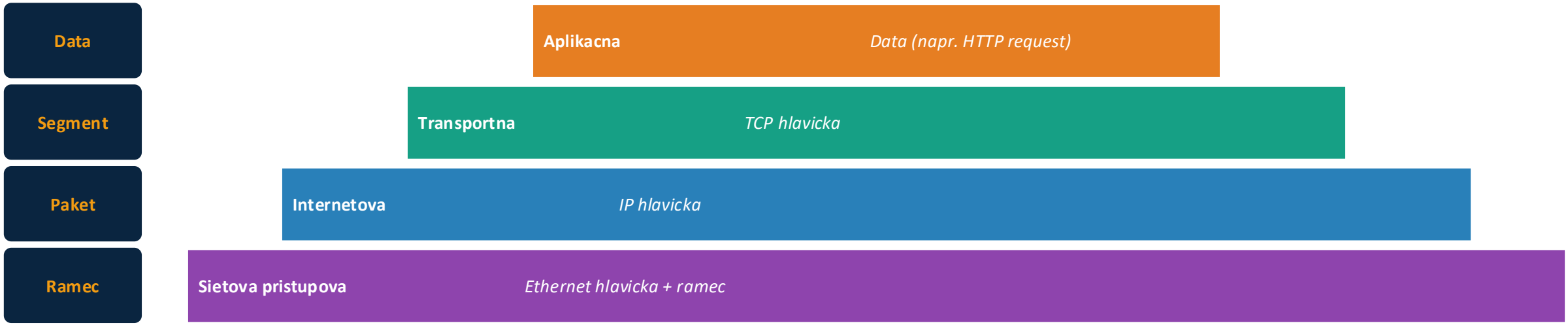
curl — surovy HTTP request

bash — HTTP

```
$ curl -v https://example.com

* Trying 93.184.216.34:443...
* Connected to example.com
* TLS handshake completed (TLS 1.3)

> GET / HTTP/2
> Host: example.com
> User-Agent: curl/8.4.0

< HTTP/2 200
< content-type: text/html
< server: ECS (dcb/7F84)

<!doctype html>
<html><head><title>Example</title>...
```

Encapsulacia — ako data putuju cez vrstvy

Kazda vrstva pridava svoju 'hlavicku' okolo dat. Na strane prijemcu sa hlavicky postupne odlupuju.



Odosielatel: kazda vrstva pridava svoju hlavicku

Prijemca: hlavicky sa postupne odlupuju

Tomuto procesu sa hovori 'pushovanie obalkou' — predstav si list, ktory dostane obalku, na nu nalepku, na to balicek a nakoniec ide do auta.

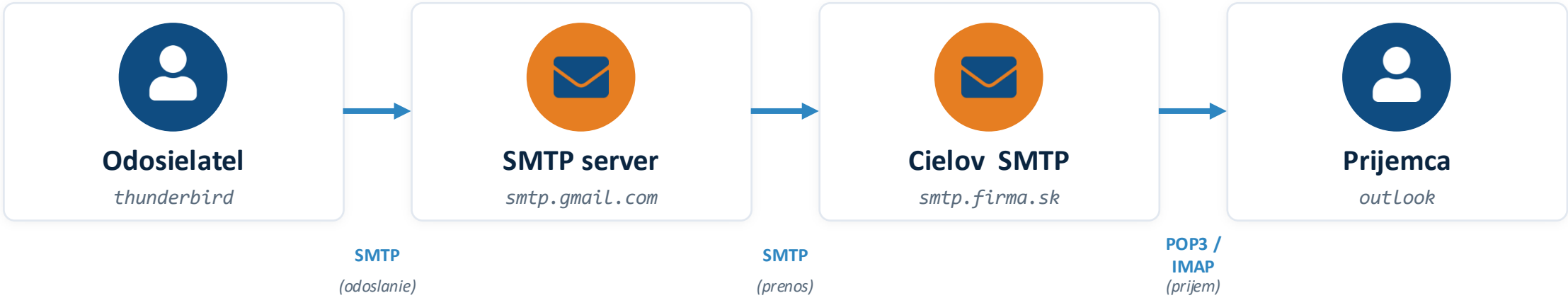
Co sa stane, ked napises 'google.com'?

1	**Aplikacna**	Prehliadac chce nacitat https://google.com
2	**DNS**	Najprv sa pyta DNS: 'Aku ma google.com IP?' → 142.250.184.46
3	**Transportna**	Otvori sa TCP spojenie na port 443 (3-way handshake)
4	**TLS**	Vymeni sa TLS certifikat — sifrovany kanal
5	**HTTP request**	Posle sa GET / HTTP/2 — server vrati HTML
6	**IP routing**	Pakety putuju cez routery az k Googlu (cez 8-15 hopov)
7	**Fyzicke**	Vsetko prebieha cez Wi-Fi, optiku, podmorske kable...

Cele to obvykle trva 100–500 ms

Realny priklad: Posielanie emailu

Email putuje viacerymi servermi a pouziva 3 rozne protokoly:



SMTP (25/465/587)

Posielanie — push z klienta na server

POP3 (110/995)

Stiahne emaily a vymaze ich zo serveru

IMAP (143/993)

Synchronizuje — emaily ostavaju na serveri

Realny priklad: Video streaming (YouTube, Netflix)

PRECO NIE OBYCAJNE TCP ALE UDP?

Pri videu sa chce plynulost, nie 100% spravnost

Ked TCP strati paket, caka az ho dostane znova. Pri zivom prenose to znamena zamrznute video. UDP ide dalej a stratu ignoruje — klient ju ked mozne dopocita.

ALE NETFLIX A YOUTUBE POUZIVAJU TCP!

Lebo to nie je 'zivy' prenos — buffer 30 sekund vopred. Real-time aplikacie (Zoom, Twitch live, hry) pouzivaju UDP / WebRTC.

PROTOKOLY V STREAMINGU

HTTPS / HTTP/2

Netflix, YouTube — adaptive bitrate streaming

RTP / RTSP

Realtime Transport Protocol — IP kamery, VoIP

WebRTC

Peer-to-peer pre videohovory (Meet, Discord)

QUIC (UDP)

Novy protokol — Google, YouTube, Cloudflare

Wireshark — pozeranie do paketov

CO JE TO?

Najpouzivanejsi nastroj na zachytenie a analyzu sietove komunikacie. Free a open-source.

CO VIDIME VO WIRESHARK

- Vsetky pakety na nasom rozhrani v realnom case
- Filtre podla protokolu / portu / IP
- Detail kazdej vrstvy v jednom pakete
- Zachyt do .pcap suboru na neskorsiu analyzu
- Statistiky, grafy, IO toky

Wireshark — capture en0

Filter: `tcp.port == 443`

#	Time	Source	Dest	Proto	Info
1	0.000	192.168.1.42	142.250.184.46	TCP	SYN
2	0.012	142.250.184.46	192.168.1.42	TCP	SYN-ACK
3	0.013	192.168.1.42	142.250.184.46	TCP	ACK
4	0.020	192.168.1.42	142.250.184.46	TLS	ClientHi
5	0.034	142.250.184.46	192.168.1.42	TLS	ServerHi
6	0.082	192.168.1.42	142.250.184.46	HTTP	GET /
7	0.105	142.250.184.46	192.168.1.42	HTTP	200 OK

```
▶ Frame 6: 287 bytes on wire
▶ Ethernet II, Src: 00:1b:44...
▼ Internet Protocol Version 4
    Src: 192.168.1.42, Dst: 142.250.184.46
▼ Transmission Control Protocol, Src: 51234, Dst: 443
▼ Hypertext Transfer Protocol
```

Wireshark: jeden HTTP paket cez vsetky vrstvy

Tu vidno presne ako kazda vrstva pridala svoju hlavicku — a ako sa to vsetko zaroven prejavilo:

Sietova pristupova *Ethernet II*	Src MAC: 00:1b:44:11:3a:b7 Dst MAC: c8:69:cd:9c:2e:11 Type: IPv4 (0x0800)
Internetova *IPv4*	Src IP: 192.168.1.42 Dst IP: 142.250.184.46 TTL: 64, Protocol: TCP (6)
Transportna *TCP*	Src Port: 51234 Dst Port: 443 Flags: PSH, ACK Seq: 1024
Aplikacna *HTTP / TLS*	GET / HTTP/2 Host: google.com User-Agent: Mozilla/5.0...

Cvicenia pre vas

1 Mapa cesty

20 min

Pustite tracert na 5 roznych domen (google.com, github.com, mit.edu, baidu.com, abc.net.au). Porovnajte pocet hopov a casy. Co vidite?

2 Aktivne spojenia

15 min

Zistite, ktore aplikacie na vasom PC prave teraz komunikuju (netstat -an). Najdite spojenie s prehliadacom a urcte port serveru.

3 DNS detektiv

15 min

Pomocou nslookup zistite: A zaznamy pre vsku.sk, MX zaznamy pre gmail.com, NS zaznamy pre google.com. Co vam to povie?

4 Wireshark capture

30 min

Spustite Wireshark, otvorte http://example.com (nie HTTPS!) a najdite GET request. Ake hlavicky tam su?

Zhrnutie

Najdolezitejsie zo vsetkeho

01 Fyzicka	02 Internetova	03 Transportna	04 Aplikacna
Fyzicky prenos cez Ethernet, Wi-Fi. MAC adresy.	IP adresy a smerovanie. Protokoly IP, ICMP, ARP.	Porty + spolahlivost (TCP) alebo rychlost (UDP).	HTTP, DNS, SMTP — sluzby pre uzivatela.

Otazky?

CISCO sietove zariadenia — uvod

KTO JE CISCO?

Cisco Systems, Inc.

Najvacsi vyrobca sietovych zariadeni na svete (od 1984). Routery, switche, firewally, AP a software (IOS, NX-OS) — pohanaju vacsinu firemnych sieti, datacentier a chrbticu internetu.

PRECO TO STUDOVAT?

- CCNA / CCNP — najuznavanejsie certifikacie
- Pochopis principy aplikovatelne na vsetky vendory
- Vacsina firiem ma aspon nieco od Cisca

HLAVNE PRODUKTOVE RADY

SWITCHE

Catalyst (9200, 9300, 9500, 2900, 3650)

Enterprise LAN switche

ROUTERY

ISR / ASR (1100, 4000, 9000)

Pre pobocky aj operatorov

FIREWALLY

ASA / Firepower

Bezpecnost na perimetri

WI-FI

Aironet / Catalyst 9100

Bezdrotove pristupove body

HLAS

IP Phone (7800, 8800)

VoIP telefony pre Webex/CUCM

Router vs Switch — kluc ovy rozdiel

Najcastejsia otazka. Obe to su "krabicky s portami" — ale robia uplne ine veci:

SWITCH

"Spaja zariadenia v jednej sieti"

Vrstva	Vrstva 2 — Spojova / Internetova
Adresy	Pracuje s MAC adresami
Rozsah	Jedna LAN siet
Porty	Vela portov (24, 48, 96)
Domena	Ohranicuje collision domain
Priklad	Cisco Catalyst 9300-48P

ROUTER

"Spaja rozne siete medzi sebou"

Vrstva	Vrstva 3 — Sietova / Internetova
Adresy	Pracuje s IP adresami
Rozsah	Medzi LAN / WAN / internet
Porty	Malo portov (4–8)
Domena	Ohranicuje broadcast domain
Priklad	Cisco ISR 4451-X

Router vs Switch — kluc ovy rozdiel

Najcastejsia otazka. Obe to su "krabicky s portami" — ale robia uplne ine veci:

SWITCH

"Spaja zariadenia v jednej sieti"

ROUTER

"Spaja rozne siete medzi sebou"

Switch — Layer 2 zariadenie

AKO TO ROBI

MAC tabulka — mapa portov a adries

Switch sa uci sam — ked vidi ramec, zapamata si MAC adresu odosielatela a port, na ktorom prisiel. **Ked potom dostane ramec pre tu MAC, vie ho poslat priamo na spravny port**.

KLUC OVE FUNKCIE

- **MAC learning —** automaticky si plni MAC tabulku
- **Forwarding —** posiela ramec len na spravny port (nie broadcast)
- **VLAN —** logicke rozdelenie na samostatne siete
- **STP —** Spanning Tree zabranuje slucke v topologii
- **PoE —** napajanie cez ethernet (telefony, AP, kamery)

MAC TABULKA

```
Catalyst 9300# show mac-address-table
```

VLAN	MAC	TYP	PORT
10	0050.56be.1a4d	DYN	Gi1/0/1
10	0050.56be.7c2e	DYN	Gi1/0/3
20	00d0.bc12.a9e7	DYN	Gi1/0/5
20	5897.bdc0.f81b	DYN	Gi1/0/7
99	001b.4411.3ab7	STA	Gi1/0/12
10	ffff.ffff.ffff	BROADCAST	All

DYN = naucene dynamicky STA = staticke

Switch presne vie, kde ktora MAC "sedi".

Router — Layer 3 zariadenie

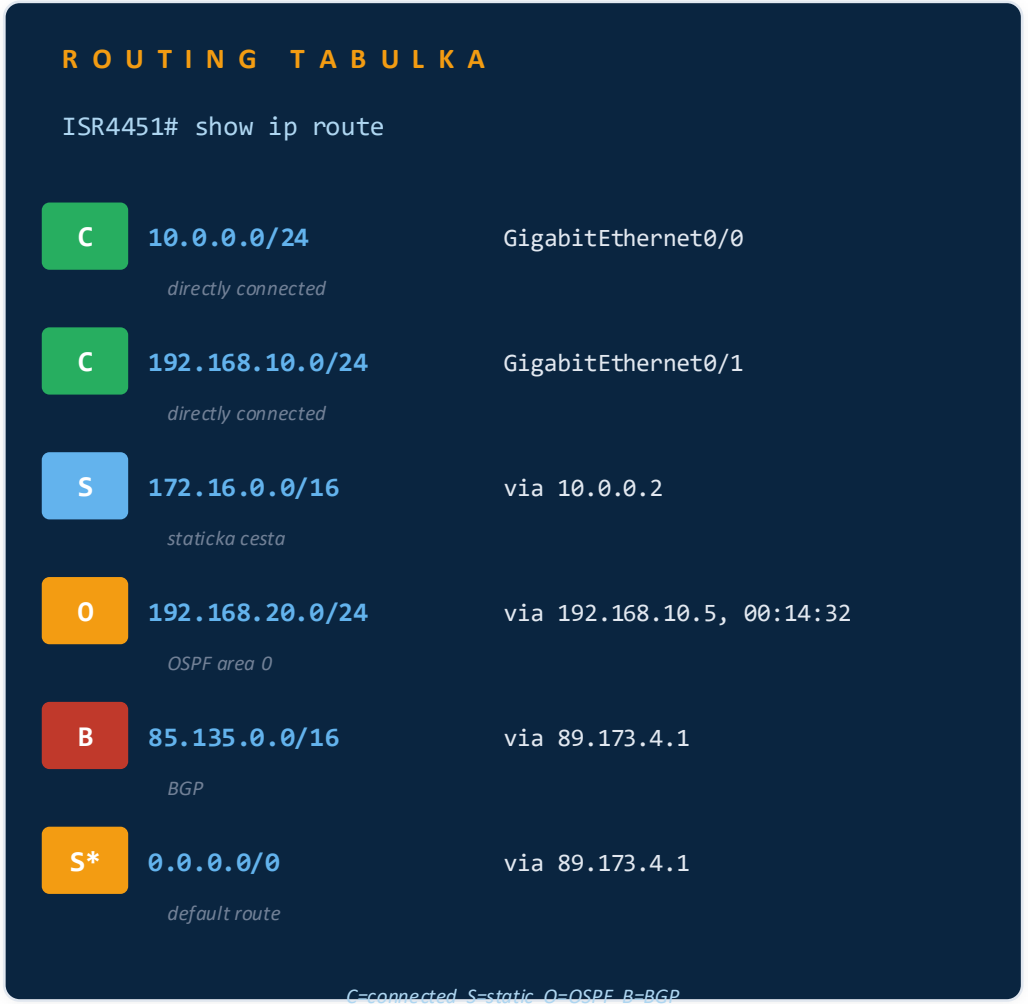
AKO TO ROBI

Routing tabulka — mapa sieti

Router pre kazdy paket pozrie cielovu IP, **najde najlepsiu cestu v routing tabulke a posle ho na konkretne rozhranie**. Cestu sa moze naucit staticky (admin) alebo dynamicky (RIP, OSPF, EIGRP, BGP).

KLUC OVE FUNKCIE

- **IP routing —** rozhoduje o ceste paketu medzi sietami
- **NAT —** preklada privatne IP na verejne (1 link, vela klientov)
- **DHCP —** moze rozdavat IP adresy klientom
- **ACL / firewall —** filtruje pakety podla pravidiel
- **VPN —** sifrovany tunel medzi pobockami



Fixne vs Modularne zariadenia

Pri vybere routera/switchu rozhoduje, ci portove rozhrania mozes neskor menit alebo doplnat:

FIXNE (fixed)

Plug-and-play. Porty su pevne dane.

CHARAKTERISTIKA

- Pevne dany pocet a typ portov (24× GE, 48× GE...)
- Lacnejsie, mensie, nizsia spotreba
- Rychla instalacia — vybalit a zapojit
- Bez moznosti rozsirit nove rozhrania

TYPICKE PRIKLADY

- Catalyst 9200-24P (24-portovy switch)
- ISR 1100 — pobockovy router
- Meraki MS / MR — cloud-managed

MODULARNE (modular)

Sasi + slot y. Konfiguracia podla potreby.

CHARAKTERISTIKA

- Sasi (chassis) so slotmi pre line cards / SFP
- Vymenitelne porty: medene, optika, 1G/10G/100G
- Zalozne napajacie zdroje, hot-swap
- Drahsie, ale skalovatelne na roky dopredu

TYPICKE PRIKLADY

- Catalyst 9400/9500 — modularny switch
- ASR 9000 — operatorske routery
- Nexus 9500 — chrbtica datacentra

Cisco IOS — zakladne prikazy

CO JE IOS?

Internetwork Operating System — operacny system, ktory bezi na vsetkych Cisco zariadeniach. **Konfiguruje sa cez prikazovy riadok (CLI) cez konzolu, SSH alebo Telnet**.

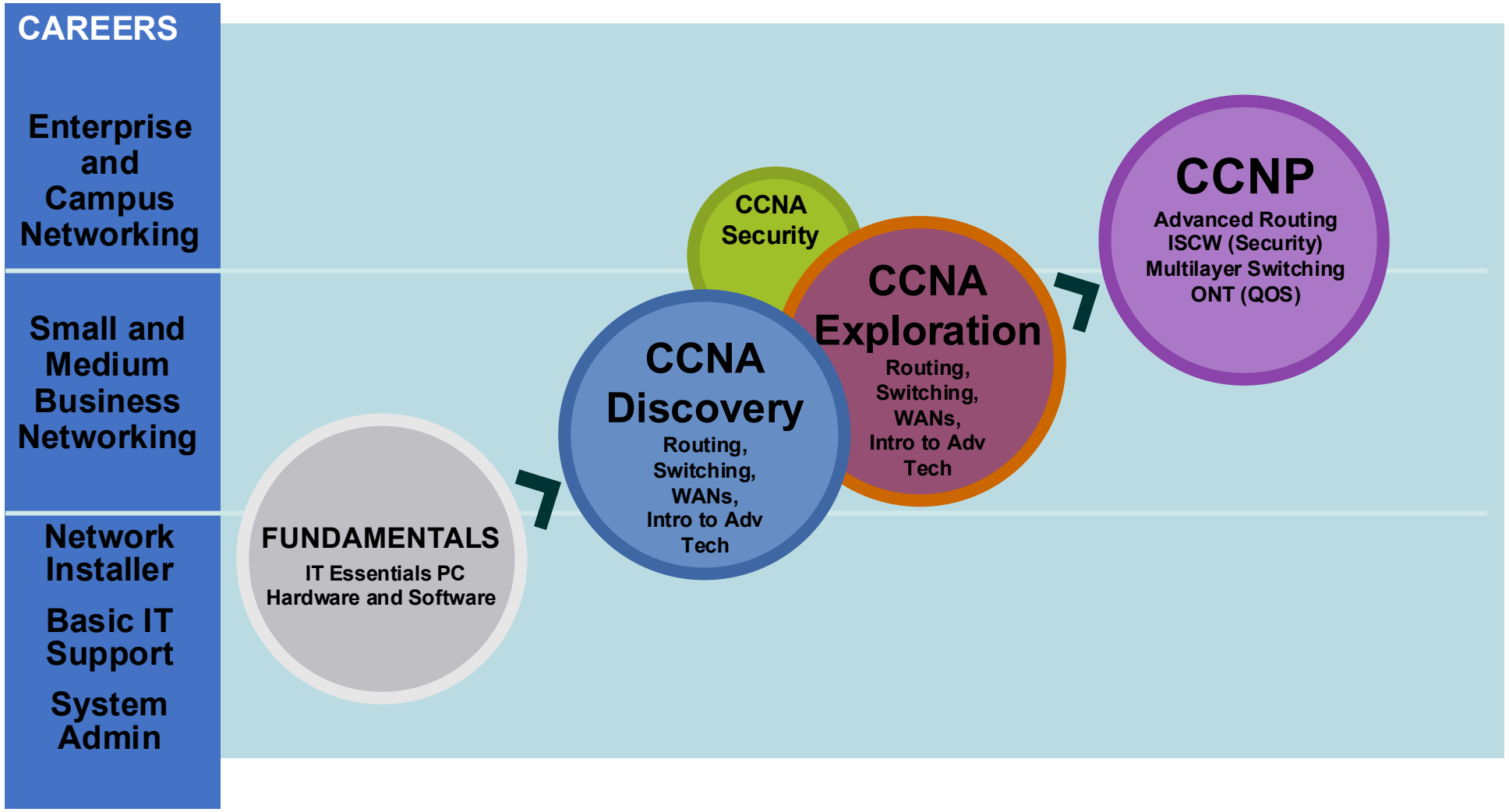
REZIMY (PROMPT)

`Router>`	user EXEC — len prezeranie
`Router#`	privileged — diagnostika
`Router(config)#`	global config — zmeny
`Router(config-if)#`	interface config

console — Catalyst 9300

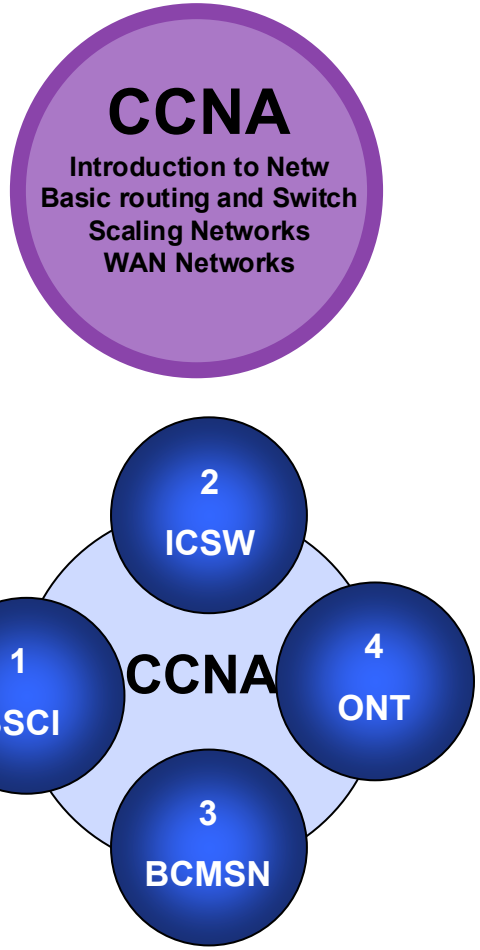
```
# === ZAKLADNA DIAGNOSTIKA ===
Router> enable
Router# show version
Cisco IOS Software, ISR Software (X86_64), Version 17.9
Router# show ip interface brief
Interface              IP-Address      OK?  Status     Protocol
GigabitEthernet0/0     10.0.0.1        YES  up         up
GigabitEthernet0/1     192.168.10.1    YES  up         up

# === KONFIGURACIA ===
Router# configure terminal
Router(config)# hostname R1-Bratislava
Router(config)# enable secret Cisco123!
Router(config)# interface GigabitEthernet0/0
Router(config-if)# ip address 10.0.0.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# end

# === ULOZENIE ===
Router# copy running-config startup-config
[OK]
```

Cisco Certified Network A & P

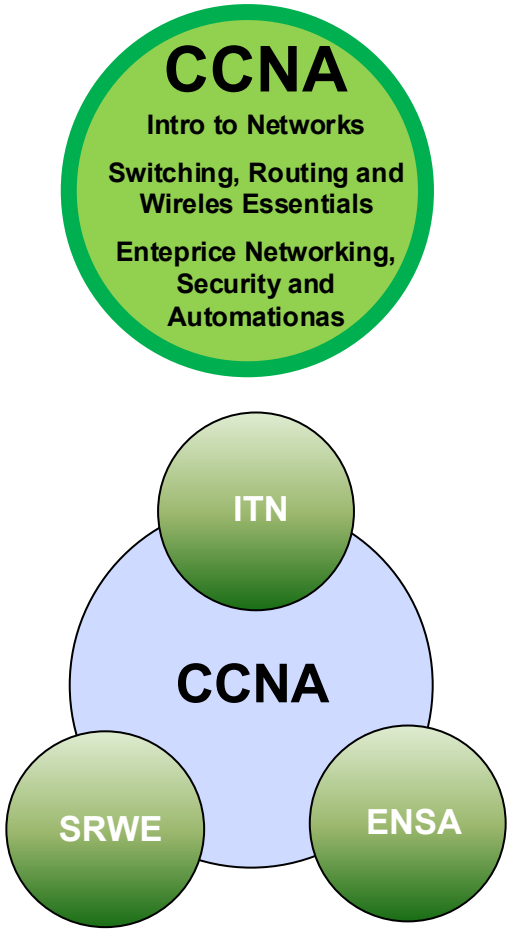


Možnosť certifikácie CCNA

Minulosť



Súčasnosť



Priemysel

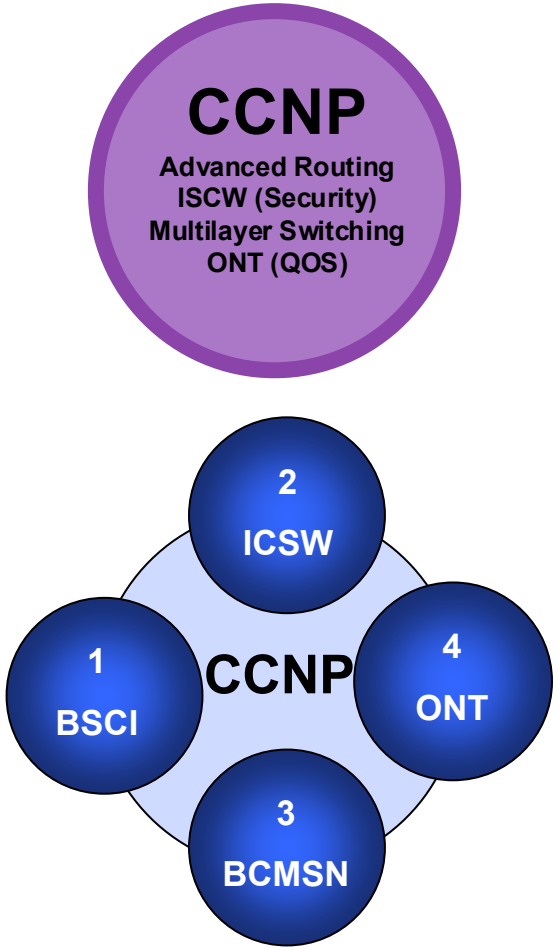
[300-101 ROUTE]

[642-813 SWITCH]

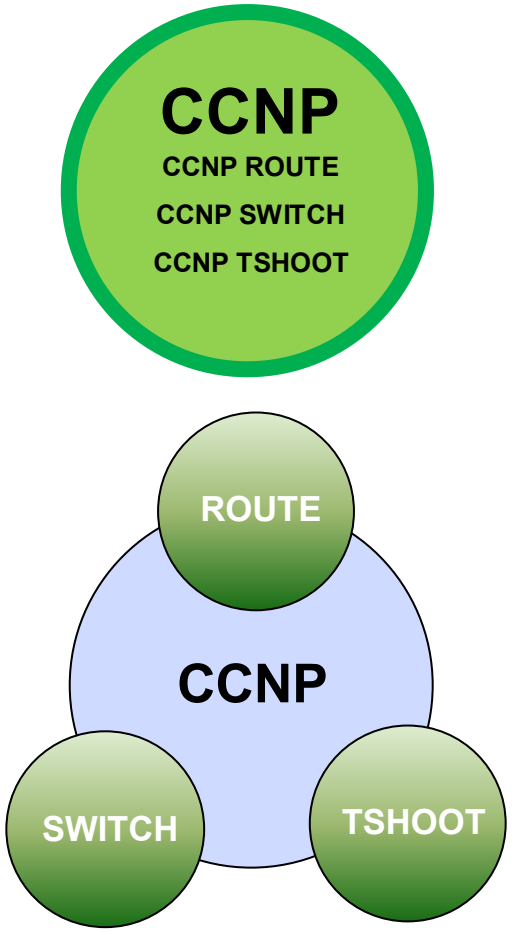
[642-832 TSHOOT]

Možnosť certifikácie CCNA

Minulosť



Súčasnosť



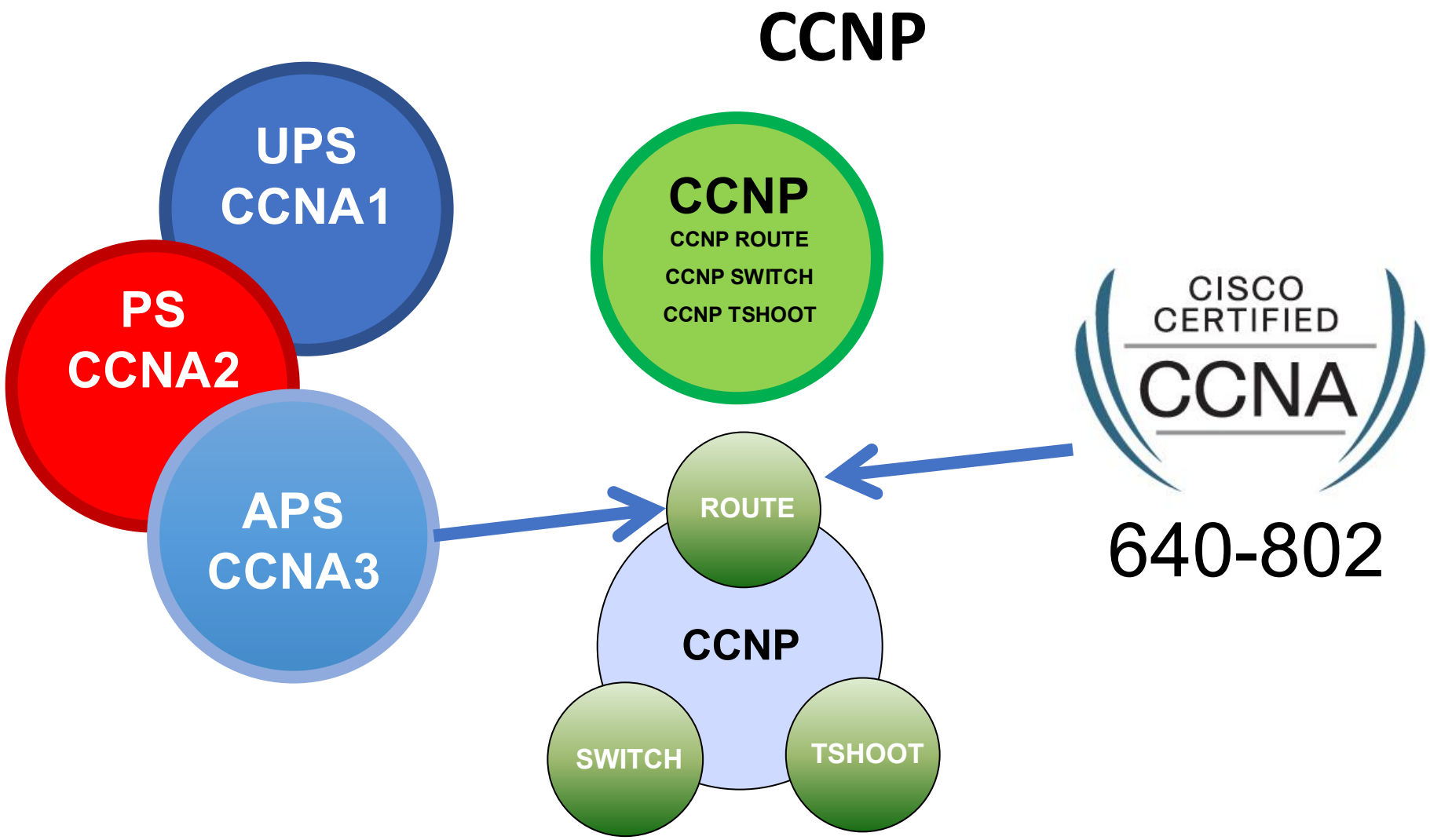
Priemysel

[300-101 ROUTE]

[642-813 SWITCH]

[642-832 TSHOOT]

Prerekvizity predmetov



Laboratórium počítačových sietí BN_32 – 514

Je orientované na systematickú a cielenú výučbu predmetov súvisiacich s návrhom a budovaním lokálnych počítačových sietí LAN na báze implementácie a realizácie základných konfigurácií aktívnych sieťových komponentov.

Laboratórium počítačových sietí

Laboratórium počítačových sietí

Laboratórium počítačových sietí

Obsahová náplň predmetov

CCNA v7 Course #1	CCNA v7 Course #2	CCNA v7 Course #3
Networking Today	Basic Device Configuration	Single-Area OSPFv2 Concepts
Basic Switch and End Device Configuration	Switching Concepts	Single-Area OSPFv2 Configuration
Protocol Models	VLANs	WAN Concepts
Physical Layer	Inter-VLAN Routing	Network Security Concepts
Number Systems	STP	ACL Concepts
Data Link Layer	Etherchannel	ACLs for IPv4 Configuration
Ethernet Switching	DHCPv4	NAT for IPv4
Network Layer	SLAAC and DHCPv6 Concepts	VPN and IPsec Concepts
Address Resolution	FHRP Concepts	QoS Concepts
Basic Router Configuration	LAN Security Concepts	Network Management
IPv4 Addressing	Switch Security Configuration	Network Design
IPv6 Addressing	WLAN Concepts	Network Troubleshooting
ICMP	WLAN Configuration	Network Virtualization
Transport Layer	Routing Concepts	Network Automation
Application Layer	IP Static Routing	
Network Security Fundamentals	Troubleshoot Static and Default Routes	
Build a Small Network		

New/significantly changed content

Dakujem za pozornost — na rade ste vy!

Otazky?

gabriel.bugar@tuke.sk