## Sigurnost računalnih sustava

# Mrežna sigurnost

## - prvi dio (IP, UDP, TCP)

doc. dr. sc. Ante Đerek
doc. dr. sc. Stjepan Groš
izv. prof. dr. sc. Miljenko Mikuc
izv. prof. dr. sc. Marin Vuković

# Sigurnost u komunikacijskim mrežama

- U uvodu smo vidjeli definiciju opće sigurnosti
- U kontekstu mreža zanima nas sigurnost:
  - **Komunikacije**, odnosno podataka u prijenosu
  - **Komunikacijskih sustava**, odnosno uređaja koji se upotrebljavaju za pružanje i korištenje komunikacijskih usluga te pohranu informacija

# Model prijetnje na Internetu [RFC3552]

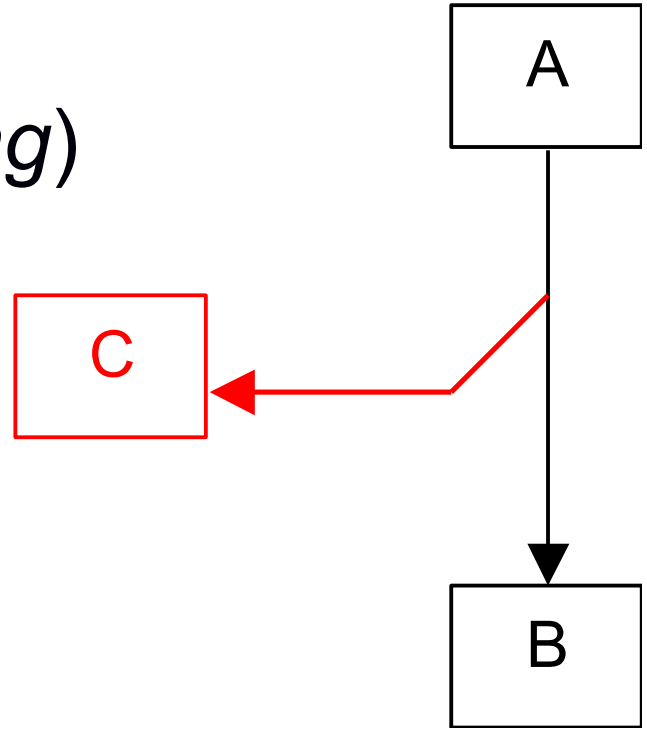
- Podrazumijeva se da krajnje točke komunikacije nisu kompromitirane
- Napadač ima potpunu kontrolu nad komunikacijskim kanalom
- Napadač može čitati bilo koji PDU (engl. Protocol Data Unit)
- Neprimjetno uklanjati, mijenjati ili ubacivati PDU-ove

# Komunikacijska sigurnost

- U nastavku ćemo pogledati prijetnje koje mogu narušiti sigurnost informacija koje razmjenjuju dva čvora, A i B
  - Alternativno ćemo govoriti „sigurnost komunikacije” pod čim se podrazumijeva sigurnost podataka koju razmjenjuju A i B
- Pretpostavljamo da moraju biti ispunjeni svi sigurnosni zahtjevi na podatke koji se razmjenjuju između A i B kako bi se te informacije smatrale sigurnima(!)
  - Svi zahtjevi su: tajnost, raspoloživost, cjelovitost, autentičnost i neporecivost
  - U određenim slučajevima neće biti potrebno zadovoljiti sve zahtjeve, ali onda neke od prijetnji više ne postoje!

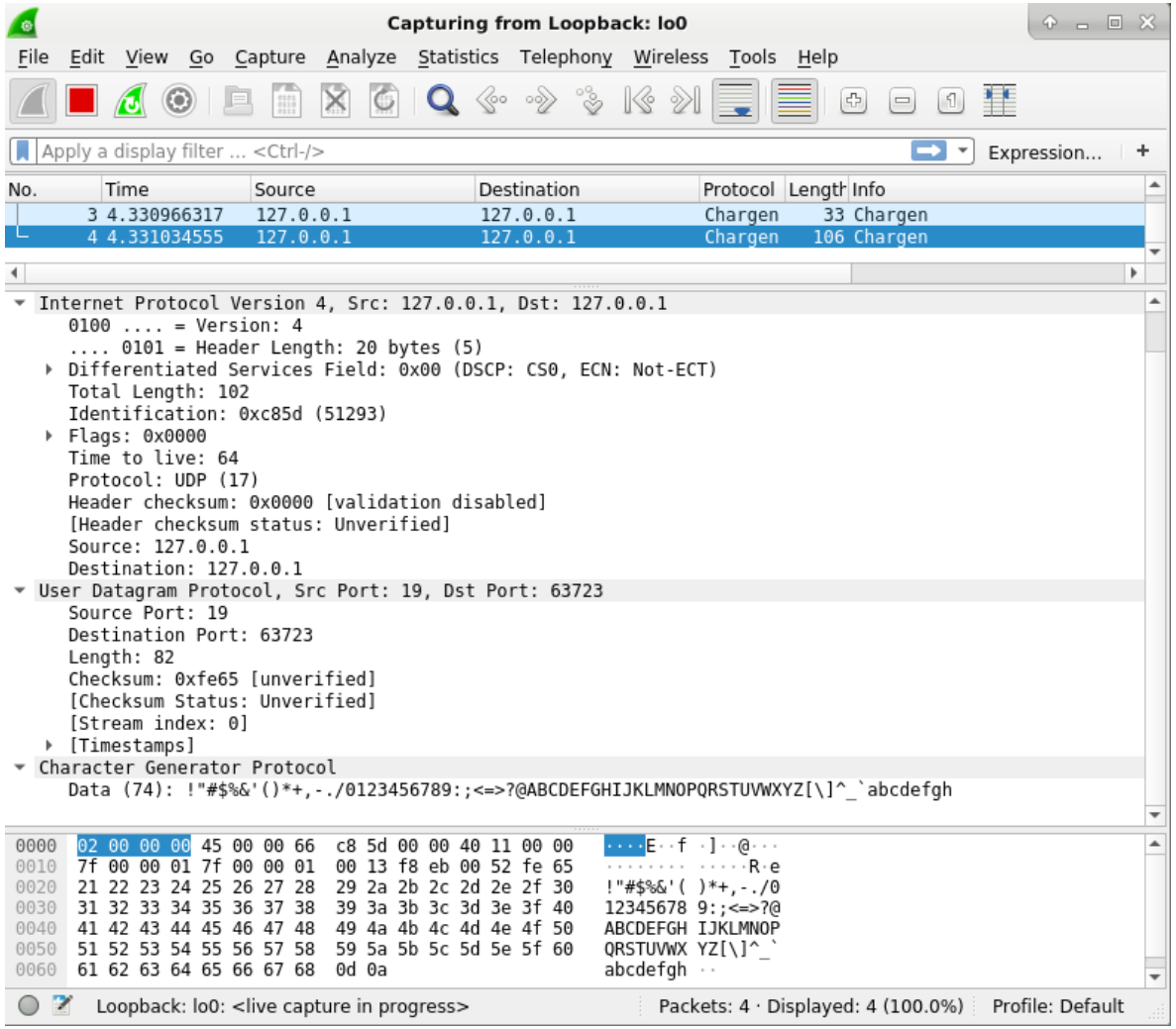
# Presretanje, prisluškivanje

- Presretanje (*interception*), prisluškivanje (*evesdropping*), njuškanje mreže (*network sniffing*), prisluškivanje na vodu (*wiretapping*)
  - elektronička komunikacija se presreće i preuzima informacija
  - Potencijalne štete
    - Neovlaštena uporaba podataka
    - Potencijalno narušavanje privatnosti
- Zakonski regulirano presretanje (*lawfull interception*)



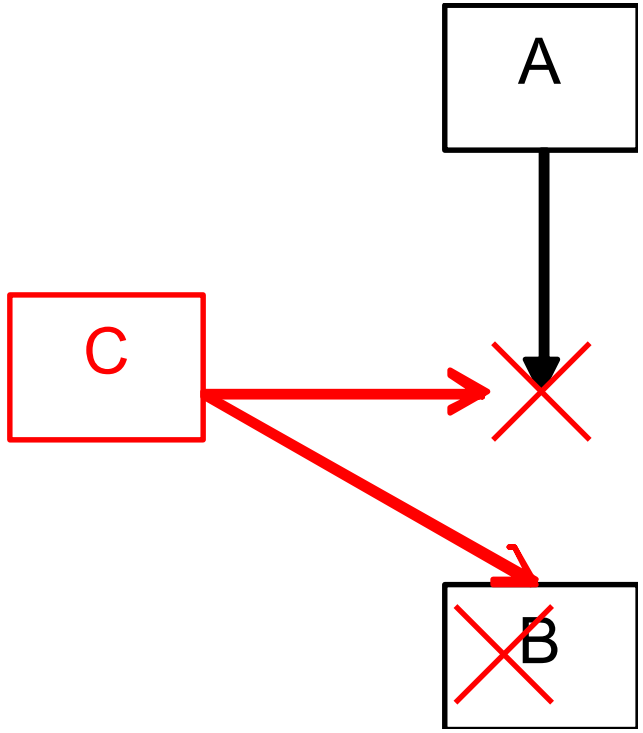
# Network sniffing

- temelj za mnoge napade
  - napadač postavlja svoju mrežnu karticu u promiskuitetni način rada - vidi sav promet na tom segmentu
  - mrežna kartica predaje sve pristigle pakete IP sloju
  - mnogi protokoli prenose autentifikacijske podatke u obliku čistog teksta => username/password itd.
- alati: Wireshark, tcpdump, …



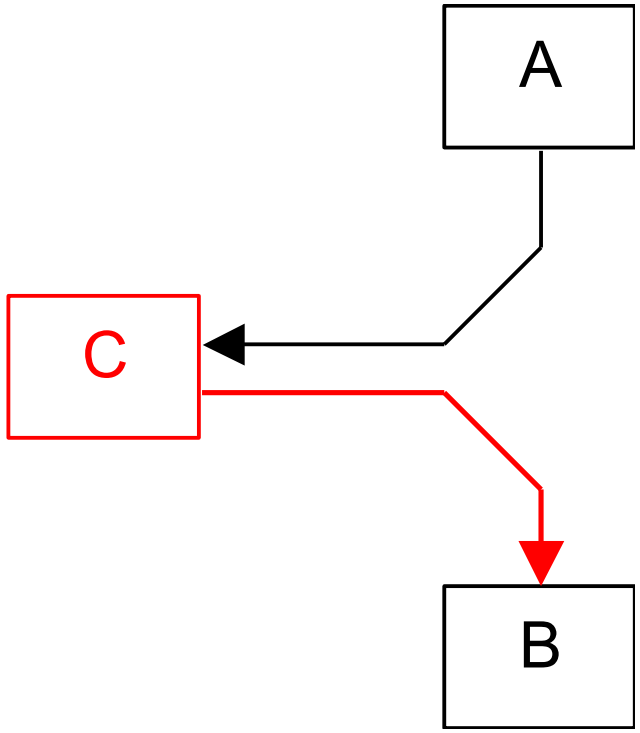
# Prekidanje, uskraćivanje

- Prekidanje (engl. interruption)
  - prekidanje normalnog tijeka komunikacije, usluge ili aplikacije
- Uskraćivanje usluge (engl. denial of service)
  - onemogućavanje usluge izazivanjem preopterećenja mreže ili umreženog sustava



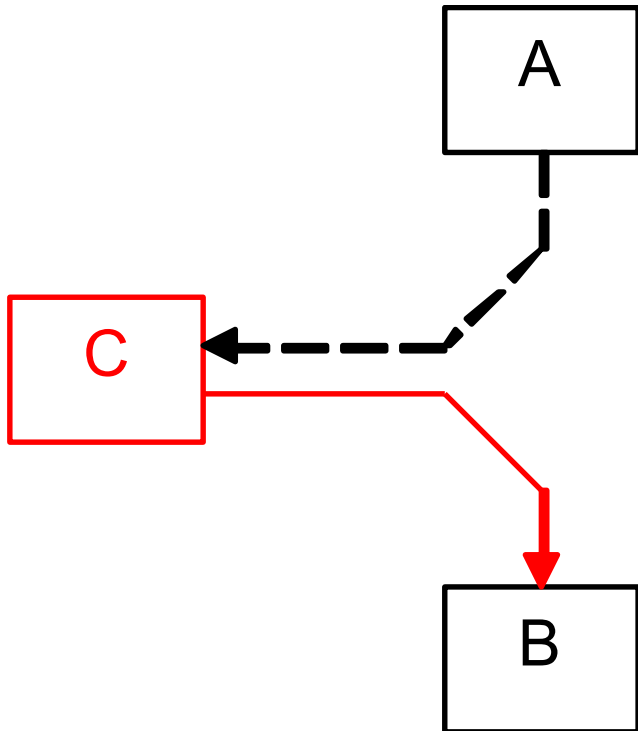
# Promjena, kašnjenje

- Promjena (engl. modification, tampering)
  - Promjena ili uništenje informacije
  - Kašnjenje može izazvati isti učinak – podatak postaje nevažan



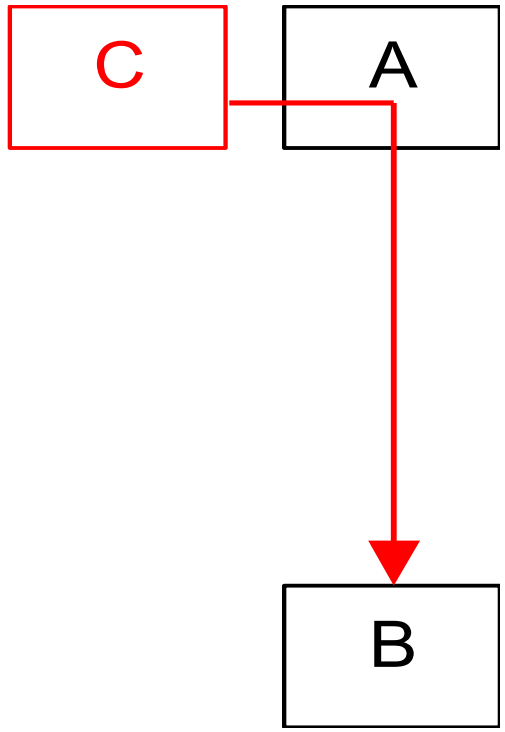
# Umetanje, ponavljanje

- Umetanje, ubacivanje (engl. fabrication)
  - Ubacivanje zlonamjerne  informacije
- Ponavljanje (engl. replay)
  - Ubacivanje informacije prethodno preuzete presretanjem



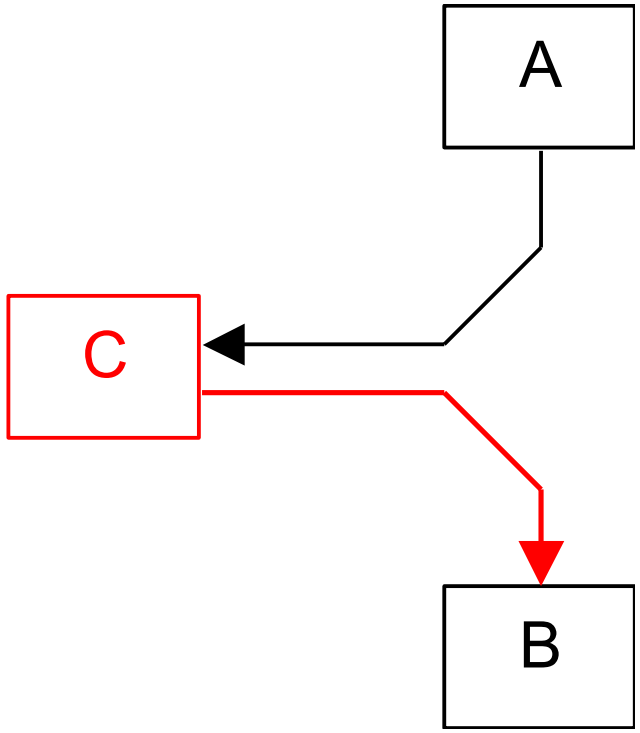
# Lažno predstavljanje

- Lažno predstavljanje
  - Maskiranje (engl. masquerade)
  - Lažno predstavljanje (engl. impersonation)
  - Preuzimanje identiteta i uloge korisnika



# „Čovjek u sredini”

- Često se u kontekstu komunikacija govori o napadu „čovjek u sredini”
  - engl. Man in the Middle, MITM
- To je situacija u kojoj su prisutne sve prethodno spomenute prijetnje
  - Kako bi sve navedene prijetnje bile ostvarive napadač se mora nalaziti negdje na putu kojim se prenose podaci
- Najbolji položaj za napadača i najgori za branitelja



# Gdje se nalaze ranjivosti?

- Fizička ranjivost
- Ranjivosti u protokolima
  - Ako pojedini protokol ima ranjivost, tada sve implementacije imaju tu ranjivost
- Ranjivosti u implementacijama
  - Protokol nema ranjivost već je ona posljedica nekakve pogreške u implementaciji
  - Ako ima više nezavisnih implementacija neće biti sve ranjive
- Ranjivosti u konfiguraciji i korištenju
  - Ranjivost koja je specifična za pojedinu okolinu

# Ranjivosti protokola

# Ranjivosti protokola

- Dosta protokola Interneta je nastalo u vrijeme kada sigurnost nije bila visoko na listi prioriteta
- Primjeri ranjivih protokola: ARP, Ethernet, IP, TCP, UDP, niz aplikacijskih protokola
- Navedeni protokoli su ranjivi jer nemaju mehanizama uz pomoć kojih bi se jamčilo očuvanje sigurnosnih zahtjeva

# Ranjivost protokola ARP

- ARP - protokol za pretvaranje 32 bitnih IP adresa u 48 bitne Ethernet (MAC) adrese
- ako računalo A želi poslati IP datagram računalu B ili usmjeritelju u lokalnoj mreži, tada ono mora znati njegovu MAC adresu
- A šalje broadcast ARP zahtjev na mrežu (uključujući svoje preslikavanje)
- B odgovara računalu A, porukom ARP odziv
- preslikavanje se lokalno pohranjuje u svakom računalu u ARP cache: `arp -an`

# Ranjivost protokola ARP

<table>
<tr><td colspan="2">tip hardvera (2 okteta)</td><td>tip protokola (2 okteta)</td></tr>
<tr><td>duljina hw adrese<br>(1 oktet)</td><td>duljina prot. adr.<br>(1 oktet)</td><td>kod operacije<br>(2 okteta)</td></tr>
<tr><td colspan="3">hardverska (ethernet, MAC) adresa pošiljatelja (6 okteta)</td></tr>
<tr><td colspan="3">IP adresa pošiljatelja (4 okteta)</td></tr>
<tr><td colspan="3">hardverska (ethernet, MAC) adresa primatelja / cilja (6 okteta)</td></tr>
<tr><td colspan="3">ciljna IP adresa (4 okteta)</td></tr>
</table>

- ovisno o tipu poruke, određena polja su prazna:
  - za ARP: odredišna HW adresa,
  - za RARP: sve osim izvorišne HW adrese.

# Napad na ARP

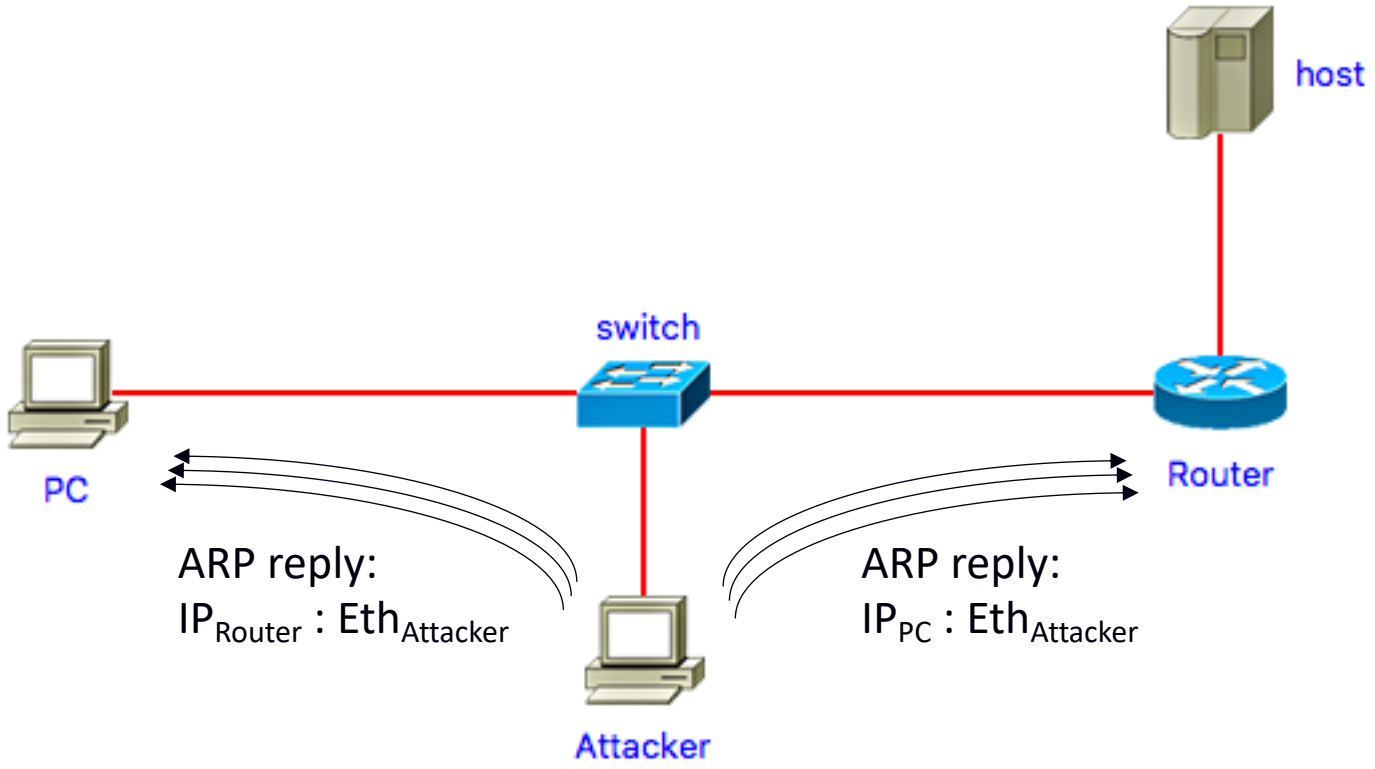
- ARP nema ugrađene mehanizme autentifikacije
- moguće je poslati odgovor prije pravog računala te vratiti lažno preslikavanje adresa (IP/HW)
- lažni ARP odgovori mogu se koristiti za spremanje krivih ARP preslikavanja na računalu kome su upućeni
- u oba slučaja dolazi do preusmjeravanja prometa
- ARP poruke mogu se slati kontinuirano kako bi se (lažni) podaci zadržali u *cacheu*

# Napad na ARP

- Cilj:
  - Prisluškivanje prometa (prospojnik (switch) prosljeđuje ethernet okvire između mrežnih sučelja na temelju ethernet adrese odredišta)
  - Prekidanje: lažno preslikavanje IP adrese usmjeritelja na nepostojeću MAC adresu (DoS napad)
  - Promjena
  - Ometanje

- alati: arpoison, ettercap, dsniff, parasite

# Napad na ARP




sysctl net.ipv4.ip_forward = 1

# Napad na ARP - otkrivanje i zaštita

- ako je preuzimanje bilo uspješno, malo je vjerojatno da će korisnici napadnutog računala išta primijetiti
- najjednostavniji način: ispis ARP *cachea*
  - ako se MAC adresa od određene IP adrese promijenila napadačevo računalo se identificira s pomoću te MAC adrese te po mogućnosti fizički odspaja s mreže
- može se detektirati s nekog trećeg računala, na kojem se njuška mreža i traže lažni ARP odzivi
- u slučaju DoS napada, lagano je ustanoviti da nešto nije u redu

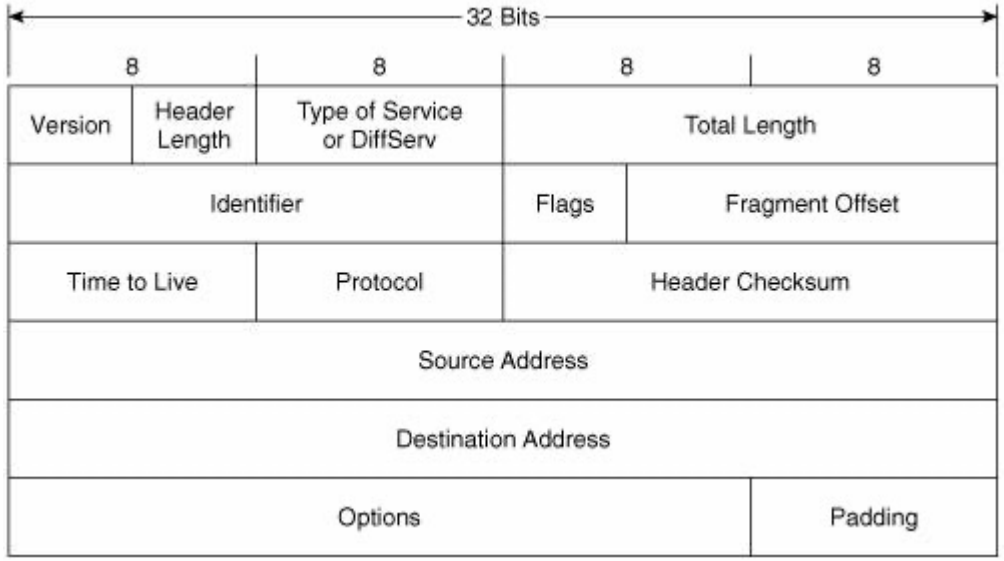
# Napad na ARP - zaštita

- ako postoji neobično ponašanje u mreži korisno je pogledati ARP *cache*
- korištenje hardvera koji će učiniti takve napade nemogućima ili više vidljivima
  - korištenje komutatora, *switch*, s mogućnošću “zaključavanja” priključaka (*port security*)
- onemogućavanje ARP-a i njegova ručna konfiguracija

# IP

# Ranjivosti protokola IPv4

- Podaci koji se prenose nisu ni na koji način zaštićeni!
  - Laka izmjena pojedinih polja paketa
  - Najčešće lažiranje izvorišnih IP adresa (ne koriste se za usmjeravanje!)
- Izrazito velik sigurnosni problem

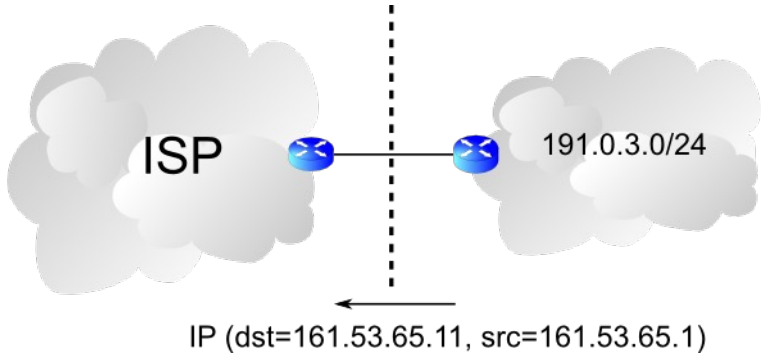


# IP zavaravanje (engl. IP spoofing)

- Slanje IP datagrama s lažnom adresom pošiljatelja
  - Najčešće se zloupotrebljava u (D)DoS napadima
- Slanje IP datagrama s lažnom adresom pošiljatelja koju primatelj smatra sigurnom
  - Zaštita:
    - Filtriranje neispravnih izvorišnih adresa
    - Zabrana korištenje IP adrese za autorizaciju i zabrana nekih servisa (onemogućavanje svih r* naredbi: rlogin, rcp, rsh)
    - Šifriranje cijelog mrežnog prometa
- „State of IP Spoofing” (https://spoofer.caida.org/summary.php)

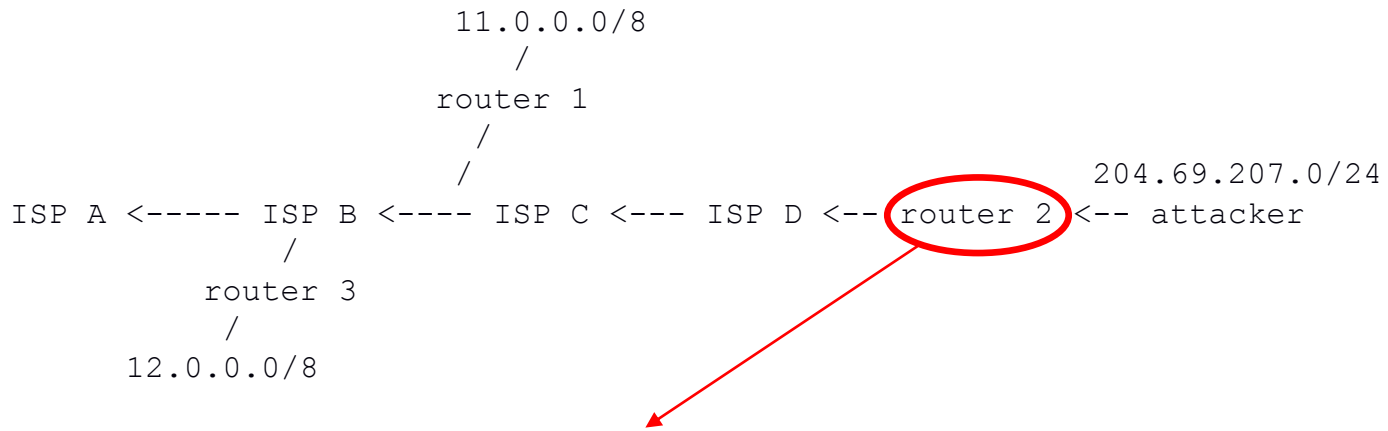
# Filtriranje neispravnih izvorišnih adresa

- Problem paketa s neispravnim izvorišnim adresama bi se djelomično riješio ispravnim podešavanjem usmjernika (RFC 2827)




- Usmjernici bi trebali filtrirati neispravne adrese
  - Međutim, to dosta često nije napravljeno
  - Posljedica: (D)DoS napadi
- Privatne IP adrese također moraju biti filtrirane
  - 10.0.0.0/8, 127.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16, nealocirane IP adrese

# RFC 2827 – "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing"




- ISP D na ruteru 2 filtrira dolazne pakete od napadača (*attacker*):

```
IF    packet's source address from within 204.69.207.0/24
THEN  forward as appropriate

IF    packet's source address is anything else
THEN  deny packet
```

- a napadač ("*attacker*"), za svoju zaštitu, može filtrirati sve dolazne pakete iz Interneta s izvorišnom adresom 204.69.207.0/24

# “Nonroutable” mrežne adrese

- `0.0.0.0/8` – na primjer DHCP “broadcast request”
- `127.0.0.0/8` – na primjer localhost 127.0.0.1
- `169.254.0.0/16` – DHCP “autoconfigure”
- `192.0.2.0/24` – "TEST-NET," alocirano za primjere i dokumentaciju
- `198.18.0.0/15` – RFC 2544, testiranje performansi
- `224.0.0.0/4` – multicast adrese, filtrirati ako se ne koriste
- `240.0.0.0/4` – stara klasa E

# IP fragmentacija

- Fragmentacija je obavezan dio IP protokola; kad je potrebno datagram podijeliti na manje dijelove prije učahurivanja u okvir podatkovne veze (duljina IP datagrama > MTU)
  - Moguće na izvorišnom računalu i na svakom usmjeritelju na putu do odredišta
- Svaki fragment se dostavlja nezavisno
- Može zavarati neke vatrozide i sustave za detekciju uljeza

# IP fragmentacija

- Datagram se sastavlja na krajnjem odredištu
  - svaki fragment se usmjerava nezavisno
- Svi fragmenti imaju isti identifikacijski broj (IP ID)
- Pomak (*fragment offset*) određuje smještaj fragmenta u sastavljenom datagramu
- Zastavica "*more fragments*" postavljena je u svim fragmentima osim u zadnjem

# IP fragmentacija - primjeri napada

- „Ping of Death” (1996.)
  - DoS (“Denial of Service”) napad koji prekoračuje maksimalnu veličinu IP datagrama
  - kreira se i šalje fragmentirani IP datagram ukupne duljine veće od 65535 okteta
  - teoretski bilo koji IP paket, ali obično baš “ICMP echo request”
  - *“Fragment Offset”* je takav da je ukupna veličina zapakiranog datagrama veća od maksimalno dozvoljene veličine: → *buffer overflow, kernel panic*
- „Teardrop” (Linux kernel < 2.0.32)
  - napadač šalje dva fragmenta koji se djelomično prekrivaju
  - “crash” kernela nakon sastavljanja fragmenata.
- Ima i novijih:
  - search “IP fragmentation”: https://cve.mitre.org/cve/search_cve_list.html

# IP fragmentacija - primjeri napada

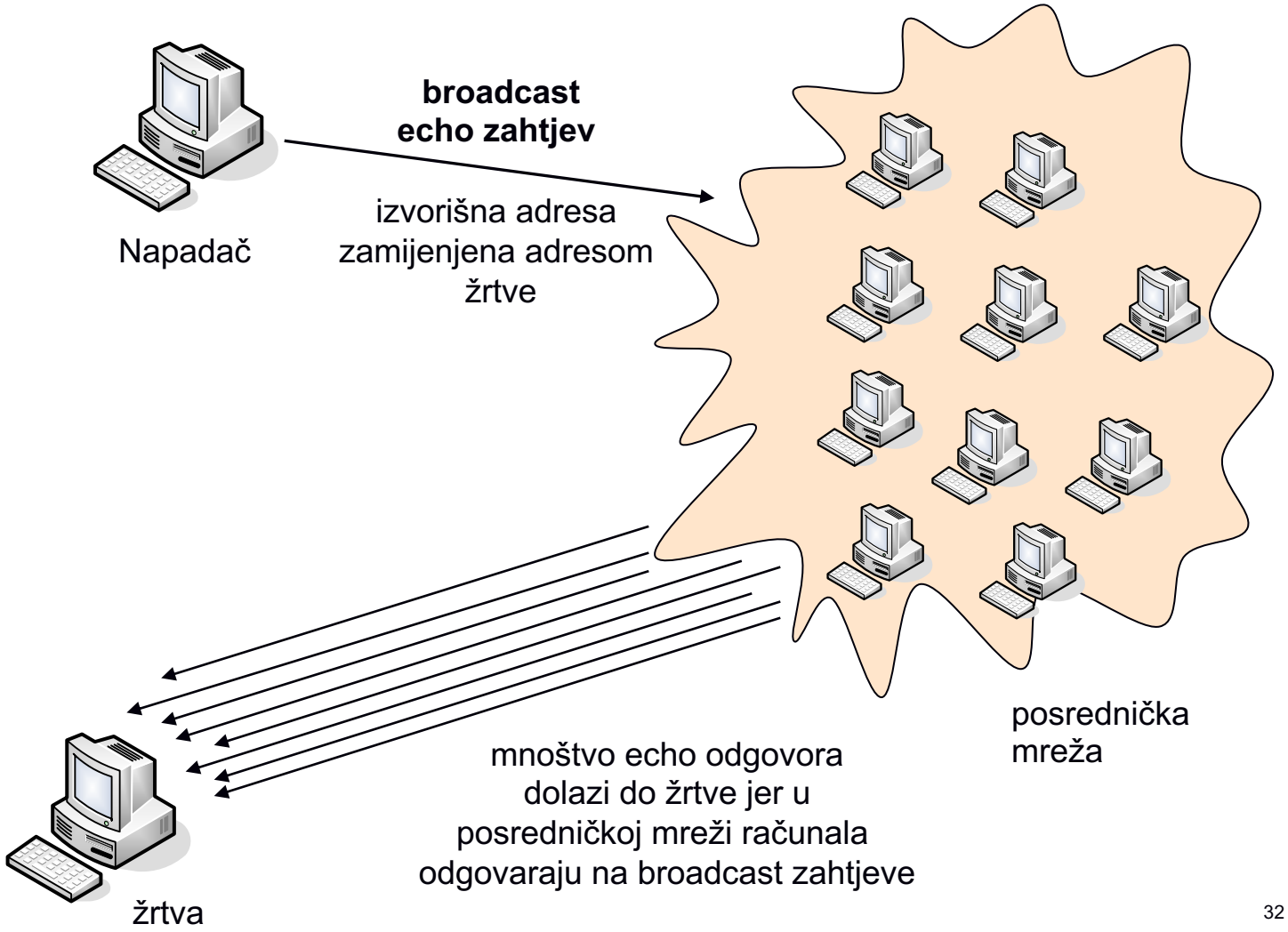
- "*TCP overwrite*"
  - varijacija napada "*teardrop*"
  - nije napad tipa DoS već se pokušava prevariti vatrozid
  - IP datagram se fragmentira, TCP zaglavlje sadrži dozvoljeni port, na primjer 80, pa ga vatrozid propušta
  - neki sljedeći fragment ima „pomak" postavljen na 1 što znači da će port biti prepisan (npr. novi port će biti 23), sastavljeni paket preusmjerava se na novi port
  - vatrozid treba provjeravati minimalni pomak fragmenta!

# Ranjivosti i zloupotrebe protokola ICMP

- U pravilu se radi o DoS napadima
- Iskorištavanje tipa „ICMP redirect” za zlonamjerno preusmjeravanja prometa
- Uskraćivanje usluge slanjem lažiranih ICMP poruka o nedostižnom odredištu
- Implementacija prikrivenih kanala (engl. covert channel) korištenjem ICMP poruka
- Enumeracija računala na mreži

# Ranjivosti i zloupotrebe protokola ICMP

- napad “smurf”
  - započinje slanjem echo zahtjeva na sveodredišnu (“broadcast”) adresu posredničke mreže s lažiranom izvorišnom adresom jednakom adresi ciljne mreže (žrtve)
  - računala u posredničkoj mreži odgovaraju slanjem echo odziva
  - odgovori idu na adresu žrtve
  - posrednička mreža i ciljna mreža zagušene prometom
  - napad se pojačava slanjem zahtjeva na različite posredničke mreže



# Protokol DHCP

- Služi za automatsku dodjelu adresa i mrežnih parametara
  - Klijent šalje svima na mreži poruku DHCPDISCOVER
    - Klijent u tom trenutku ne zna adresu
  - Poslužitelji odgovaraju klijentu s porukom DHCPOFFER
  - Klijent odabire poslužitelj i šalje svima DHCPREQUEST
  - Poslužitelj odgovara s DHCPACK
- Fiksiranje adresa na temelju MAC adrese radi kontrole pristupa
  - Moguće i na temelju identifikatora

# Problemi protokola DHCP

- Nema nikakve zaštite poruka
  - Bilo tko može slati i primati DHCP poruke
- Lažni DHCP poslužitelji na mreži
  - Napadi uskraćivanja usluga
  - Preusmjeravanje prometa
- Bilo koji klijent može zatražiti parametre
  - Lako se zaobilazi MAC/ID zaštita
  - Moguće iscrpljivanje svih raspoloživih adresa („DHCP Starvation attack”)

# IPv6

# Protokol IPv6

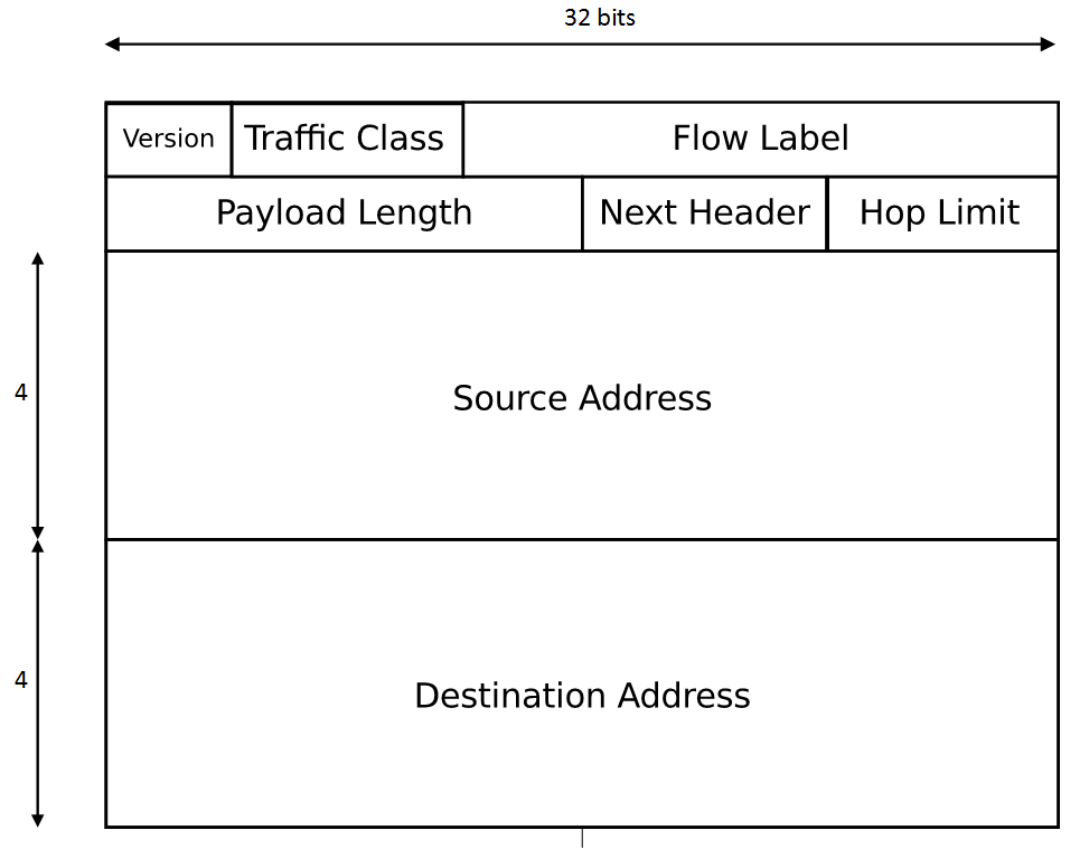
- Zašto uopće spominjati sigurnost protokola IPv6?
  - Na razini RIR-ova IPv4 adrese su iscrpljene te je neminovno uvođenje IPv6
  - Značajni napori da se popularizira i uvede IPv6
- Dosta operacijskih sustava već ima omogućen IPv6
  - Windows OS počevši sa Windows Vista u podrazumijevanoj konfiguraciji ima omogućen IPv6
  - Linux / BSD / macOS već dugo vremena imaju omogućen IPv6 u standardnoj konfiguraciji
- Koliko ljudi je toga svjesno?

# Cilj razmatranja protokola IPv6

- Upozoriti na činjenicu kako je IPv6 drugačiji od IPv4
  - IPv6 nije radikalno drugačiji, ali je dovoljno da ima svojih specifičnosti
- Nedostatak operativnog iskustava
- Upoznati se s ranjivostima
  - Specifičnim za protokol IPv6
  - Koje ranjivosti su zajedničke
  - Kojih ranjivosti više nema u odnosu na protokol IPv4

# Promjene u protokolu IPv6 u odnosu na IPv4

- Osnovne izmjene u protokolu IPv6
  - Adrese su 128 bita
  - Pojednostavljeno zaglavlje
  - Fragmentacija se više ne provodi u mrežnom sloju
  - ARP se više ne koristi
  - Automatsko podešavanje mrežnih parametara
- Zaglavlje protokola IPv6 i dalje nema zaštite!



# IPv6 adrese

- Adrese su 128 bitne
  - Pišu se u 8 grupa po 16 bita, svaka grupa 4 heksadecimalne znamenke
- Primjeri
  - 1234:5678:9abc:def0:1234:5678:9abc:def0
  - 1234:5678:0000:0000:1234:0000:0000:def0
  - 1234:5678:0:0:1234:0:0:def0, 1234:5678::1234:5678:0:def0,
  - 1234:5678:0:0:1234::def0
- Klase adresa
  - lokalne (link local), globalne, višeodredišne (multicast)
  - „Anycast” podskup globalnih adresa

# Ranjivosti kojih više nema u IPv6

- Odnosno, one koje su umanjene.
- Skeniranje IPv6 mreža je otežano, ali
  - Postoje specifične adrese:
    - Svi čvorovi: FF01::1, FF02::1
    - Svi usmjernici: FF01::2, FF02::2
    - All DHCP agents: FF02::1:2
  - Korisnici će vjerojatno dodjeljivati lako pamtljive adrese uređajima
- Ne koriste se više broadcast adrese
- Onemogućena je fragmentacija u usmjernicima

# Ranjivosti zajedničke protokolima IPv4 i IPv6

- Skeniranje jedne adrese je i dalje moguće
- Razrješavanje IP adresa u MAC adresu
  - Ne koristi se više ARP već ICMPv6, ali sve je ostalo isto
- Protokoli ICMPv4 i ICMPv6 i dalje ranjivi
- Protokol DHCP se i dalje koristi u obje mreže
- Protokol IPsec se koristi za zaštitu oba protokola
  - Krivo se ponekad kaže kako je IPv6 sigurniji od IPv4
  - Jedina razlika je što u normama piše da se mora implementirati IPsec ako implementacija želi biti uskladiva s IPv6 normom
  - Ali, implementacije za IPv4 i IPv6 jednako podržavaju IPsec

# Ranjivosti specifične za protokol IPv6 (1)

- Samostalno podešavanje IPv6 adrese
  - Obavlja se na temelju MAC adrese.
  - Problem s privatnošću.
    - Mogućnost slučajno generiranih IPv6 adresa, ali one su također problematične!
- Problem velikog adresnog prostora
  - Teško je kontrolirati tko koristi koju adresu.
  - Problem za sigurnosne stijene jer potencijalno trebaju čuvati puno podataka.
- Višeodredišne adrese
  - Lako propitivanje za pojedinim uređajima na lokalnoj mreži

# Ranjivosti specifične za protokol IPv6 (2)

- Zloupotreba mehanizma DAD (Duplicate address detection) radi uskraćivanja usluge
- Objava usmjerničkih podataka
  - Napadač lakše dolazi do informacija potrebnih za spajanje
  - Može slati lažirane objave usmjerničkih podataka
- Nedostatak operativnog iskustva
- Automatsko tuneliranje
  - Uvedeno radi tranzicije s IPv4 na IPv6
- Sigurnosni uređaji još nisu dovoljno sazreli

# Protokol ICMPv6

- Vrlo značajan za ispravan rad protokola IPv6
  - Daleko značajniji no što je to bio ICMPv4 za IPv4
- Posljedično, nije moguće filtrirati sav ICMPv6 promet
  - Mreža neće raditi
  - Razrješavanje IPv6 u MAC adresu, autokonfiguracija, …

# Poboljšanje sigurnosti na mrežnom sloju

- Protokol IP ne nudi nikakvu zaštitu
  - Može se lažirati, sadržaj lako čitljiv
  - Kako sigurno povezivati lokalne mreže i udaljene korisnike?
- Opcije na mrežnom sloju
  - Kriptiranje i zaštita integriteta
    - Virtualne privatne mreže (engl. Virtual Private Networks, VPNs)
- Za potpunu zaštitu preporučljivo je koristiti i (komplementarna) rješenja na višim slojevima
  - TLS/HTTPS/SSH ili neka druga metoda kriptiranja i autentikacije
  - Moguće je navedene protokole koristiti i bez zaštite u mrežnom sloju!

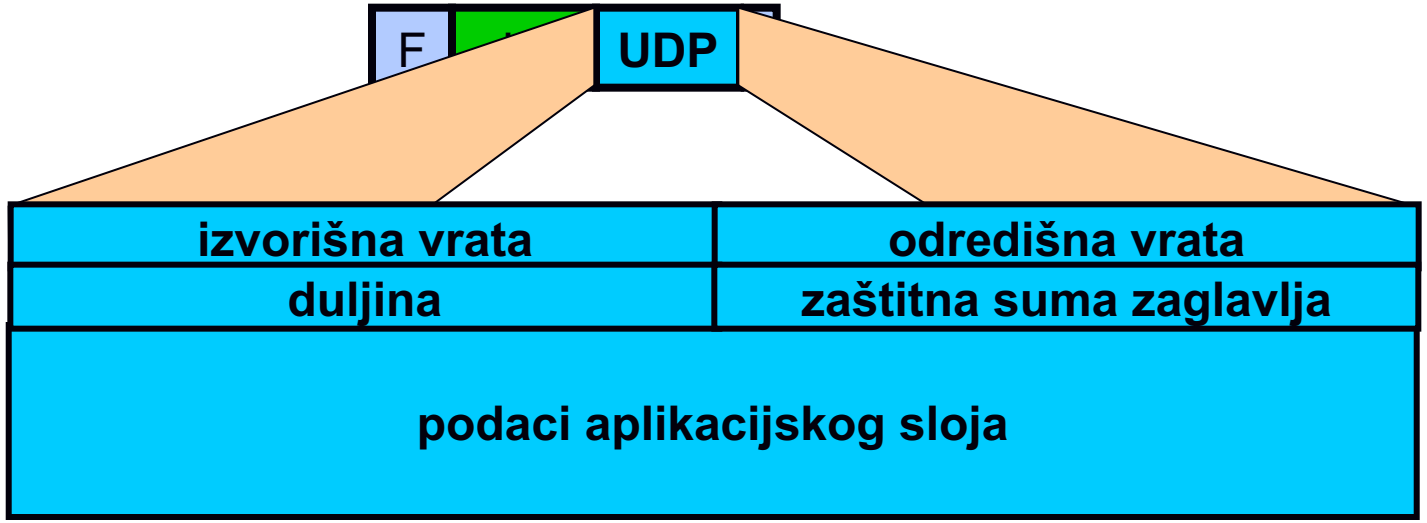
# UDP

# Protokol UDP

- Nespojni transportni protokol
- Nema ugrađene mehanizme za pouzdan prijenos
- Nema kontrole toka
- Često se koristi za prijenos višemedijskih podataka (efikasniji je od TCP) i za usluge temeljene na principu zahtjev / odziv (DNS, NIS, NFS, RPC)

# Protokol UDP

- duljina UDP zaglavlja: 8 okteta



# Napadi na UDP

- UDP obmana - (UDP *spoofing*)
  - mijenjanjem izvorišne IP adrese “predstavljamo se” kao drugo računalo
  - IP adresa je jedini način identifikacije računala u protokolu UDP
  - ne šalju se potvrde
- UDP otimanje - (UDP *hijacking*)
  - napadač sluša vezu
  - odgovara na klijentov UDP zahtjev prije poslužitelja slanjem paketa s promijenjenom izvorišnim adresom
  - klijent misli da je primio paket od poslužitelja
  - nema identifikacije paketa

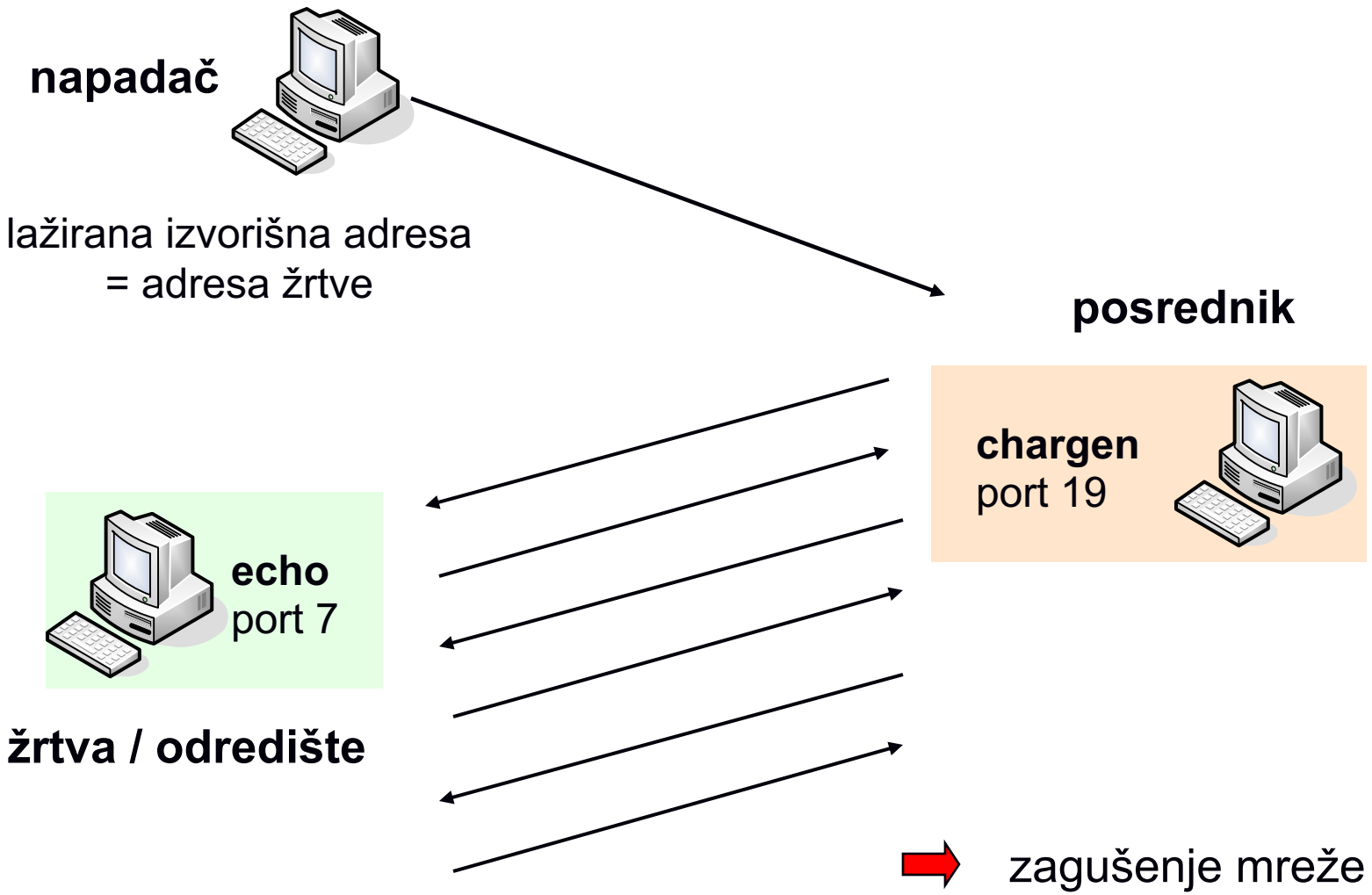
# Napadi na UDP

- UDP oluje - (UDP *storms*)
  - jedan paket je dovoljan za pokretanje napada!
  - obično se pošalje nekoliko paketa kako bi se pojačalo djelovanje
  - može se koristiti bilo koji servis koji automatski odgovara na primljeni UDP datagram: echo (7), chargen (19), daytime (13), time (37), …
  - i usmjeritelji često podržavaju nekoliko dijagnostičkih usluga
  - petlja se izvodi dok jedno računalo ne završi (može biti potreban i reboot)

# UDP Small Services

| Naziv | *Port* | Opis usluge |
|---|---|---|
| echo | 7/udp | server echoes the data that the client sends |
| daytime | 13/udp | server returns the time and date in a human readable format |
| chargen | 19/udp | server responds with a datagram containing a string of ascii characters |
| time | 37/udp | server returns the time as a 32-bit binary number |
| | | |

# UDP storm / UDP flood



# *UDP amplification, UDP reflection*

- servisu koji koristi UDP (bez autentifikacije) pošalje se upit s lažiranom izvorišnom adresom a njegov odziv sadrži više podataka od upita
- “UDP-Based Amplification Attacks”
  - https://www.us-cert.gov/ncas/alerts/TA14-017A

# Primjeri *UDP amplification*, *UDP reflection*

- "DNS amplification" - 28 do 54 puta
- "NTP amplification" – 556.9 puta
  - Network Time Protocol – protokol za sinkronizaciju vremena
  - loša konfiguracija omogućava slanje upita poslužitelju o posljednjih 600 sinkroniziranih računala
- "SNMP amplification" – teoretski do 650 puta
- SSDP - 30.8 puta
- CharGEN - 358.8 puta

# Primjeri *UDP amplification*, *UDP reflection*

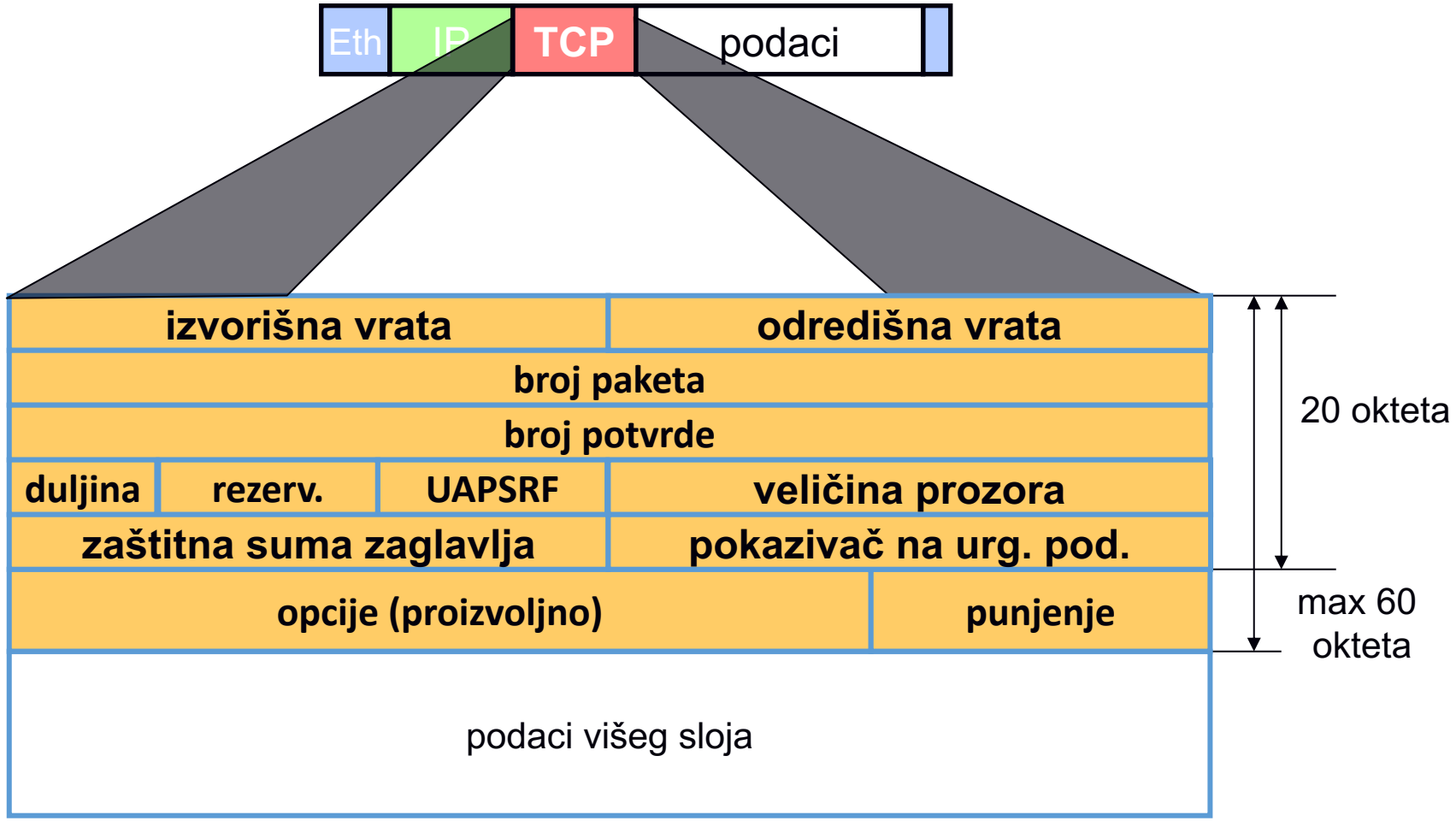
- akamai's [state of the internet] / security Q2 2016 report
  - https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/akamai-q2-2016-state-of-the-internet-security-report.pdf
  - https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/q2-2017-state-of-the-internet-security-report.pdf
- DDoS korištenjem DNS
  - 2012. godine, 65 Gbit/s
  - 2013. godine, 300 Gbit/s
- DDoS korištenjem NTP (Network Time Protocol)
  - 2014. godine, 100 Gbit/s
  - 2014. godine, 400 Gbit/s (CloudFlare)

# TCP

# Protokol TCP

- Konekcijski (spojno) orijentirani transportni protokol
- Pouzdan
  - istek vremenske kontrole (*timeout*) i retransmisija
  - potvrde
  - nema dupliciranja
  - slaže pakete
  - kontrolni zbroj
  - omogućuje kontrolu toka
- Obostrana veza

# Format TCP segmenta



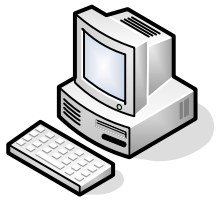
# Slijedni brojevi i brojevi potvrde

- SEQ
  - slijedni broj (“*Sequence number*”)
  - označava redni broj prvog okteta koji se prenosi u korisničkim podacima
- ACK
  - broj potvrde (“*Acknowledgment number*”)
  - označava redni broj okteta koji pošiljatelj ove potvrde očekuje primiti
  - ujedno potvrđuje da su svi podaci do tog okteta primljeni

# TCP zastavice

- SYN dogovaranje početnih brojeva pri uspostavi veze
- FIN završeno slanje podataka
- ACK broj potvrde je postavljen
- URG postavljen je *urgent pointer*
- PSH primatelj treba predati podatke aplikaciji što je prije moguće
- RST resetira vezu

# Primjer TCP sjednice



klijent
(port = 33623/tcp)

SYN

SYN - ACK

ACK

[ACK postavljen u ostatku sjednice]

[razmjena podataka]

poslužitelj
(port = 80/tcp)

FIN ACK

ACK

FIN ACK

ACK

I klijent i poslužitelj mogu inicirati zatvaranje konekcije

# Slanje podataka i kontrola toka

- u paketu se šalje potvrda o zadnjim ispravno primljenim podacima
- paket se prihvaća samo ako je unutar veličine predajnog prozora (“transmission window”)
- za potvrdu se može koristiti i prazni segment (segment bez podataka)
- paketi sa zastavicama SYN ili FIN povećavaju slijedni broj iako ne sadrže podatke
- protokol kliznog prozora (“*Sliding Window Protocol*”)
  - omogućava slanje više paketa prije nego što dođe potvrda o prispijeću paketa
  - veći protok podataka u odnosu na protokol “stop-and-wait”

# Napad TCP SYN flood (1)

- Poslužitelj po primitku SYN segmenta rezervira resurse
  - veza je u poluotvorenom stanju koje traje neko vrijeme
  - dopušten je samo određen broj poluotvorenih veza
- "Problem" za napadača
  - Računalo koje primi SYN+ACK, a nije poslalo SYN, odgovara s RST
  - Napadač mora koristiti adresu s koje neće stići odgovor!

# Napad TCP SYN flood (2)

```
$ netstat -anf inet
Proto Recv-Q Send-Q Local Address          Foreign Address        (state)
tcp4       0      0 10.0.1.10.22          192.168.1.182.11008    SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.225.28014    SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.175.44828    SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.184.28987    SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.0.10303      SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.237.25561    SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.186.48231    SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.53.20148     SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.14.60914     SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.68.35857     SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.74.57236     SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.156.2794     SYN_RCVD
tcp4       0      0 10.0.1.10.22          192.168.1.217.59919    SYN_RCVD
. . .
```

- CERT® Advisory CA-1996-21 TCP SYN Flooding and IP Spoofing Attacks
- Defining Strategies to Protect Against TCP SYN Denial of Service Attacks
  - https://www.cisco.com/c/en/us/support/docs/ip/ip-multicast/14760-4.html

# Napad TCP SYN flood (3)

- Ne postoji standardizirana niti potpuna zaštita
- Neke metode zaštite
  - Povećanje broja dozvoljenih poluotvorenih veza
  - Skraćenje trajanja poluotvorene veze
  - Smanjenje količine stanja poluotvorene veze (SYN cache)
- Zaštita uz pomoć kolačića (SYN cookies)
  - Za inicijalni SYN se uopće ne čuva stanje
    - Stanje se rekonstruira iz završnog odgovora
  - “kolačić” je posebno odabran 32 bitni slijedni broj
    - ISN klijenta, MSS klijenta, vremenski brojač, adresa i pristup

# Napad TCP SYN flood (4)

- Zaštita uz pomoć kolačića (nastavak)
  - Problem nedovoljne količine prostora u TCP zaglavlju
  - Niz opcija nije podržan, primjerice skaliranje prozora, različite veličine MSS-a (3 bita)
- SYN napad je lekcija za sve novije protokole
- bang.c napad – varijacija na temu SYN napada
- Amplificirani napad
  - Poslužitelju se šalje SYN segment s lažiranom adresom žrtve
  - Server obavlja retransmisiju SYN+ACK segmenta

# Primjer TCP DDoS napada

- “Record-breaking DDoS reportedly delivered by >145k hacked cameras”
  - http://arstechnica.com/security/2016/09/botnet-of-145k-cameras-reportedly-deliver-internets-biggest-ddos-ever/
  - rujan 2016.
  - 145607 kamere / dvr (1-30 Mbps po IP)
  - > 1.5 Tbps DDoS
  - tcp/ack, tcp/ack+psh, tcp/syn

# Napadi na protokol TCP

- Za napad je bitan položaj napadača
  (za napad SYN flood nije!)
- Na putu kojim prolaze TCP segmenti (engl. on path)
  - MITM napad, napadač može pratiti i mijenjati komunikaciju
  - Jedina **potpuna** zaštita je IPsec
  - TLS ne štiti od napada uskraćivanja usluge
- Van puta kojim prolaze TCP segmenti (engl. off path)
  - Napadač ne može pratiti komunikaciju i mora pogađati određene parametre
  - Manje mogućnosti napada, ali s propusnošću veze raste prijetnja

# RST i FIN napadi na protokol TCP

- Prema TCP specifikaciji po primitku ispravnog RST segmenta potrebno je raskinuti vezu
- RST napad
  - Slanje segmenta s postavljenom RST zastavicom
  - Problem je pogoditi parametre TCP veze
    - slijedni broj (unutar prozora!), izvorišna i odredišna IP adresa, izvorišni i odredišni pristup (port)
    - 1Mbps, MSS=1500, 100ms latencije, 15 skokova, WIN 35000
    - 1:57000, 40 okteta RST, 20s (na 1Mbps)
    - za 16384 pristupa, 91 sat
- FIN napad
  - Sličan RST napadu jedino se zatvara pojedini kraj veze