

Komunikacijski protokoli in omrežna varnost

Uvod in ponovitev osnov predmeta

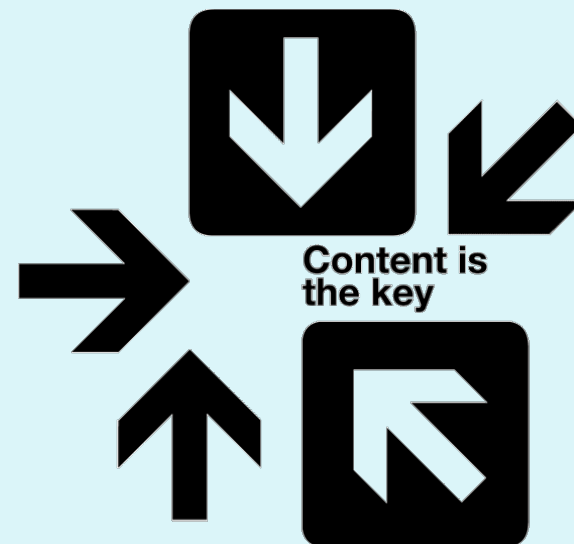
Komunikacijski protokoli in omrežna varnost

- **Profesor:**
dr. Andrej Brodnik
- **Asistent:**
as. Aleks Huč
as. dr. Gašper Fele Žorž
- **Izvedba predmeta:**
 - 3 ure predavanj - 2 dela, 2 uri laboratorijskih vaj tedensko
 - kontakt: e-mail, govorilne ure, forum na strani predmeta



Vsebina predmeta

- ponovitev osnov računalniških komunikacij (ISO/OSI, TCP/IP, protokoli, storitve, varnost),
- zagon stroja
- nadzor in upravljanje omrežij,
- razpošiljanje (*multicasting*),
- aplikacije v stvarnem času,
- varnost: overovljenje, avtorizacija, beleženje, varni prenosi, VPN, certificiranje, požarni zidovi, IDS sistemi,
- podatki za delovanje omrežja, LDAP,
- IEEE 802.



Vsebina predmeta - okvirni načrt

teden	predavanja	DN	LN	vaje
datum	#	# oddaja	oddaja	tema
03. 10. 2022	1 Uvod v predmet	1		
10. 10. 2022	2 Zagon stroja	1		VirtualBox, KPOV Judge
17. 10. 2022	3 Nadzor in upravljanje omrežij	1	27. 10. 2022	Nastavitve mreže
24. 10. 2022	4 Dan mrtvih			inetd
31. 10. 2022	5 Promet za aplikacije v stvarnem času	2		DHCP
07. 11. 2022	6 Razpošiljanje	2		NFS, zagon Linux
14. 11. 2022	7 Razpošiljanje	2	24. 11. 2022	SNMP
21. 11. 2022	8 Varnostni elementi omrežij	3		25. 11. 2022 Čas po mreži
28. 11. 2022	9 KOLOKVIJ 1			VLC
05. 12. 2022	10 Overovljenje, avtorizacija, beleženje (3		VPN – preprost
12. 12. 2022	11 Overovljenje, avtorizacija, beleženje (3, 4	3, 4	22. 12. 2022	VPN, CA
19. 12. 2022	12 Podatki za delovanje omrežja (LDAP)	4		Netfilter
26. 12. 2022	13 Družina IEEE 802	4	05. 01. 2023	LDAP, fusiondirectory
02. 01. 2023	14 <i>vabljen predavanje</i>			LDAP, PAM, nsswitch
09. 01. 2023	15 KOLOKVIJ 2		13. 01. 2023	Radius

Predavanja so ob torkih, datum pa je ponedeljek v tednu.
DN se tudi oddajajo v četrtek do polnoči.
LN se odda v petek do polnoči.

Obveznosti predmeta

Končna ocena (≥ 50):

• 4 domače naloge:	20%
• laboratorijski nalogi	40%
• <u>pisni izpit ali 2 kolokvija:</u>	40%
	100%

Obveznosti:

- domače naloge ≥ 40 , vsaka domača naloga ≥ 20
- laboratorijski nalogi ≥ 40 , vsaka laboratorijska naloga ≥ 20
- pisni izpit ≥ 50 , vsak od kolokvijev ≥ 40
- (*KPOV judge*)
- DNo in DNn

KPOV judge

Obrnjena učilnici:

- za (skoraj) vsake vaje je pripravljena predpriprava
- rešite in oddate preko spleta **pred** vajami
- oceni se samodejno

Obveznosti predmeta

Pri oceni se še upošteva:

- dopolnjevanje RFCjev
- sodelovanje na forumih
- pomoč kolegom
- priprava sledi protokolov
- ...

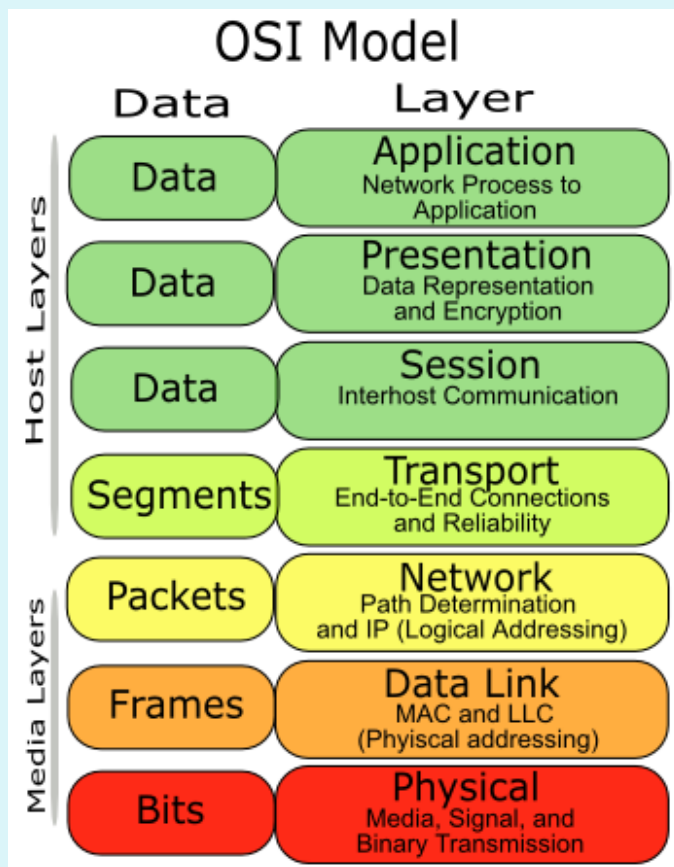
Literatura

- J. F. Kurose, K. W. Ross: Computer Networking, 5th edition, Addison-Wesley, 2010.
- A. Farrel: The Internet and Its Protocols: A Comparative Approach, Morgan Kaufmann, 2004.
- E. Cole: Network Security Bible, Wiley, 2nd edition, 2009.
- Mani Subramanian: Network Management: An introduction to principles and practice, Addison Wesley Longman, 2000
- RFCji
- ...

Ponovitev osnov računalniških komunikacij

ISO/OSI model

- model vsebuje 7 plasti, ki definirajo sloje sorodnih funkcij komunikacijskega sistema



aplikacijska plast

predstavitvena plast

sejna plast

transportna plast

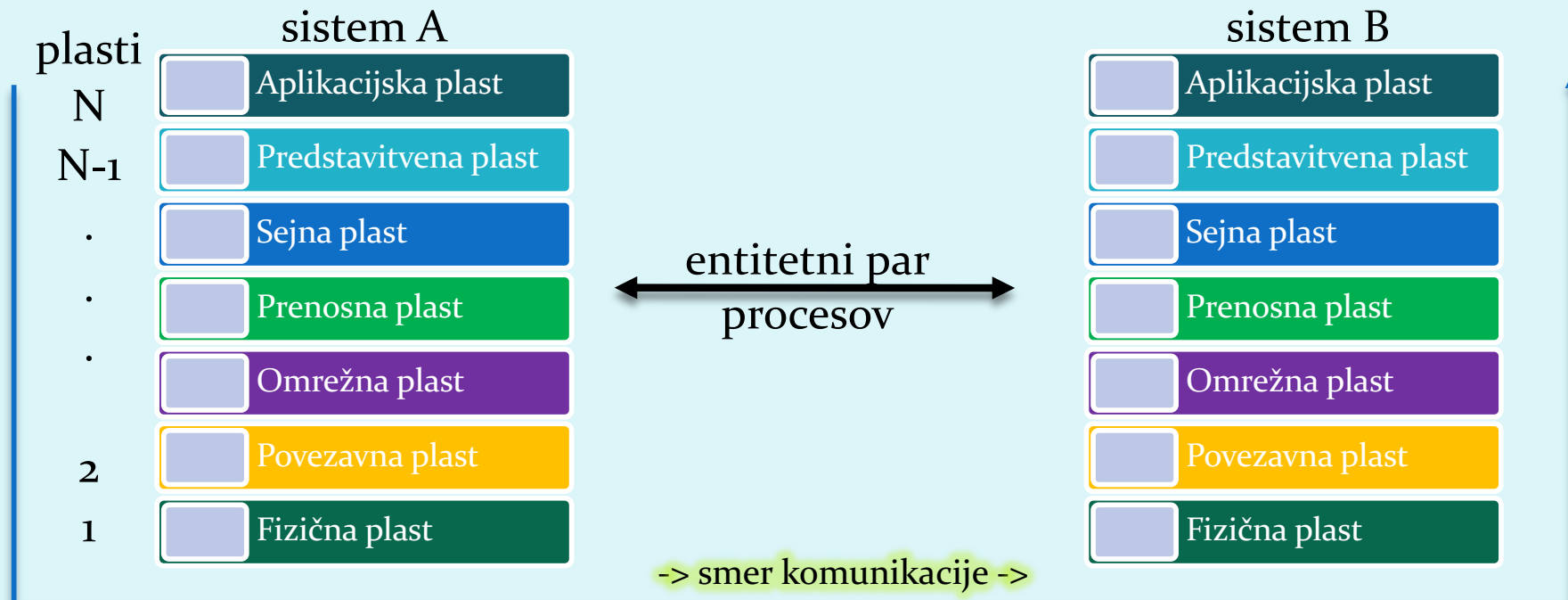
omrežna plast

povezavna plast

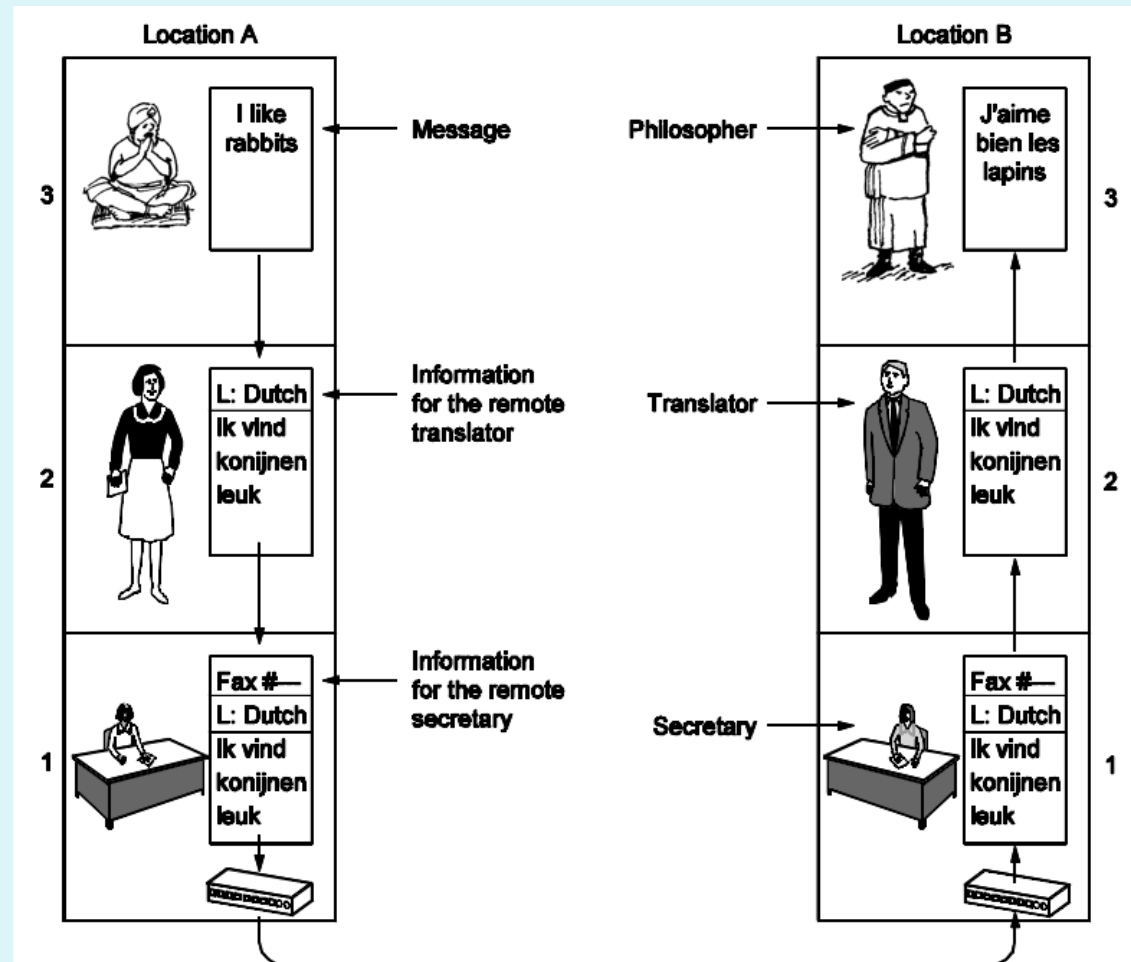
fizična plast

ISO/OSI model

- plast N nudi storitve (streže) plasti N+1
- plast N zahteva storitve (odjema) od plasti N-1,
- protokol: pravila komuniciranja med istoležnima procesoma,
- entitetni par: par procesov, ki komunicira na isti plasti



Analogija: pogovor med dvema filozofoma

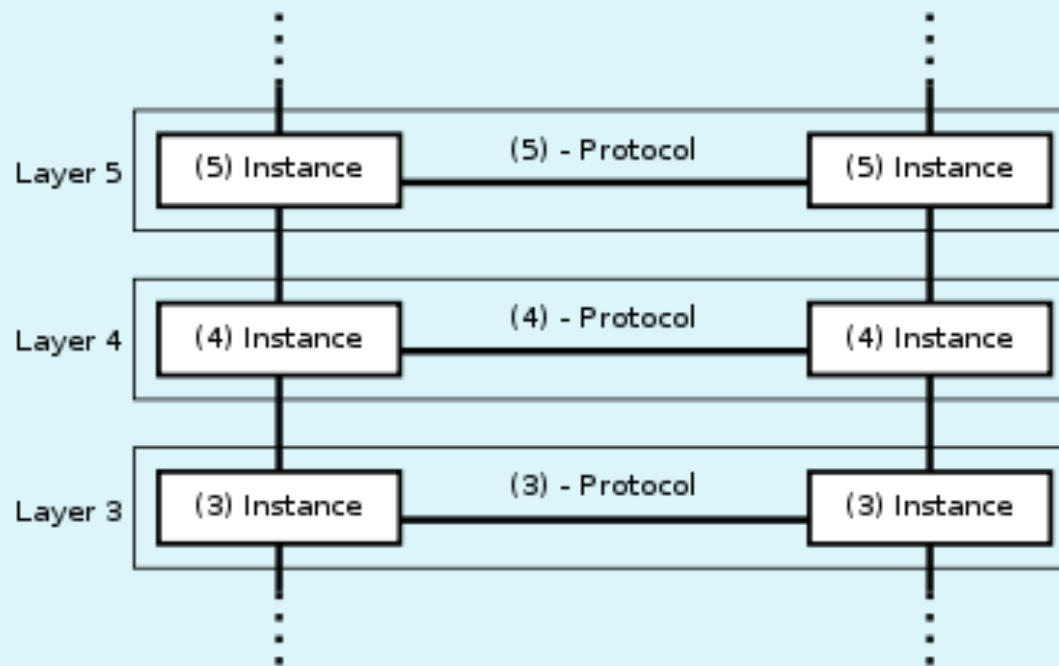


- Zakaj plasti?
 - sistematična zasnova zgradbe sistema,
 - sprememba implementacije dela sistema je neodvisna od ostalega sistema

ISO/OSI model

In še drugače:

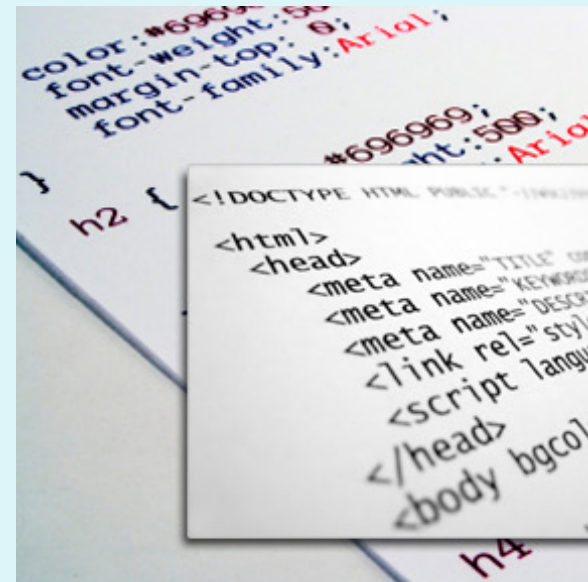
- vsaka plast ima svoje protokole (= jezik, s katerim se pogovarja istoležni entitetni par procesov),
- protokoli so specifični za storitve, ki jih plast zagotavlja.



OSI plasti: podrobneje

- **Aplikacijska plast**

- najbližja uporabniku,
- omogoča interakcijo aplikacije z omrežnimi storitvami,
- standardne storitve: telnet, FTP, SMTP, SNMP, HTTP



OSI plasti

- **Predstavitvena plast**

- določa pomen podatkov med entitetnima paroma aplikacijske plasti,
- sintaksa in semantika,
- določa kodiranje, kompresijo podatkov, varnostne mehanizme

- **Sejna plast**

- nadzor pogovora (množice povezav) med aplikacijama,
- logično povezovanje med aplikacijami,
- običajno vgrajena v aplikacije.

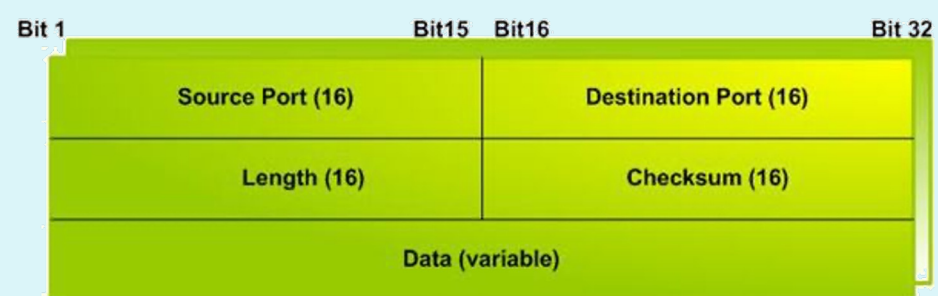
OSI plasti

- **Transportna plast** (enota: SEGMENT)
 - učinkovit, zanesljiv in transparenten prenos podatkov med uporabnikoma; te storitve zagotavlja višjim plastem,
 - mehanizmi: kontrola pretoka, segmentacija, kontrola napak,
 - povezavni, nepovezavni prenosi,
 - TCP, UDP, IPSec, GRE, L2TP, PPP

The TCP Segment Format

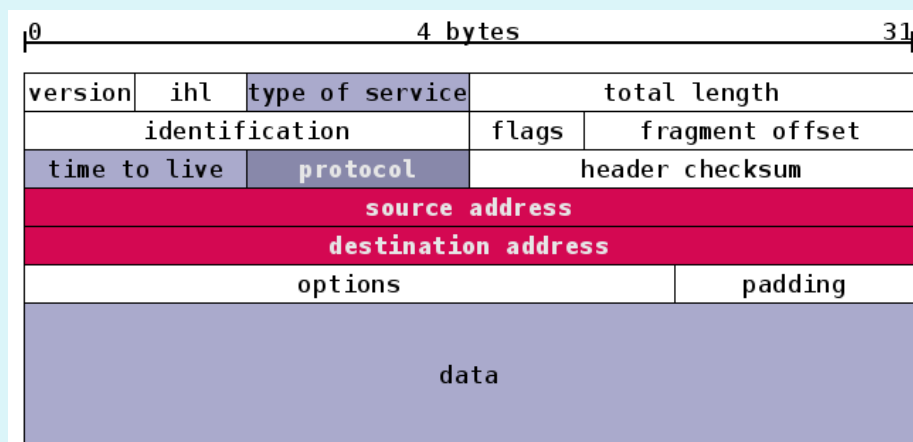


The UDP Segment Format



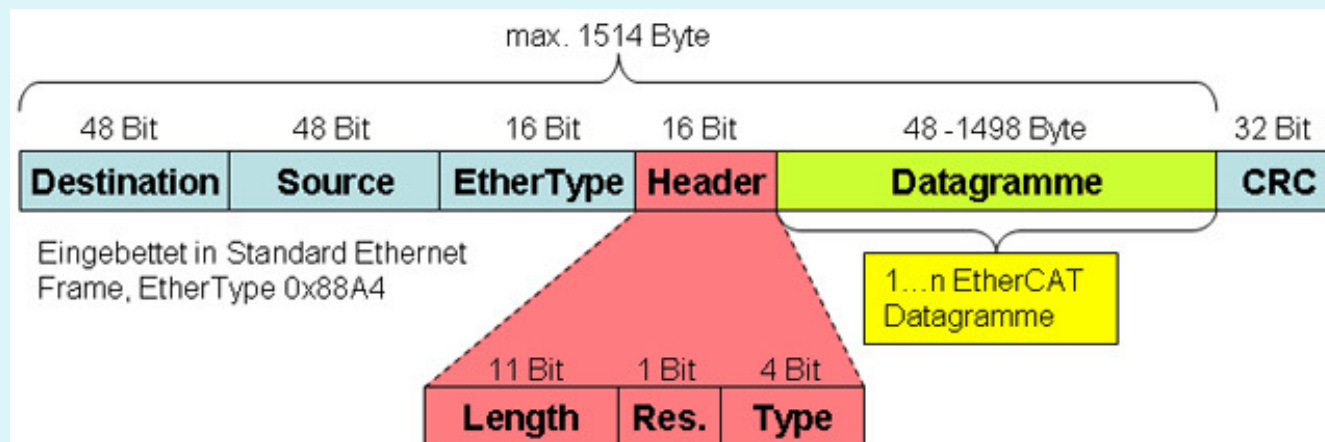
OSI plasti

- **Omrežna plast** (enota: PAKET)
 - usmerjanje (povezavne in nepovezavne storitve)
 - prenos paketov od izvornega do ciljnega računalnika,
 - lahko zagotavlja: zagotovljeno dostavo, pravilno zaporedje, fragmentacijo, izogibanje zamašitvam,
 - usmerjanje, usmerjevalniki, usmerjevalni algoritmi,
 - protokoli: IP, ICMP, IPSec, IGMP, IPX



OSI plasti

- **Povezavna plast** (enota: OKVIR)
 - asinhrona/sinhrona komunikacija,
 - fizično naslavljanje: npr MAC naslov,
 - zaznavanje in odpravljanje napak (pariteta, CRC, checksum)
 - kontrola pretoka, okvirjanje
 - protokoli: Ethernet, PPP, Frame Relay



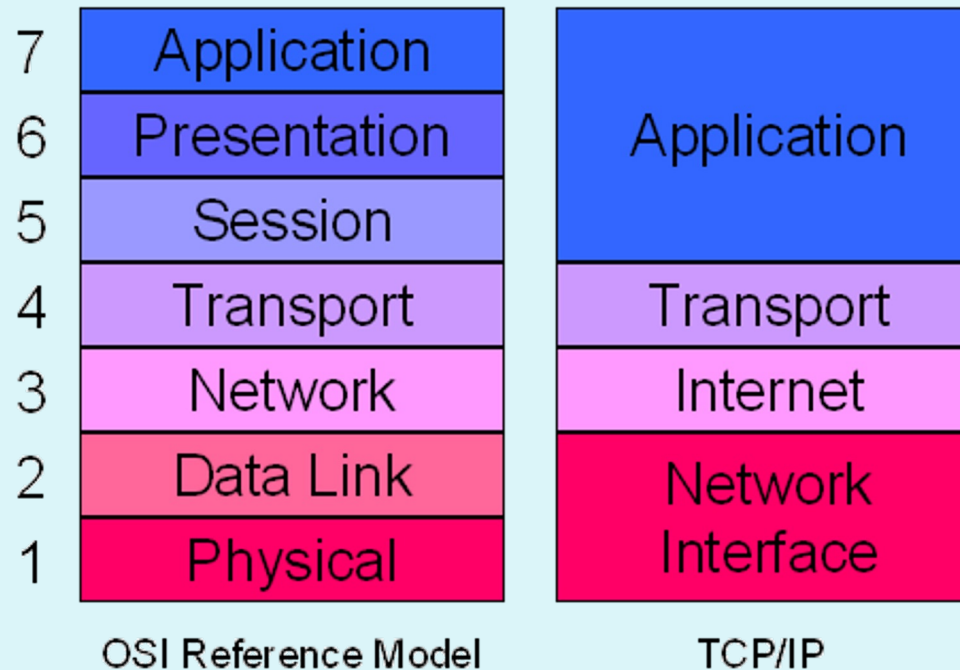
OSI plasti

- **Fizična plast**

- prenos bitov po kanalu (baker/optika/brezžično),
- digitalni, analogni medij,
- UTP, optika, koaksialni kabli, brezžična omrežja,
- RS-232, T1, E1, 802.11b/g, USB, Bluetooth



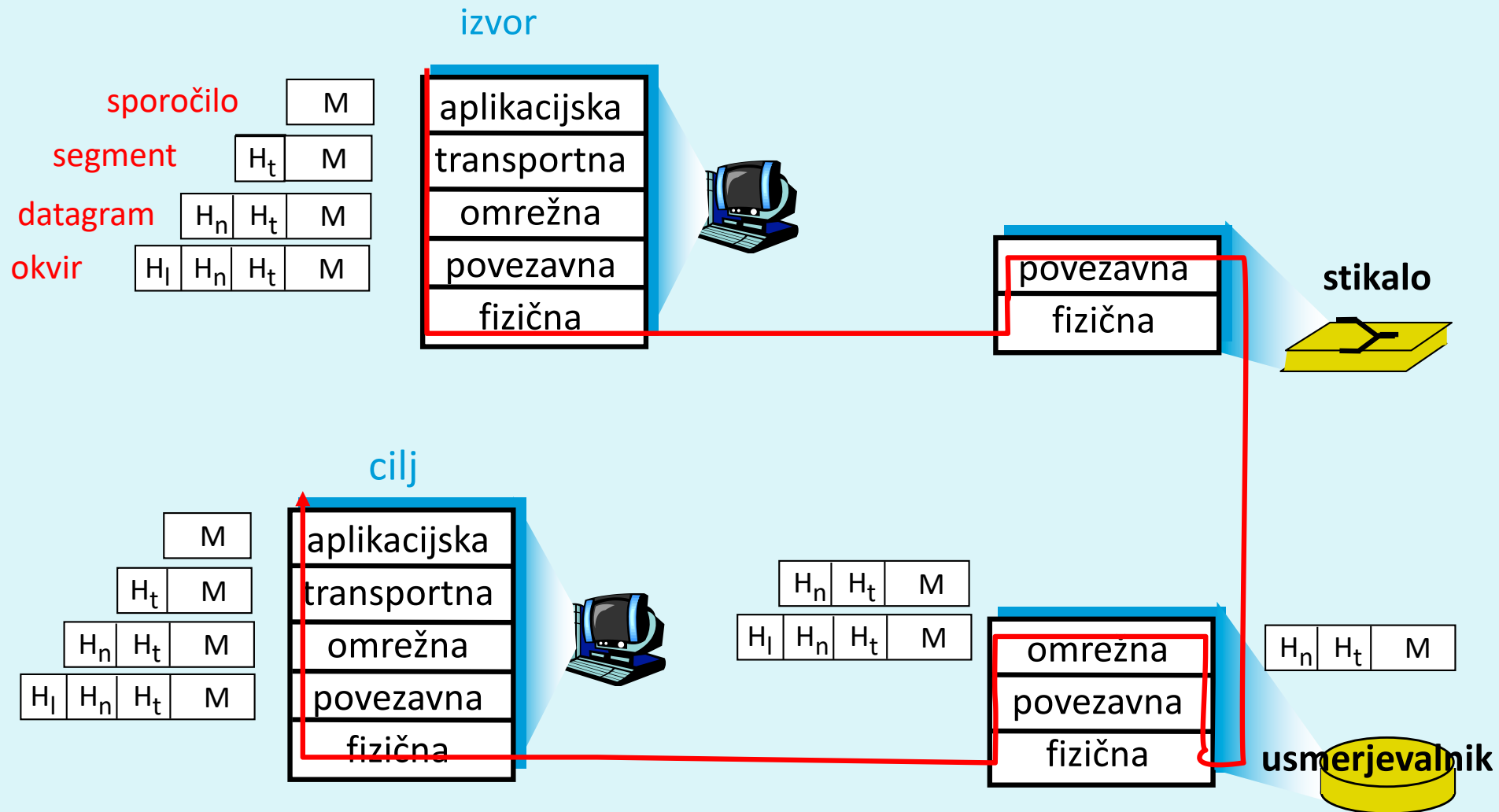
OSI model in model TCP/IP



Primerjava modelov:

- ISO OSI: **de iure**, teoretičen, sistematičen, pomanjkanje implementacij (izdelkov),
- TCP/IP: **de facto**, prilagodljiv, nesistematičen, fleksibilen, veliko izdelkov

Enkapsulacija



Omrežna in transportna plast: podrobneje

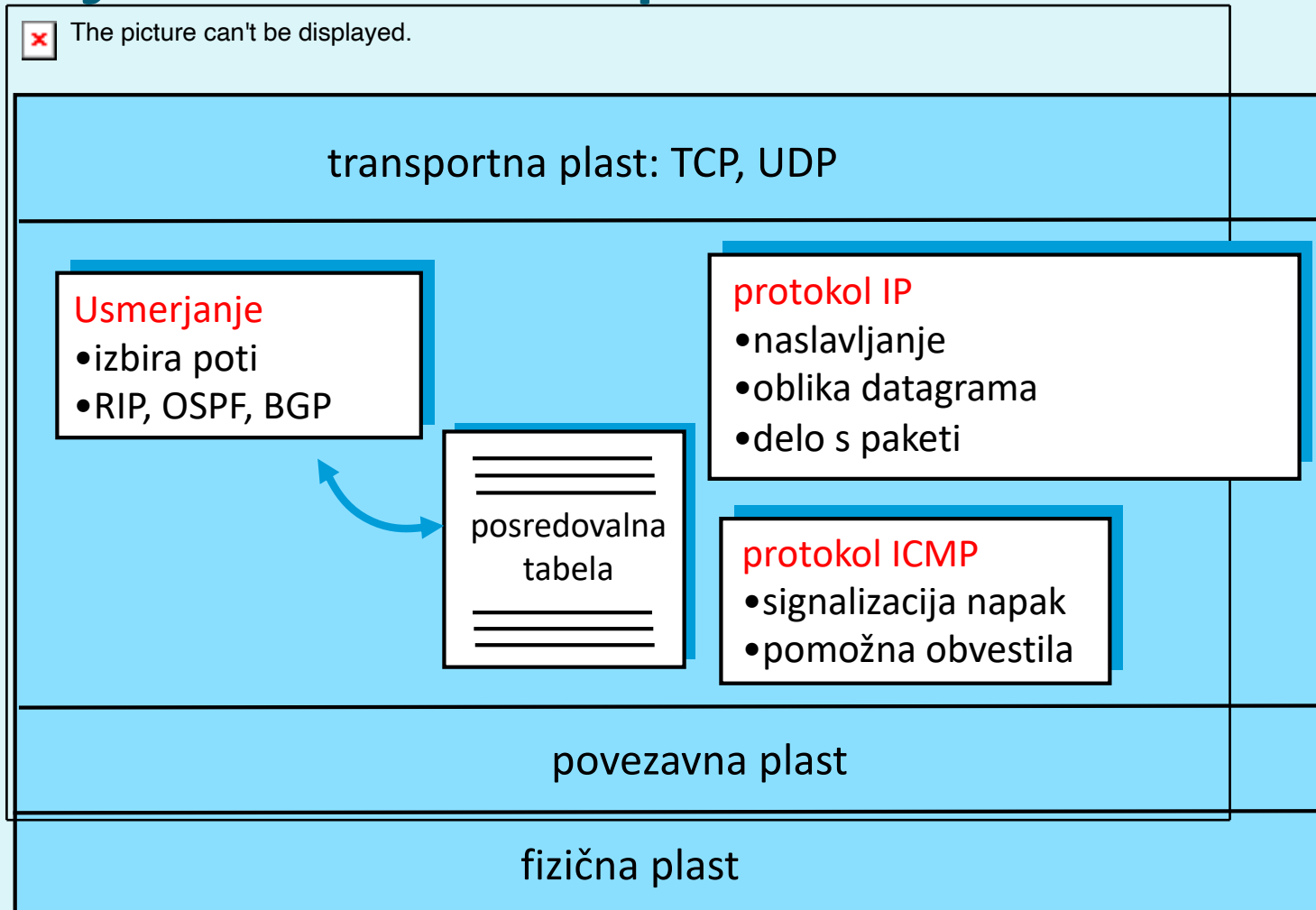
Omrežna plast:

Funkcije omrežne plasti



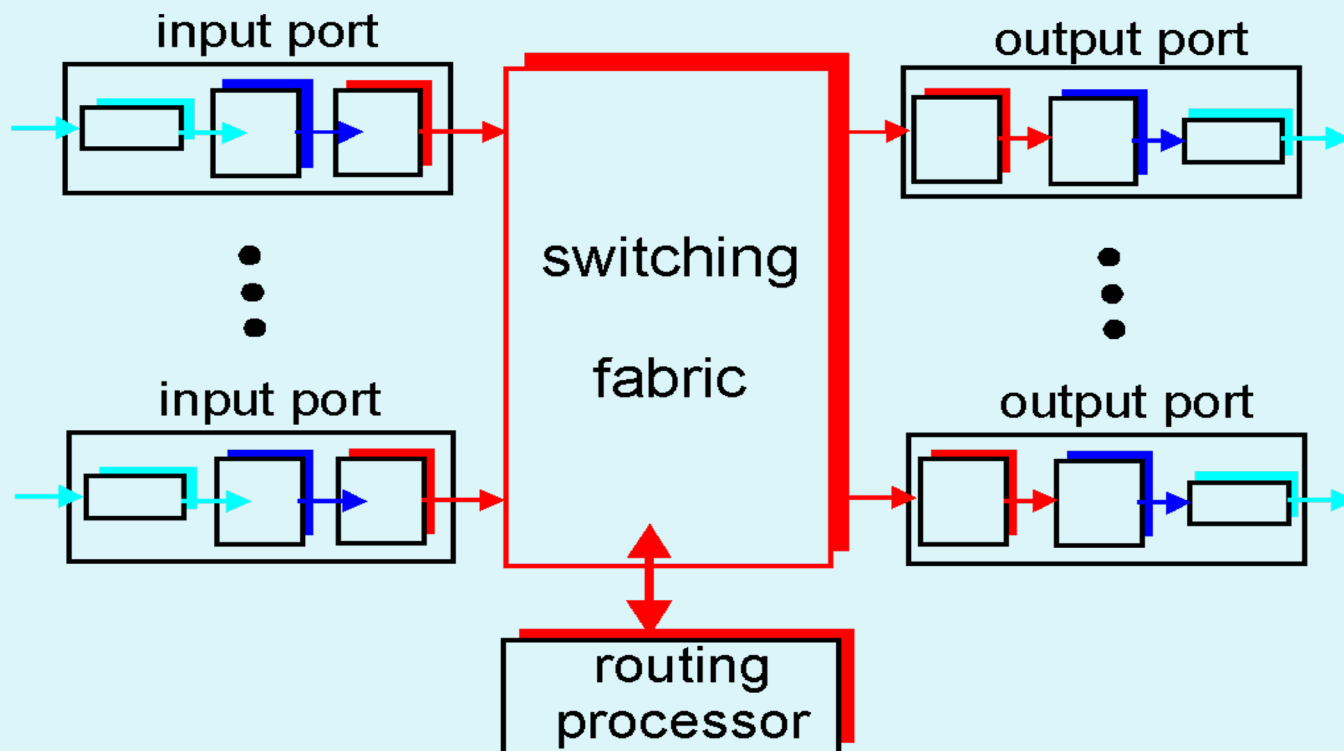
The picture can't be displayed.

funkcije
omrežne
plasti



Omrežna plast: Usmerjevalniki

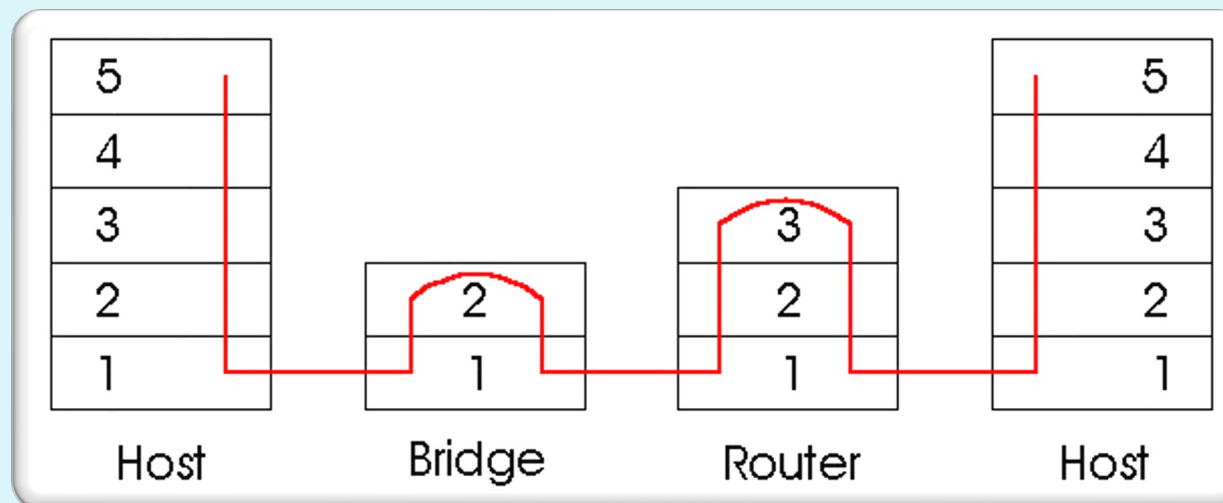
- uporaba usmerjevalnih (*routing*) protokolov (RIP, OSPF, BGP)
- posredovanje (*forwarding*) datagramov med vhodnimi in izhodnimi vrati



Omrežna plast:

Primerjava aktivne opreme

- **usmerjevalnik (router):**
 - naprava, ki deluje na OMREŽNI plasti
 - vzdržujejo usmerjevalne tabele, izvajajo usmerjevalne algoritme,
- **stikalo (switch):**
 - naprava, ki deluje na POVEZAVNI plasti,
 - vzdržujejo tabele za preklapljanje, izvajajo filtriranje in odkrivanje omrežja
- **povezovališče (hub):**
 - naprava, ki deluje na fizični plasti, danes niso več v rabi



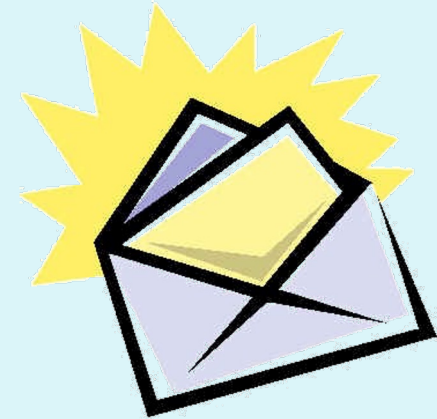
Omrežna plast: IPv4

- protokol na omrežni (3.) plasti OSI modela
- **IPv4 naslov** je 32 bitni naslov vmesnika. Primer:

11000001 00000010 00000001 01000010

ali

193.2.1.66



- **Podomrežje** je množica IP naslovov, ki so med seboj dosegljivi brez posredovanja usmerjevalnika. Maska (32 bitov) določa del IP naslova, ki predstavlja naslov podomrežja. Primer:

11111111 11111111 11110000 00000000 (255.255.255.240)

pomeni, da prvih 20 bitov IP naslova predstavlja naslov omrežja, preostalih 12 pa naslov vmesnika.

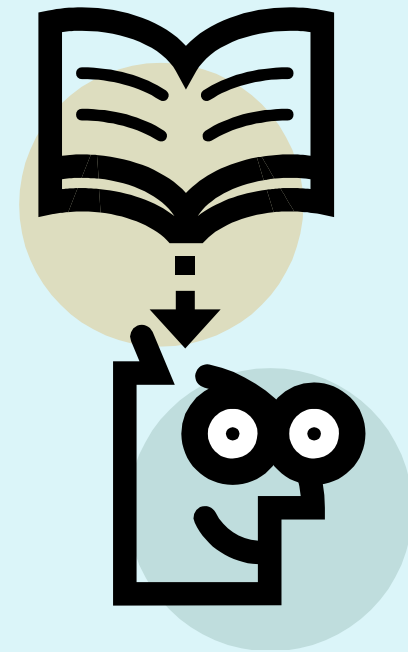
Omrežna plast:
Vaja!

- Podana sta IP naslov nekega vmesnika in maska podomrežja:

193.90.230.25 /20

Kakšen je naslov podomrežja?

Kakšen je naslov vmesnika?



Omrežna plast: IPv6

- **Prednosti:**

- večji naslovni prostor: 128 bitov
- hitro usmerjanje in posredovanje ter QoS omogoča že format glave, fragmentacije ni,
- implementacija IPSec znotraj IPv6 obvezna.

- **Naslov:** sestavljen iz 64 bitov za ID podomrežja + 64 bitov za ID vmesnika

```
0010000111011010 0000000011010011 0000000000000000 0010111100111011  
0000001010101010 0000000011111111 1111111000101000 1001110001011010
```

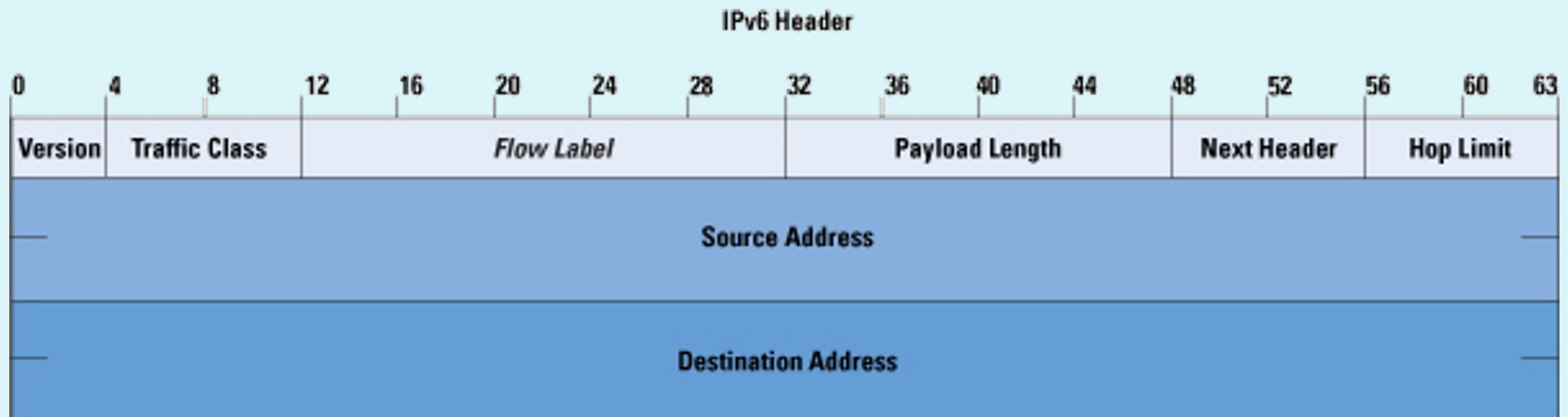
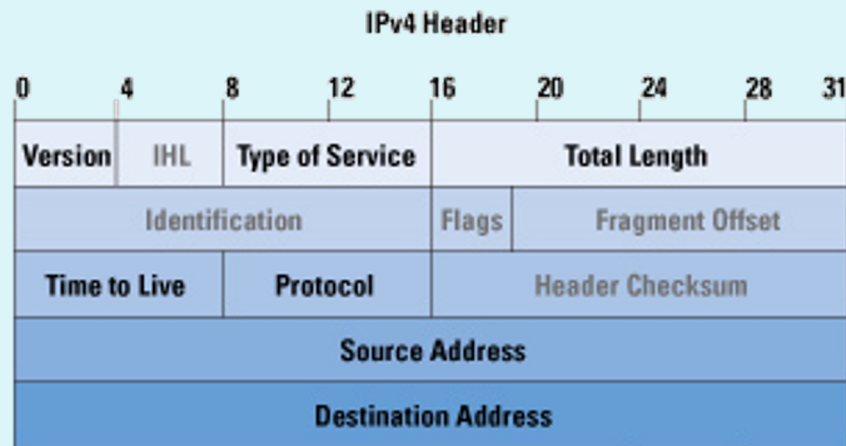
Zapisan šestnajstiško, ločeno z dvopičji

21DA:00D3:0000:0000:02AA:00FF:FE28:9C5A ali (brez vodilnih ničel)

21DA:D3:0:0:2AA:FF:FE28:9C5A ali (izpustimo bloke ničel)

21DA:D3::2AA:FF:FE28:9C5A

Omrežna plast: Primerjava IPv4 in IPv6



Omrežna plast:

IPv6 - načini naslavljanja

- **UNICAST:**
naslavljanje posameznega omrežnega vmesnika
- **MULTICAST:**
naslavljanje skupine omrežnih vmesnikov, dostava vsem vmesnikom v množici
- **ANYCAST:**
je naslov množice vmesnikov, dostava se izvede enemu (najbližjemu?) vmesniku iz te množice

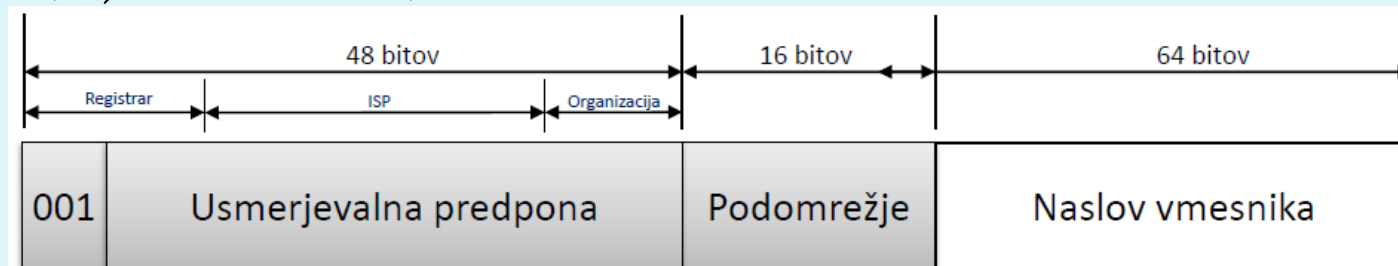


Vsak vmesnik ima lahko več naslovov različnih tipov.
(BROADCAST naslovov - v IPv6 ni več!)

Omrežna plast:

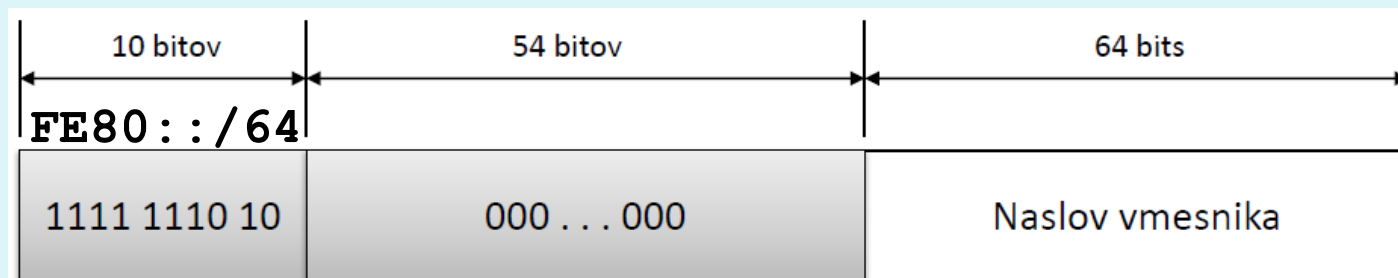
IPv6 - vrste unicast naslovov

1.) **globalni unicast** (= javni naslovi)



2.) **posebni naslovi** (localhost ::1, nedefiniran o::o, IPv4 naslovi)

3.) **link-local naslovi** (znotraj 1 povezave, adhoc omrežja)



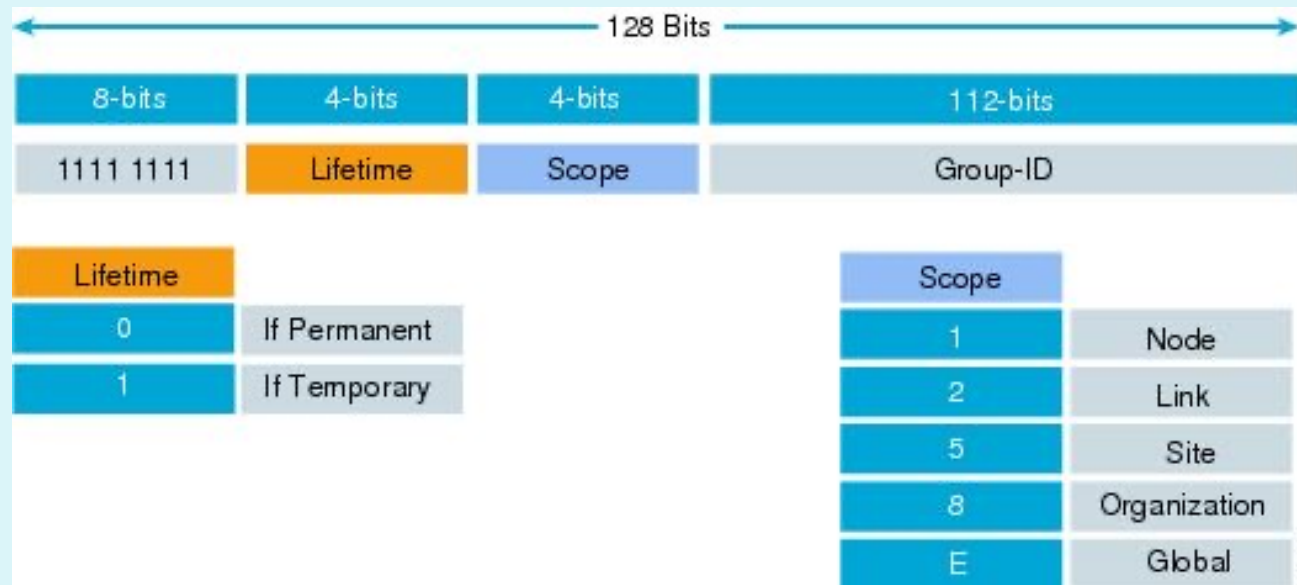
4.) **site-local** (=privatni naslovi, znotraj org., se ne usmerjajo, FEC0::/10)

5.) **unique-local** (=zasebni naslovi, dodeli registrar, znotraj org. se ne usmerjajo, so bolj strukturirani, FC00::/7)

Omrežna plast:

IPv6 – razpošiljanje (*multicast*)

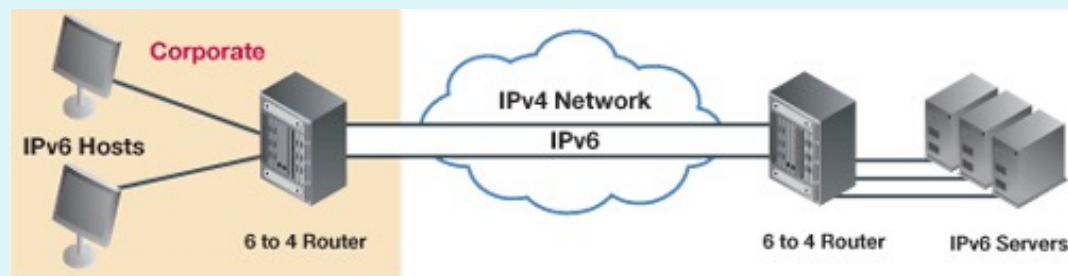
- 1.) FF02::1 (link local: vsi VMESNIKI)
- 2.) FF02::2 (link local: vsi USMERJEVALNIKI)
- 3.) Struktura naslova:



Omrežna plast:

IPv6 v omrežjih IPv4

- 1.) **dvojni sklad (dual-stack)**: usmerjevalniki poznajo IPv4 in IPv6. Z možnimi govori IPv6, z ostalimi pa IPv4.
- 2.) **tuneliranje**: IPv6 paket zapakiramo v enega ali več IPv4 paketov kot podatke.



Omrežna plast: Usmerjanje



• NAČINI

- statično / dinamično (upoštevanje razmer v omrežju)
- centralizirano / porazdeljeno (glede na poznavanje stanja celega omrežja)
- po eni poti / po več poteh

• IMPLEMENTACIJE:

- z vektorjem razdalj (RIP, IGRP, EIGRP)
- glede na stanje omrežja (OSPF, IS-IS)

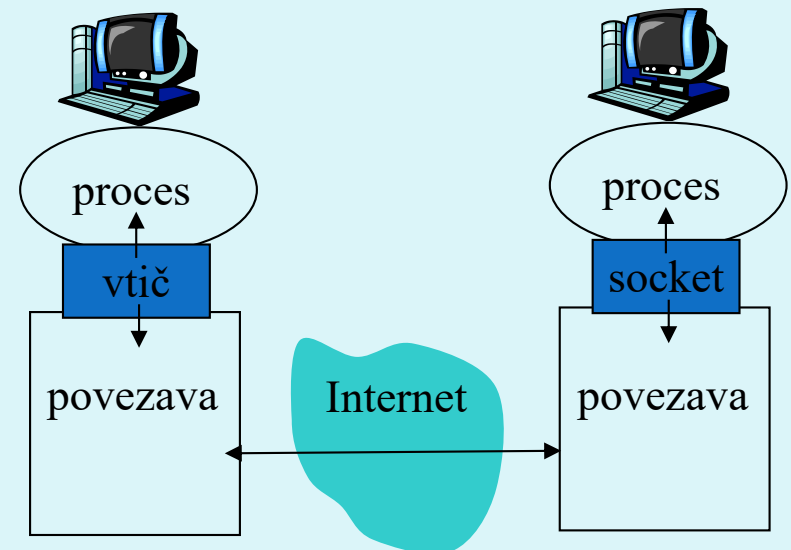
Transportna plast: Funkcionalnosti

- **Naloga:**

- Sprejem sporočila od aplikacije
- Sestavljanje segmentov v sporočilo za omrežno plast
- Predaja aplikacijski plasti

- **Vtič**

- vmesnik med transportno in aplikacijsko plastjo,
- proces naslovimo z IP številko in številko vrat (www: 80, SMTP: 25, DNS: 53, POP3: 110).



Transportna plast:

Povezavno in nepovezavno

- **Povezavna in nepovezavna komunikacija**

- TCP in UDP; ter ostali protokoli
- vzpostavitev, **prenos**, podiranje – povezave

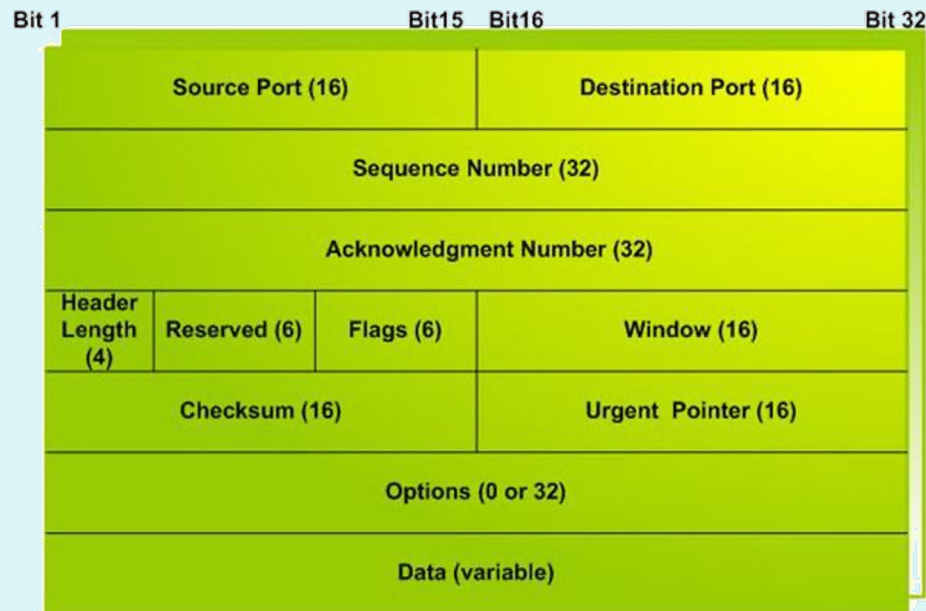


- **Potrjevanje**

- v protokolu (TCP)
- v aplikaciji (UDP)
- neposredno (ACK in NACK)
- posredno (samo ACK, sklepamo na podlagi števil paketa)
- sprotno potrjevanje: naslednji paket se pošlje šele po prejemu potrditve
- tekoče pošiljanje: ne čaka se na potrditve.

Transportna plast: TCP in UDP

The TCP Segment Format



The UDP Segment Format



Aplikacijska plast:

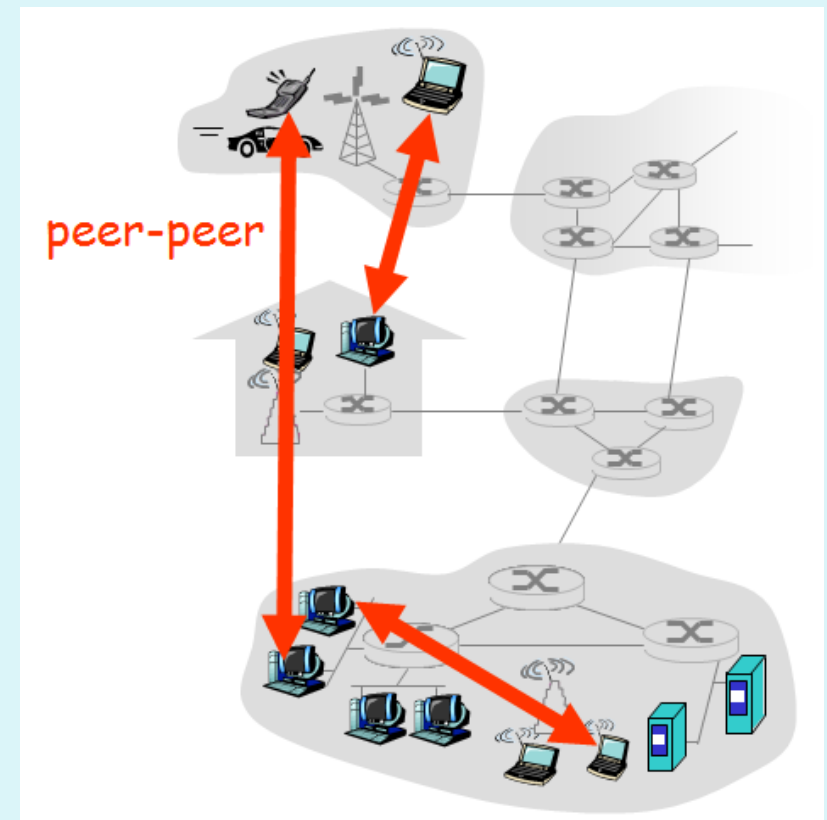
- **Klasične storitve – odjemalec-strežnik**

- telnet, ssh; rdesktop
- ftp, sftp
- WWW in HTTP,
- SMTP, POP₃, IMAP, MAPI
- DNS,
- SNMP, LDAP, RADIUS, ...
- ...

Aplikacijska plast:

- **Novejše storitve – P2P:**

- komunikacija poljubnih dveh končnih sistemov,
- strežniki niso nenehno prižgani,
- prekinjene povezave / spremembe IP naslovov,
- primeri: BitTorrent, Skype



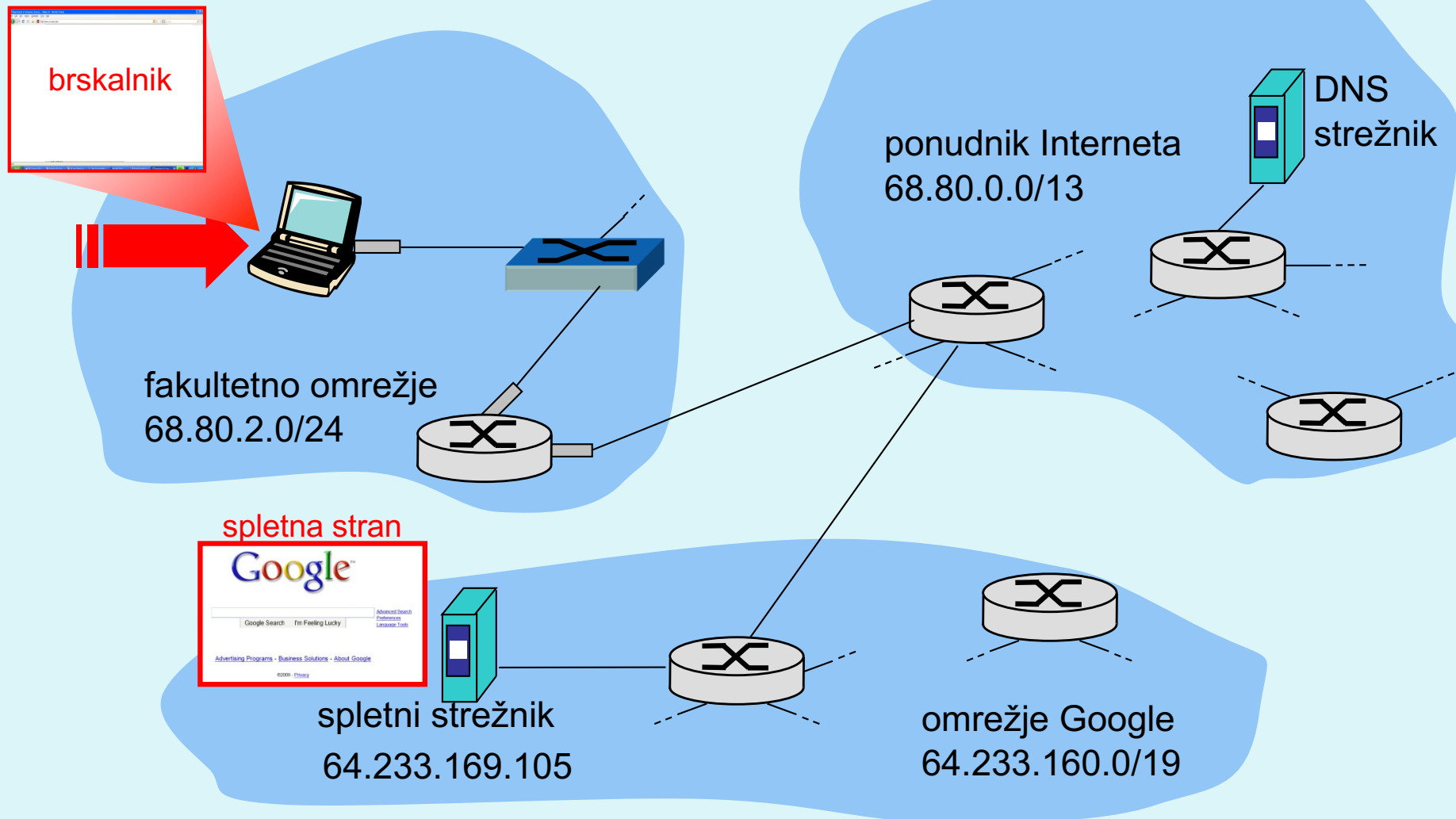
Omrežna in transportna plast:

Iz preteklosti za prihodnost

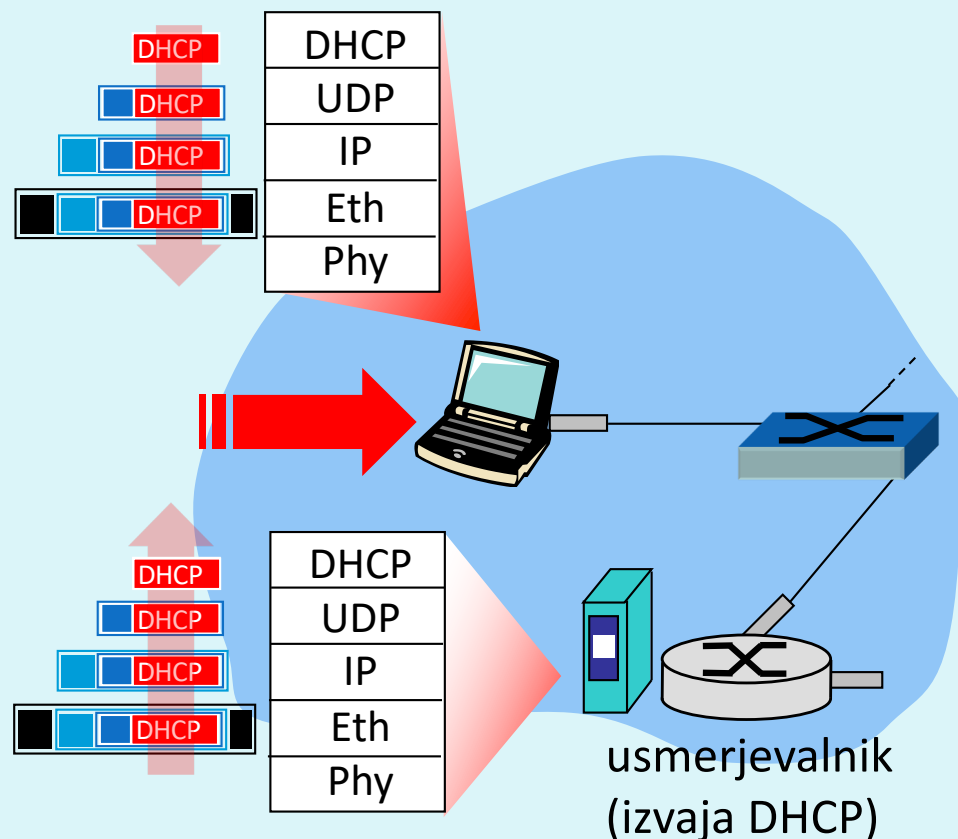
- **Problem:** pomanjkanje IPv4 naslovov
 - izkoristek zasebnih naslovnih prostorov
 - NAT prehodi – običajno hkrati požarni zidovi
 - preprosto v odjemalec-strežnik sistemih
 - v P2P potrebujemo preslikovalni naslov v zunanjem svetu
- V IPv6 NAT prehodi niso potrebni

Primer komunikacije

Primer komunikacije: spletno brskanje

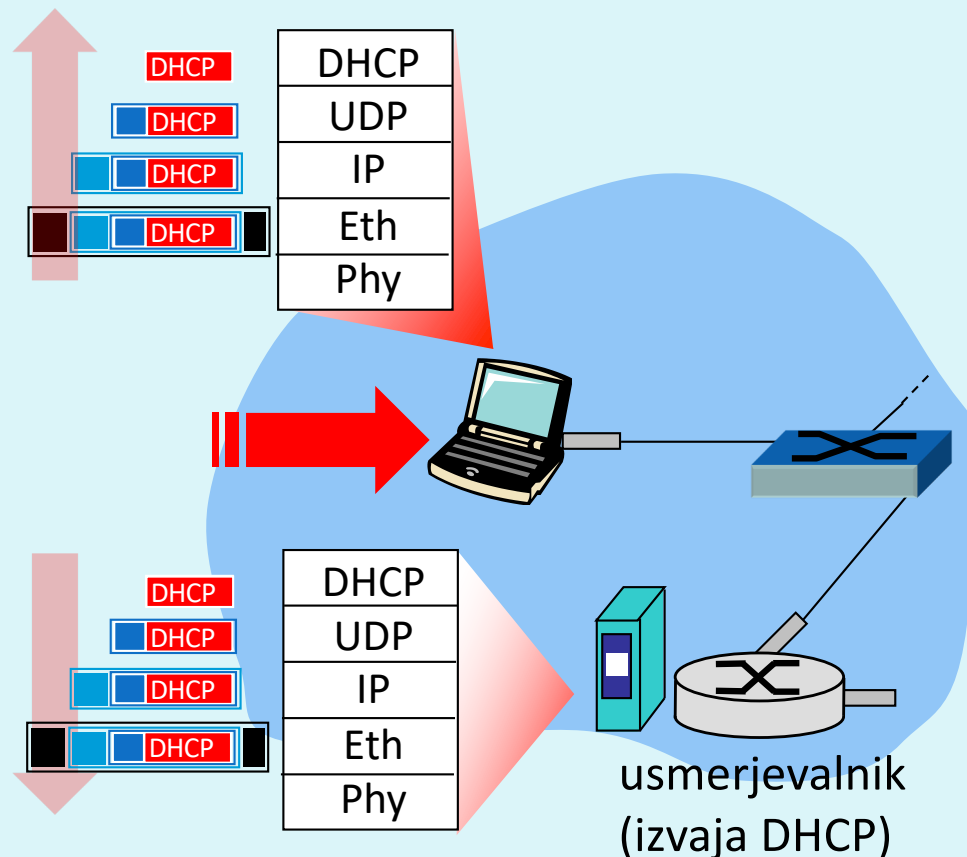


Primer komunikacije: spletno brskanje



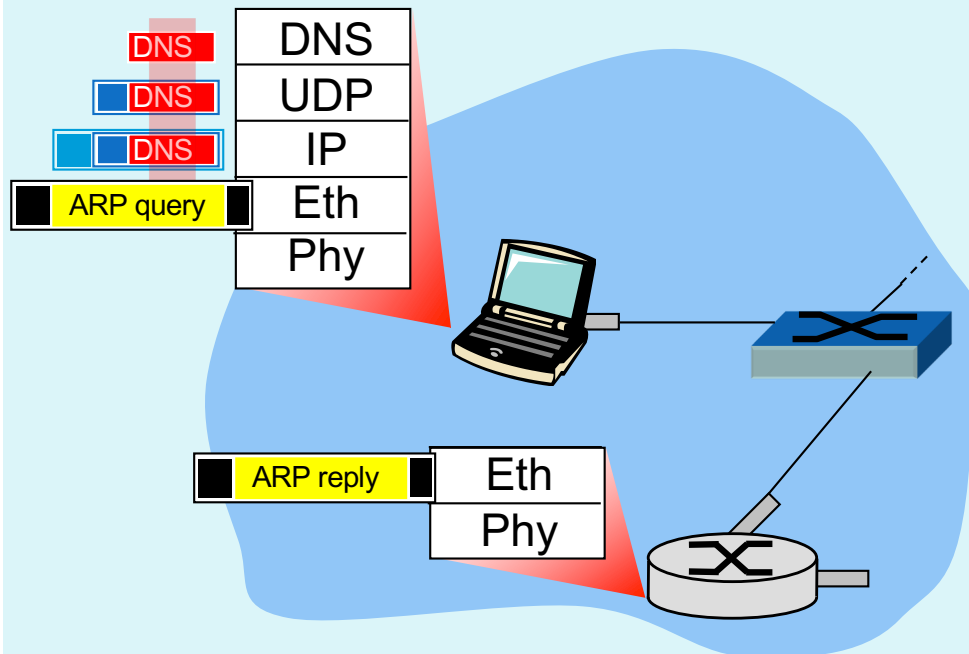
- notesnik ob priklopu na omrežje potrebuje **IP naslov** in podatke prehoda ter DNS strežnika: uporabi torej **DHCP**,
- zahteva DHCP se **enkapsulira**: UDP -> IP -> 802.1 Ethernet
- ethernet okvir se **odda** (*broadcast*) na omrežje, prejme ga usmerjevalnik, ki opravlja nalogo DHCP strežnika
- DHCP strežnik **prebere** vsebino DHCP zahteve

Primer komunikacije: spletno brskanje



- DHCP strežnik odgovori odjemalcu (notesniku) s paketom **DHCP ACK**, ki vsebuje njegov IP naslov ter naslove prehoda in DNS strežnika,
- odgovor **enkapsulira** DHCP strežnik (usmerjevalnik) in ga posreduje odjemalcu, ki ga **dekapsulira**,
- DHCP odjemalec dobi odgovor DHCP ACK,
- rezultat: odjemalec je pripravljen na komunikacijo.

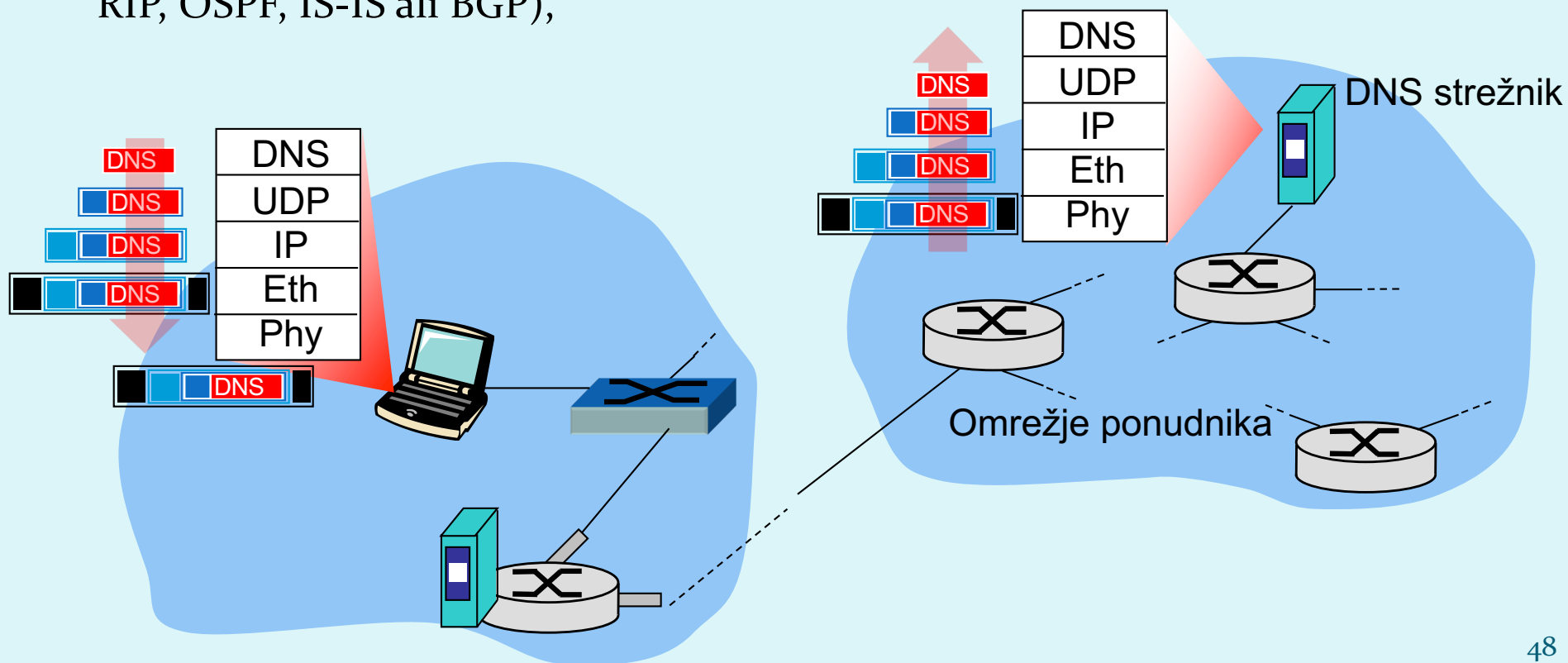
Primer komunikacije: spletno brskanje



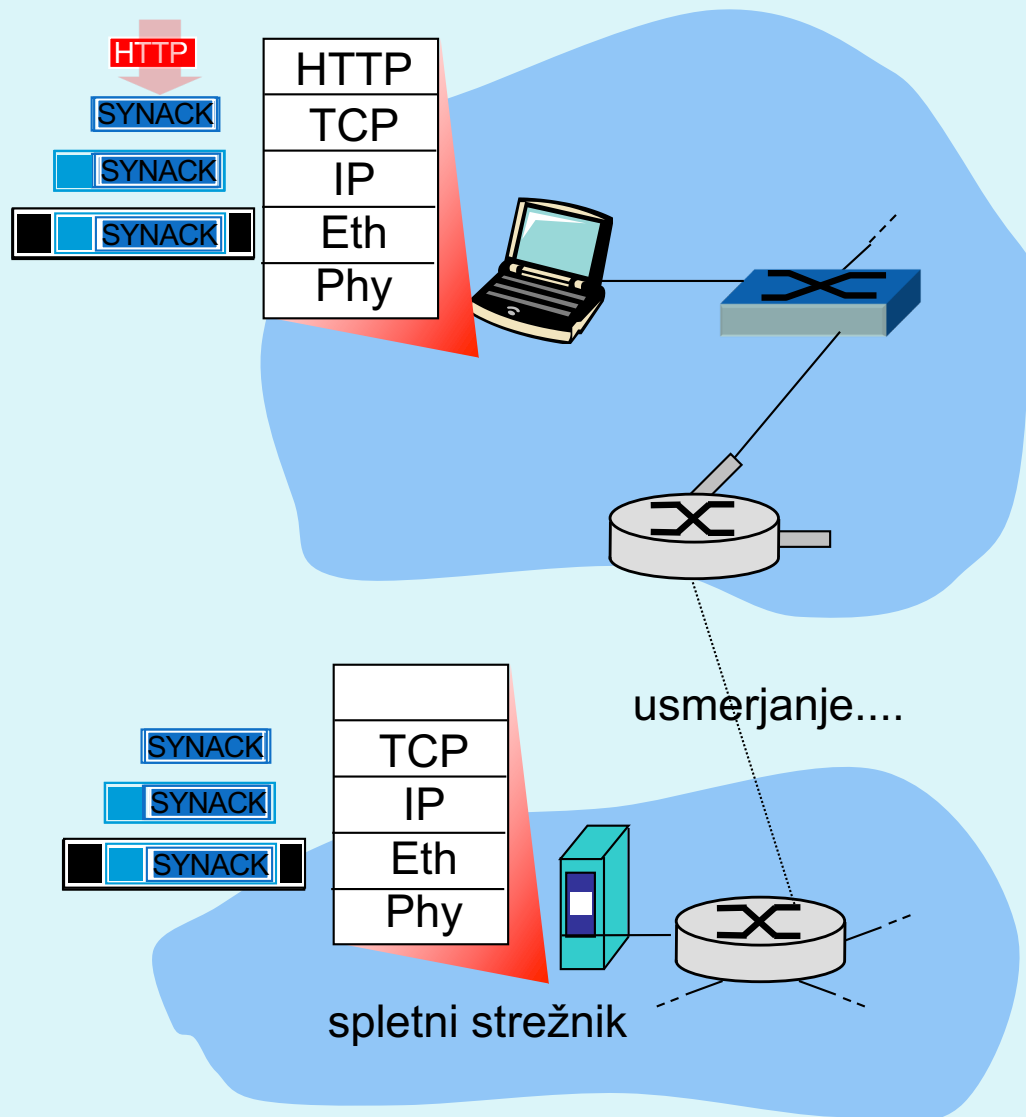
- pred pošiljanjem zahtevka HTTP, potrebujemo IP naslov strežnika `www.google.com`: **uporabi DNS**,
- enkapsulacija zahtevka DNS: UDP -> IP -> Ethernet. Potrebujemo MAC naslov usmerjevalnika: **uporabi ARP**
- razpošljemo **zahtevek ARP**, usmerjevalnik odgovori z **ARP odgovorom**, ki hrani njegov MAC naslov,
- klient sedaj pozna MAC naslov prehoda, ki mu lahko **pošlje DNS zahtevek**.

Primer komunikacije: spletno brskanje

- IP datagram z **zahtevkom DNS** se posreduje usmerjevalniku
- IP datagram se posreduje **DNS strežniku**, ki je v omrežju ponudnika (z uporabo usmerjevalnih protokolov RIP, OSPF, IS-IS ali BGP),
- DNS strežnik **dekapsulira** zahtevek in posreduje uporabniku IP naslov spletnega strežnika www.google.com

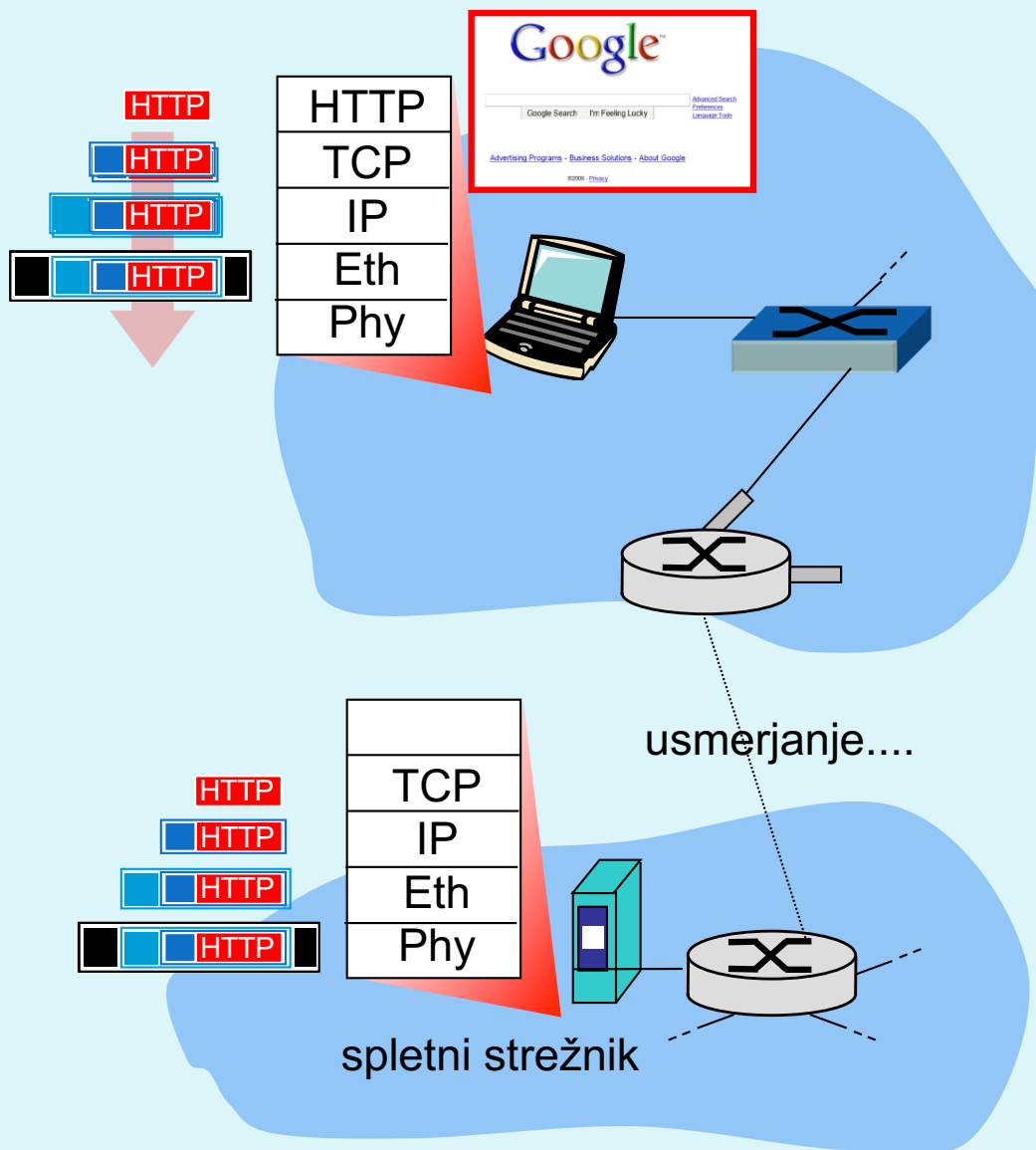


Primer komunikacije: spletno brskanje



- za pošiljanje **HTTP zahtevka**, odjemalec najprej naslovi **TCP vtič** spletnega strežnika,
- **TCP SYN** segment se preko omrežja usmeri do spletnega strežnika
- spletni strežnik odgovori s **TCP SYNACK** (potrditev rokovanja),
- sedaj je **TCP povezava vzpostavljena!**

Primer komunikacije: spletno brskanje



- **HTTP zahtevek** se pošlje na **TCP vtič** spletnega strežnika,
- **IP datagram**, ki vsebuje spletno zahtevo po strani `www.google.com` se usmeri k spletnemu strežniku
- spletni strežnik odgovori s **HTTP REPLY**, ki vsebuje vsebino strani
- IP datagram s stranjo se usmeri h klientu,
- **WWW stran je kočno prikazana!**

Zajem podatkov iz omrežja

The screenshot shows the Wireshark Network Analyzer interface. The main pane displays a list of captured packets. Packet 122 is selected, and its details are shown in the packet list pane. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Len	Time	Source	Destination	Protocol	Info
114	54	53.550000	207.183.142.87	204.252.103.16	TCP	1013 > 22 [FIN, ACK] Seq=3084 Ack=644 Win=
115	60	53.550000	204.252.103.16	207.183.142.87	TCP	22 > 1013 [ACK] Seq=644 Ack=3085 Win=16384
116	60	53.550000	204.252.103.16	207.183.142.87	TCP	22 > 1013 [FIN, ACK] Seq=644 Ack=3085 Win=
117	54	53.550000	207.183.142.87	204.252.103.16	TCP	1013 > 22 [ACK] Seq=3085 Ack=645 Win=32256
118	342	53.920000	204.252.103.79	255.255.255.255	BOOTP	[Packet size limited during capture]
119	240	54.210000	00000000.00609739b071	00000000.ffffffffffff	NMPI	[Packet size limited during capture]
120	189	54.250000	00:20:af:92:d4:5f	03:00:00:00:00:01	SMB	[Packet size limited during capture]
121	60	54.650000	08:00:4e:08:5d:56	01:80:c2:00:00:00	STP	Conf. Root = 65535/08:00:4e:08:5d:56 Cost
122	60	54.710000	207.183.142.87	204.252.102.2	POP	Request: STAT
123	66	54.710000	204.252.102.2	207.183.142.87	POP	Response: +OK 2 3467
124	60	54.710000	207.183.142.87	204.252.102.2	POP	Request: LIST

Frame 122 (60 bytes on wire, 60 bytes captured)

- Ethernet II, Src: 00:c0:4f:c7:eb:c0 (00:c0:4f:c7:eb:c0), Dst: 00:00:0c:36:00:19 (00:00:0c:36:00:19)
- Internet Protocol, Src: 207.183.142.87 (207.183.142.87), Dst: 204.252.102.2 (204.252.102.2)
- Transmission Control Protocol, Src Port: 22587 (22587), Dst Port: 110 (110), Seq: 29, Ack: 134, Len: 6
 - Source port: 22587 (22587)
 - Destination port: 110 (110)
 - Sequence number: 29 (relative sequence number)
 - [Next sequence number: 35 (relative sequence number)]
 - Acknowledgement number: 134 (relative ack number)
 - Header length: 20 bytes
 - Flags: 0x0018 (PSH, ACK)

```
0000 00 00 0c 36 00 19 00 c0 4f c7 eb c0 08 00 45 00  ...6.... 0....E.
0010 00 2e 75 02 40 00 40 06 34 ba cf b7 8e 57 cc fc  ..u.@. 4....W..
0020 66 02 58 3b 00 6e 6a 0f a9 ba a6 bd ae 90 50 18  f.X;.nj.....P.
0030 7d 78 3d cc 00 00 53 54 41 54 0d 0a                }x=...ST AT..
```

Sequence number (tcp.seq), 4 bytes P: 3632 D: 3632 M: 0

Zajem podatkov iz omrežja: primer DHCP

zahtevek

```
Message type: Boot Request (1)
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x6b3a11b7
Seconds elapsed: 0
Bootp flags: 0x0000 (Unicast)
Client IP address: 0.0.0.0 (0.0.0.0)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 0.0.0.0 (0.0.0.0)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Server host name not given
Boot file name not given
Magic cookie: (OK)
Option: (t=53,l=1) DHCP Message Type = DHCP Request
Option: (61) Client identifier
    Length: 7; Value: 010016D323688A;
    Hardware type: Ethernet
    Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Option: (t=50,l=4) Requested IP Address = 192.168.1.101
Option: (t=12,l=5) Host Name = "nomad"
Option: (55) Parameter Request List
    Length: 11; Value: 010F03062C2E2F1F21F92B
    1 = Subnet Mask; 15 = Domain Name
    3 = Router; 6 = Domain Name Server
    44 = NetBIOS over TCP/IP Name Server
    .....
```

odgovor

```
Message type: Boot Reply (2)
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x6b3a11b7
Seconds elapsed: 0
Bootp flags: 0x0000 (Unicast)
Client IP address: 192.168.1.101 (192.168.1.101)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 192.168.1.1 (192.168.1.1)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Server host name not given
Boot file name not given
Magic cookie: (OK)
Option: (t=53,l=1) DHCP Message Type = DHCP ACK
Option: (t=54,l=4) Server Identifier = 192.168.1.1
Option: (t=1,l=4) Subnet Mask = 255.255.255.0
Option: (t=3,l=4) Router = 192.168.1.1
Option: (6) Domain Name Server
    Length: 12; Value: 445747E2445749F244574092;
    IP Address: 68.87.71.226;
    IP Address: 68.87.73.242;
    IP Address: 68.87.64.146
Option: (t=15,l=20) Domain Name = "hSDL.ma.comcast.net."
```

Omrežna varnost



Omrežna varnost

- **Je področje, ki:**
 - analizira možnosti vdorov v sisteme,
 - načrtuje tehnike obrambe pred napadi,
 - snuje varne arhitekture, ki so odporne pred vdori.
- **Internet ni bil snovan ozirajoč se na varnost!**
 - *vizija interneta je sprva bila: „To je skupina ljudi, ki si med seboj zaupajo in je priključena na skupno omrežje”*
 - pri izdelavi protokola so ga proizvajalci delali z metodologijo „krpanja”,
 - varnostne mehanizme je potrebno upoštevati na vseh plasteh OSI modela.

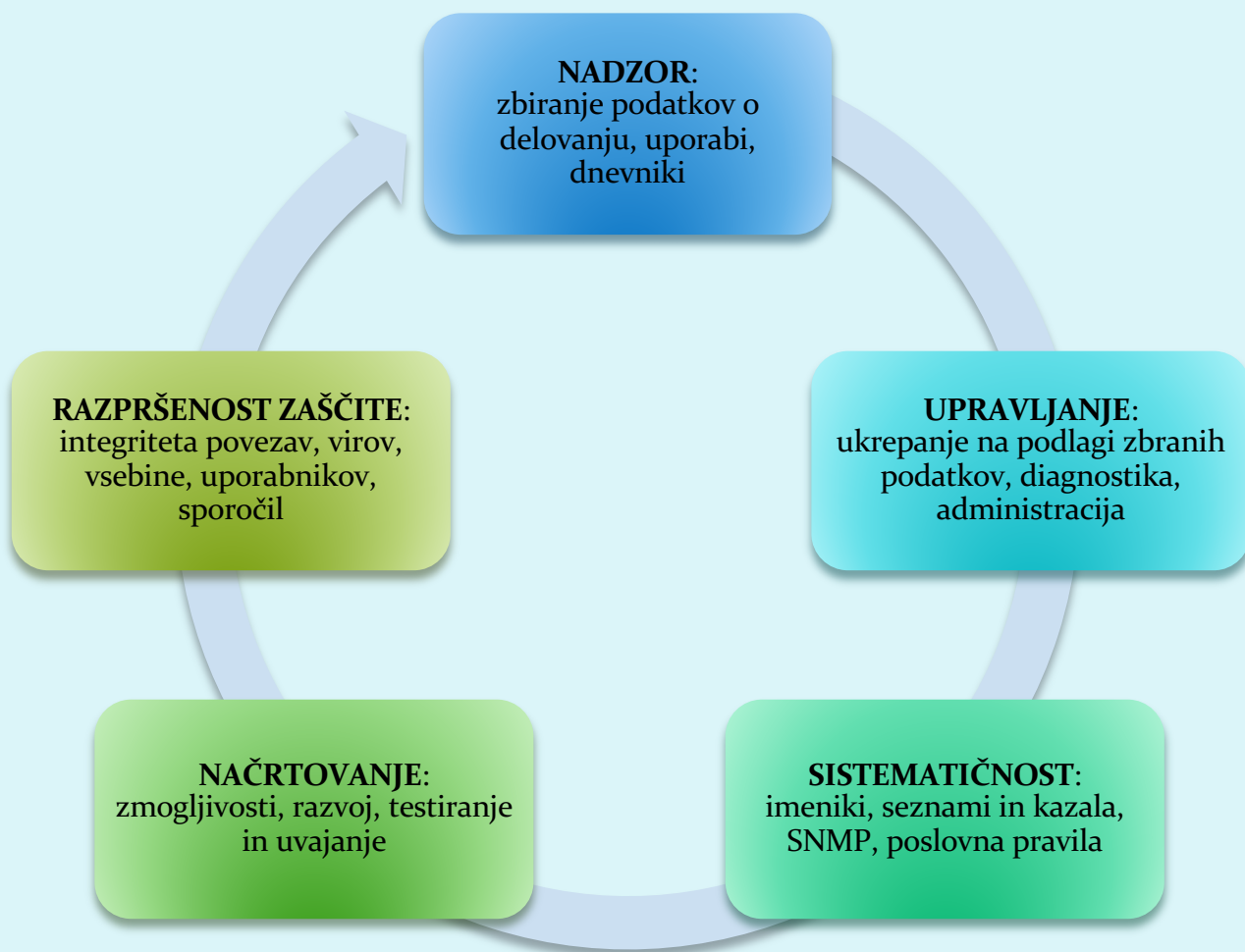
Kako lahko vdiralec škoduje sistemu?

Ima veliko možnih pristopov in tehnik!

- **prisluškovanje:** prestrezanje sporočil,
- aktivno **ponarejanje** sporočil v neki komunikaciji,
- **kraja identitete (impersonacija):** ponaredi lahko izvorni naslov ali poljubno drugo vsebino paketa,
- **prevzem povezave (hijacking):** odstrani pravega pošiljatelja ali prejemnika iz komunikacije in prevzame njegovo vlogo,
- **onemogočanje nudenja storitve (denial of service):** onemogoči uporabo regularne storitve (npr. s tem, da jo preobremeni)



Varnost: zagotavljanje zanesljivosti



Elementi varne komunikacije

- **Zaupnost** – kdo sme prebrati? (šifriranje)
- **Overovljenje (*authentication*)** – dokaži, da si res ti (identifikacija – povej, kdo si, brez dokaza)
- **Razpoložljivost in nadzor dostopa** – preprečevanje nelegitimne rabe virov (*avtorizacija (authorization)* – ugotavljanje, ali nekaj smeš storiti, *beleženje (accounting)* – kaj je kdo uporabljal)
- **Integriteta sporočila** – je bilo med prenosom spremenjeno?
- **Onemogočanje zanikanja (*nonrepudiation*)** – res si poslal / res si prejel.
- V praksi:
 - požarne pregrade, zaznava vdorov (*intrusion detection*) sistemi,
 - varnost na aplikacijski, transportni, omrežni in povezavni plasti

Zaupnost sporočil: šifriranje (zakrivanje) vsebine

Je način obrambe pred **pasivnimi** vdiralci (prisluškovalci) in **aktivnimi** vdiralci (ponarejevalci).


Sporočilo **P** šifriramo s ključem **E()** - dobimo **kriptogram E(P)**.
Kriptogram **E(P)** predelamo v izvorno obliko s ključem **D()**, dobimo izvorno sporočilo **D(E(P))=P**.

Vrste metod:

- **zamenjalne** (substitucijske, menjava znakov) / **izmenjalne** (transpozicijske, vrstni red znakov)
- **simetrične** (**E=D**, npr. DES, AES) / **asimetrične** (**E≠D**, npr. RSA, ECC)

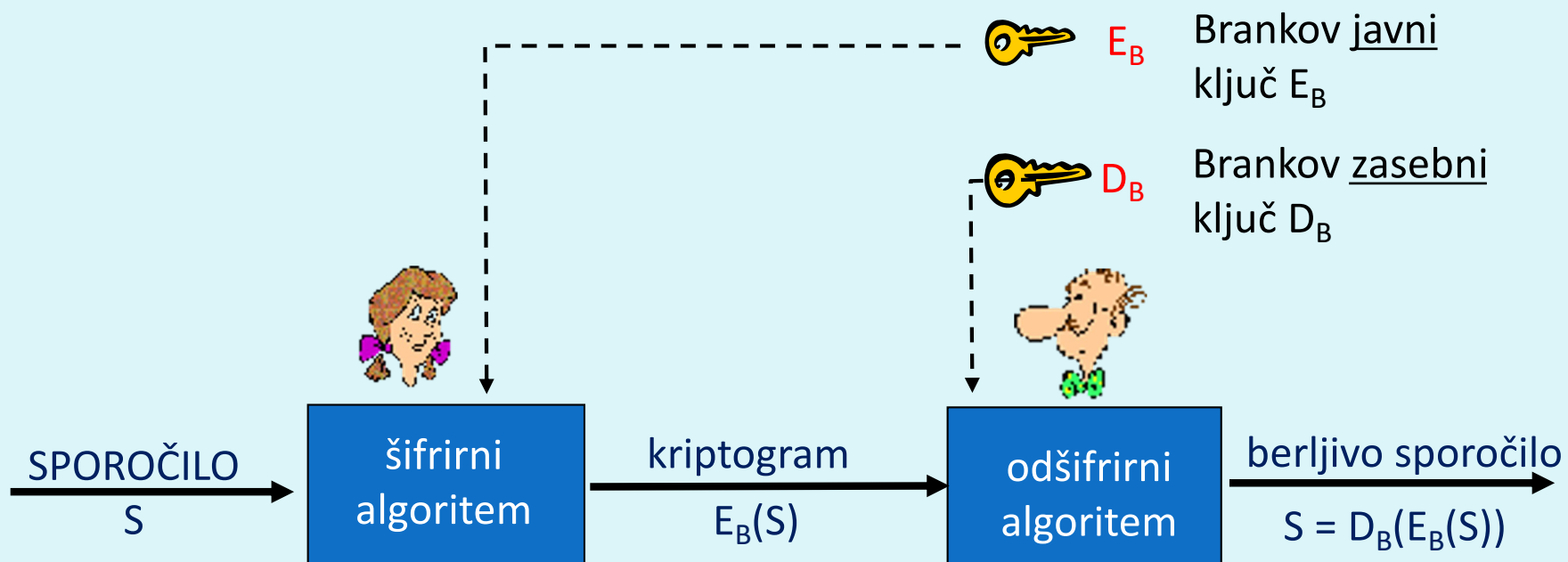
Vrste šifriranje

- Šifriranje uporablja ključe
 - šifrirni algoritem je običajno znan vsem,
 - tajni so le ključi
 - šifriranje: skrivanje vsebine
 - kriptanaliza („razbijanje” kode)
- Šifriranje z javnimi ključi
 - $E() \neq D()$: dva ključa – javni in zasebni
- Simetrično šifriranje
 - $E() = D()$: samo en ključ
- Zgoščevalne funkcije – ni šifriranje
 - ne uporabljajo ključev. Kako so lahko koristne?



$$f(x) = e_0(x)y_0 + e_1(x)y_1 + e_2(x)y_2 + e_3(x)y_3$$
$$(x+1)^2 = \left(\frac{x(x-2)}{2}\right)1 + (x(x-1))0 + \left(\frac{x(x-1)}{2}\right)$$
$$= \left(\frac{x-1)(x-2)}{2}\right)1 + (x(x-1))0 + \left(\frac{x(x-1)}{2}\right)$$
$$f(x,y) = \frac{x+1}{x} \frac{x(x-1)}{2}$$
$$(y+6x+3)^4 - (2x^2+7x+8)^2 (y+9x+6)^4 (y+1)$$
$$1)(x+6)^4(x+9)^4 \frac{x(x+6)^2}{(y+8x+3)^4}$$
$$-9b + \sqrt{3} \sqrt{4a^3 + 27b^2} (y+6x)^2 (y+10x+8) x + 1$$
$$\frac{2^{11} 3^{2/3}}{x(x+6)^2} (y+9x+6)^4$$
$$\frac{(y+8x)^2}{(1-i\sqrt{3})(-9b + \sqrt{3} \sqrt{4a^3 + 27b^2})^{1/3}} (y+8x+3)^4$$
$$\frac{1}{1/3} + \frac{1}{2^{11} 3^{2/3} x + 9} (y+8x+3)^4 (y+1)$$
$$(y+8x)^2 (y+7x+4)^4 (y+1)$$

Šifriranje z javnimi ključi



Šifriranje z javnimi ključi

- Algoritmi za šifriranje z javnimi ključi so asimetrični, E= šifrirni ključ, D= odšifrirni ključ, velja **$E \neq D$**
- Ključa **E** in **D** morata izpolnjevati naslednje zahteve glede šifriranje sporočila **S**:
 1. **$D(E(S)) = D(E(S)) = S$**
 2. Iz znanih **S** in **E(S)** mora biti nemogoče ugotoviti **D**.
 3. Iz **E** mora biti zelo težko / nemogoče ugotoviti **D**.
- Najbolj znan algoritem je **RSA** (Rivest, Shamir, Adelman). RSA uporablja velika praštevila za določitev D in E, postopek (od)šifriranja pa je enak računanju ostanka pri deljenju s produktom teh praštevil.

Problem: distribucija ključev, počasnost.

Zakaj je RSA varen?

- Denimo, da poznamo javni ključ neke osebe (določen z dvojico števil (n, e)). Za ugotavljanje zasebnega ključa d moramo poznati delitelje števila n . Iskanje deliteljev nekega velikega števila pa je težko ali neizvedljivo z današnjimi računskimi kapacitetami.
- Kako poiskati dovolj velika praštevila?
 - večkrat izvedemo „ugibanje“: generiramo veliko število, nato ga testiramo, ali je praštevilo,
 - za testiranje praštevil obstajajo danes učinkoviti algoritmi.

Integriteta

- **Integriteta uporabnikov**: dokazuje, (i) kdo je sporočilo poslal (elektronski podpis) in (ii) da sporočilo bere le pravi prejemnik (zakrivanje). $S, A \rightarrow B$:

A:: $E_B(D_A(S)) \rightarrow \mathbf{XXX}$

B:: $D_B(\mathbf{XXX}) \equiv D_B(\underline{E_B(D_A(S))}) \equiv D_A(S) \equiv E_A(D_A(S)) \rightarrow S$

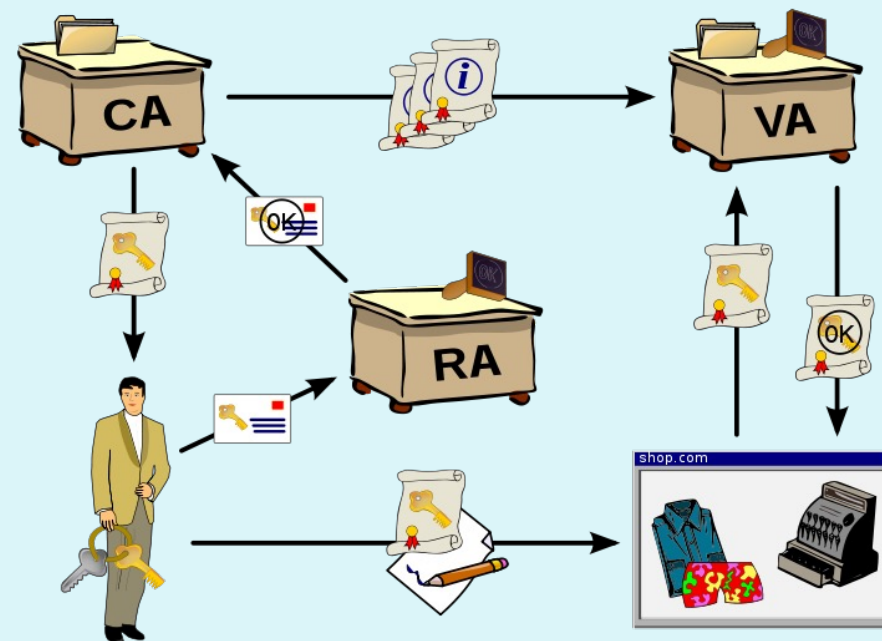
- **Integriteta sporočila**: dokazuje, da sporočilo (tudi nešifrirano!) ni bilo spremenjeno. Uporabljajo se zgoščevalne funkcij, ki izračunajo podpis/izvleček sporočila $\mathbf{sig}(S)$. To vrednost podpišemo z mehanizmom elektronskega podpisa

$$D_A(\mathbf{sig}(S)) = \mathbf{sss}$$

in \mathbf{sss} pošljemo skupaj z originalnih sporočilom S : (S, \mathbf{sss}) Prejemnik ponovno izračuna $\mathbf{sig}(S)$ in preveri $\mathbf{sss} = \mathbf{sig}(S)$.

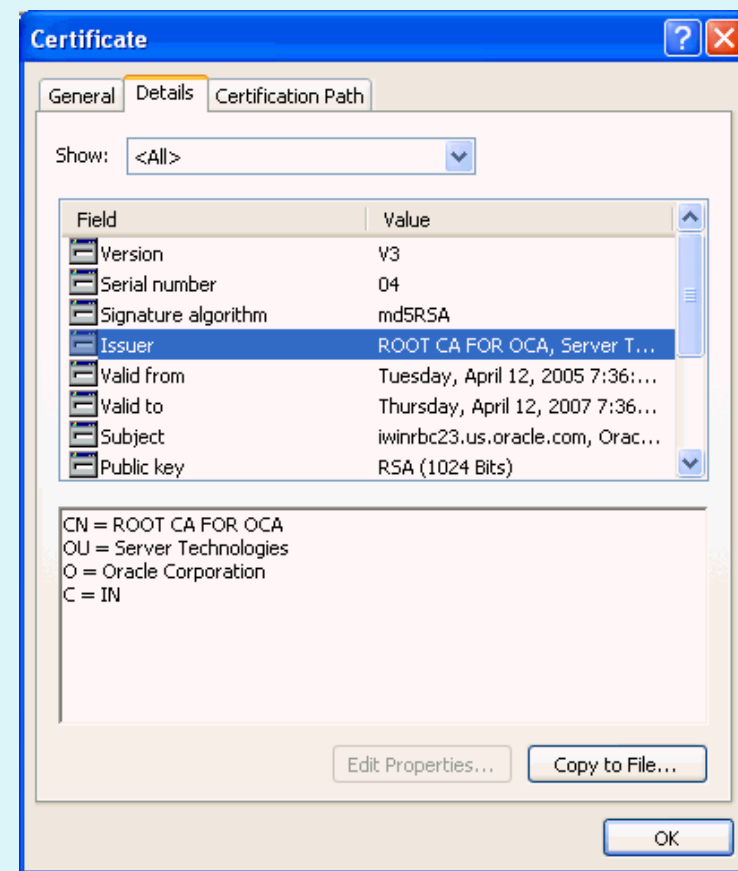
Šifriranje z javnimi ključi

- **PKI (Public Key Infrastructure)** je sistem, ki opredeljuje izdelavo, upravljanje, distribucijo, shranjevanje in preklic digitalnih certifikatov.
- Uporabnike overovimo s pomočjo javnih ključev, ki so overovljeni s strani certifikacijske agencije (*certificate authority*, **CA**).



Certifikati

- Sistem PKI vsebuje certifikacijske agencije (angl. certification authority), ki izdajajo, hranijo in preklicujejo certifikate.
- Certifikati so definirani s standardom X.509 (RFC 2459)
- Certifikat vsebuje
 - naziv izdajatelja,
 - ime osebe, naslov, ime domene in druge osebne podatke,
 - javni ključ lastnika,
 - digitalni podpis (podpisan z zasebnim ključem izdajatelja),



Naslednjič gremo naprej!

- priključitev računalnika na omrežje
- zagon računalnika: protokola DHCP in BOOTP
- arhitektura strežnik – odjemalec,
- protokol: delovanje, njegove funkcije,
- sled protokola

