

# A

- 1) Kako se razlikujeta CSMA/CA in CSMA/CD?
- 2) Kaj imata skupnega napada ARP storm in Smurf?
- 3) Narišite protokolarni sklad TCP/IP, poimenujte plasti ter navedite naloge in ključne funkcije omrežne plasti.
- 4) Kako deluje NAT-PAT?
- 5) Kaj je dvojni sklad IPv4 in IPv6 in kako deluje?
- 6) Usmerjevalnik sprejme datagram velikosti 2320 bytov.
  - a) Kakšen je največji možen MTU na izhodni povezavi, da se bo ta datagram fragmentiral na 3 fragmente, torej da 2 fragmenta ne bosta zadoščala?
  - b) Kolikšna bo velikost izhodnih datagramov – fragmentov, če je MTU tak, kot ste ga izračunali v točki a?
- 7) IP naslovi:
  - a) Vaš računalnik ima v omrežju IPv4 naslov 10.168.3.23/27. Izračunajte naslov omrežja, naslov broadcast, najmanjši in največji naslov naprave v tem omrežju. Izračunajte še koliko naprav lahko priključimo v to podomrežje.
  - b) Označite kateri od spodnjih naslovov IPv6 niso sintaktično pravilni. Označite tudi mesto napake in komentirajte zakaj so po vašem mnenju ti naslovi napačni:
    1. ::::
    2. 2001:770:10:300::134.226.81.11
    3. 231:0:a::1:b::1
    4. 2001:1470:fffd:16:103:22:0:1
    5. 2001:1470:ffFd:1234:abab:37:1::a

- 8) Usmerjanje
  - a) Iz usmerjevalnika smo prebrali spodnjo usmerjevalno tabelo:

Omrežje	Prehod	Vmesnik
10.0.0.0/16		A
10.0.1.0/26		B
192.168.12.64/26		C
10.0.2.0/24		D
0.0.0.0/0	10.0.2.1	D

Na kateri vmesnik (če je podan napišite tudi prehod) bo usmerjevalnik usmeril pakete z naslednjimi ciljnim naslovi: 10.0.0.1, 10.0.1.34, 10.0.1.123, 10.0.2.3, 192.168.12.70, 192.168.12.240, 192.168.1.1, 193.2.1.66, 10.0.3.4, 10.1.2.3

- b) Kako bi morali dopolniti usmerjevalno tabelo, če bi želeli, da naš usmerjevalnik uporablja usmerjanje z vektorjem razdalj (distance vector)? Katere od naslednjih lastnosti veljajo za to usmerjanje:
  1. statično
  2. dinamično
  3. globalno
  4. porazdeljeno
  5. po eni poti
  6. po več poteh

9) S programom Wireshark smo zajeli spodnja dva paketa. V čem se razlikujeta? Označite polja protokola omrežne plasti, ki pri obeh protokolih označujeta isto reč. Kdaj uporabimo polje »Fragment offset«, ki je del glave prvega paketa? Kaj je vsebina tega polja?

- ☐ Internet Protocol, Src: 212.235.189.155 (212.235.189.155), Dst: 212.235.189.158 (212.235.189.158)
  - Version: 4
  - Header length: 20 bytes
  - ⊕ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
  - Total Length: 60
  - Identification: 0x2682 (9858)
  - ⊕ Flags: 0x00
  - Fragment offset: 0
  - Time to live: 128
  - Protocol: ICMP (0x01)
  - ⊕ Header checksum: 0x0000 [incorrect, should be 0xef2d]
  - Source: 212.235.189.155 (212.235.189.155)
  - Destination: 212.235.189.158 (212.235.189.158)
  
- ☐ Internet Protocol Version 6
  - ⊕ 0110 .... = Version: 6
  - .... 0000 0000 .... .... .... = Traffic class: 0x00000000
  - .... .... .... 0000 0000 0000 0000 = Flowlabel: 0x00000000
  - Payload length: 40
  - Next header: ICMPv6 (0x3a)
  - Hop limit: 128
  - Source: 2001:1470:ffff::a (2001:1470:ffff::a)
  - Destination: 2001:1470:ffff::10 (2001:1470:ffff::10)

10) Kakšen XML zgradi naslednji fragment kode (rešitev napišite samo za 1 programski jezik, ne za oba!)?

### Java

```
DocumentBuilderFactory fact =
    DocumentBuilderFactory.newInstance();
DocumentBuilder parser = fact.newDocumentBuilder();
Document doc = parser.newDocument();
Element root = doc.createElement("seznam");
doc.appendChild(root);
Element oseba = doc.createElement("oseba");
root.appendChild(oseba);
oseba.appendChild(doc.createTextNode("Kapica"));
oseba.setAttribute("ime", "Rdeča");
Element priimek = doc.createElement("priimek");
root.appendChild(priimek);
```

### Python

```
impl = getDOMImplementation()
doc = impl.createDocument(None, 'seznam', None)
root = doc.documentElement
oseba = doc.createElement('oseba')
root.appendChild(oseba)
oseba.appendChild(doc.createTextNode('Kapica'))
oseba.setAttribute('ime', 'Rdeča')
priimek = doc.createElement('priimek')
root.appendChild(priimek)
```

# B

- 1) Kaj so osebna omrežja (PAN), kakšne značilnosti jih ločijo od ostalih omrežij in kateri standard jih opredeljuje?
- 2) Kdo in s kakšnim odgovorom odgovori na ARP poizvedbo, ki se sproži kot posledica iskanja fizičnega naslova k IP naslovu, ki ne sodi v lokalno omrežje? Pojasnite in narišite sliko – primer takega omrežja.
- 3) Denimo, da na FRI pride napadalec, ki na WC-ju skrivaj postavi lastno in povsem odprto brezžično dostopno točko, ki omogoča dostop do interneta prek napadalčeve omrežne hrbtnice. SSID te točke je LRK-FRI. Kako vam lahko napadalec škodi, če se priključite na LRK-FRI misleč, da je to omrežje za vas postavil asistent pri RK?
- 4) Narišite protokolarni sklad TCP/IP, poimenujte plasti ter navedite naloge in ključne funkcije omrežne plasti.
- 5) Kaj je tuneliranje IPv6 in kako deluje?
- 6) Usmerjevalnik sprejme datagram velikosti 3015 bytov.
  - a) Kakšen je najmanjši možen MTU na izhodni povezavi, da se bo ta datagram fragmentiral na 3 fragmente, torej da še ne bodo potrebni 4 fragmenti?
  - b) Kolikšna bo velikost izhodnih datagramov – fragmentov , če je MTU tak, kot ste ga izračunali v točki a)?
- 7) IP naslovi:
  - a) Vaš računalnik ima v omrežju IPv4 naslov 10.17.3.252/25. Izračunajte naslov omrežja, naslov broadcast, najmanjši in največji naslov naprave v tem omrežju. Izračunajte še koliko naprav lahko priključimo v to podomrežje.
  - b) Označite kateri od spodnjih naslovov IPv6 niso sintaktično pravilni. Označite tudi mesto napake in komentirajte zakaj so po vašem mnenju ti naslovi napačni:
    1. ::1::
    2. 2001:770:10:300::134.226.81.11
    3. 231::0:abc:1:def::1
    4. 2001:1470:fffdg:16:103:22:0:1
    5. 2001:1470:fffd:1234:aabab:37:1:a

- 8) Usmerjanje
  - a) Iz usmerjevalnika smo prebrali spodnjo usmerjevalno tabelo:

Omrežje	Prehod	Vmesnik
192.168.0.0/16		A
10.0.1.0/26		B
192.168.12.64/27		C
10.0.2.0/25		D
0.0.0.0/0	10.0.2.1	D

Na kateri vmesnik (če je podan napišite tudi prehod) bo usmerjevalnik usmeril pakete z naslednjimi ciljnimi naslovi: 10.0.0.1, 192.168.12.1, 10.0.1.123, 10.0.2.3, 192.168.12.70, 192.168.12.240, 192.168.1.1, 193.2.1.66 , 10.0.3.4, 10.1.2.3

- b) Kako bi morali dopolniti usmerjevalno tabelo, če bi želeli, da naš usmerjevalnik uporablja usmerjanje z vektorjem razdalj (distance vector)? Katere od naslednjih lastnosti veljajo za to usmerjanje: statično, dinamično, globalno, porazdeljeno, po eni poti, po več poteh?

- 9) S programom Wireshark smo zajeli spodnji promet. Pomagajte nadobudnem uporabniku razvozlati kaj ta promet pomeni (spodaj imate izpisani še IP glavi 7. in 24. paketa). Pojasnite namen programa, ki tak promet generira in opišite njegovo delovanje.

7	3.638515	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
8	3.639199	212.235.189.129	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
9	3.639521	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
10	3.640194	212.235.189.129	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
11	3.640403	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
12	3.641100	212.235.189.129	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
24	9.194353	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
25	9.194570	193.2.96.106	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
26	9.194953	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
27	9.195114	193.2.96.106	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
28	9.195315	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
29	9.195466	193.2.96.106	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
36	14.748041	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
37	14.748671	193.2.96.1	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
38	14.749084	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
39	14.750810	193.2.96.1	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
40	14.751175	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
41	14.751759	193.2.96.1	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
53	20.473128	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
54	20.474925	194.249.21.197	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
55	20.475284	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
56	20.477211	194.249.21.197	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
57	20.477412	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
58	20.479046	194.249.21.197	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
60	21.487082	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
61	21.487643	212.235.160.234	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
62	21.487984	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
63	21.488463	212.235.160.234	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
64	21.488673	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request
65	21.489165	212.235.160.234	212.235.189.155	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
66	22.501042	212.235.189.155	193.2.1.66	ICMP	Echo (ping) request

Internet Protocol, Src: 212.235.189.155 (212.235.189.155), Dst: 193.2.1.66 (193.2.1.66)	Internet Protocol, Src: 212.235.189.155 (212.235.189.155), Dst: 193.2.1.66 (193.2.1.66)
Version: 4	Version: 4
Header length: 20 bytes	Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)	Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
Total Length: 92	Total Length: 92
Identification: 0x26ad (9901)	Identification: 0x26b0 (9904)
Flags: 0x00	Flags: 0x00
Fragment offset: 0	Fragment offset: 0
Time to live: 1	Time to live: 2
Protocol: ICMP (0x01)	Protocol: ICMP (0x01)
Header checksum: 0x0000 [incorrect, should be 0x3e29]	Header checksum: 0x0000 [incorrect, should be 0x3d26]
Source: 212.235.189.155 (212.235.189.155)	Source: 212.235.189.155 (212.235.189.155)
Destination: 193.2.1.66 (193.2.1.66)	Destination: 193.2.1.66 (193.2.1.66)

- 10) Kakšen XML izpiše naslednji fragment kode (rešitev napišite samo za 1 programski jezik, ne za oba!):

### Java

```

DocumentBuilderFactory fact =
    DocumentBuilderFactory.newInstance();
DocumentBuilder parser = fact.newDocumentBuilder();
Document doc = parser.newDocument();
Element root = doc.createElement("seznam");
doc.appendChild(root);
Element oseba = doc.createElement("oseba");
root.appendChild(oseba);
oseba.setAttribute("ime", "Rdeča");
Element priimek = doc.createElement("priimek");
oseba.appendChild(priimek);

```

### Python

```

impl = getDOMImplementation()
doc = impl.createDocument(None, 'seznam', None)
root = doc.documentElement
oseba = doc.createElement('oseba')
root.appendChild(oseba)
oseba.setAttribute('ime', 'Rdeča')
priimek = doc.createElement('priimek')
oseba.appendChild(priimek)

```