

Yhteenveto

CSE-C2400 Tietokoneverkot

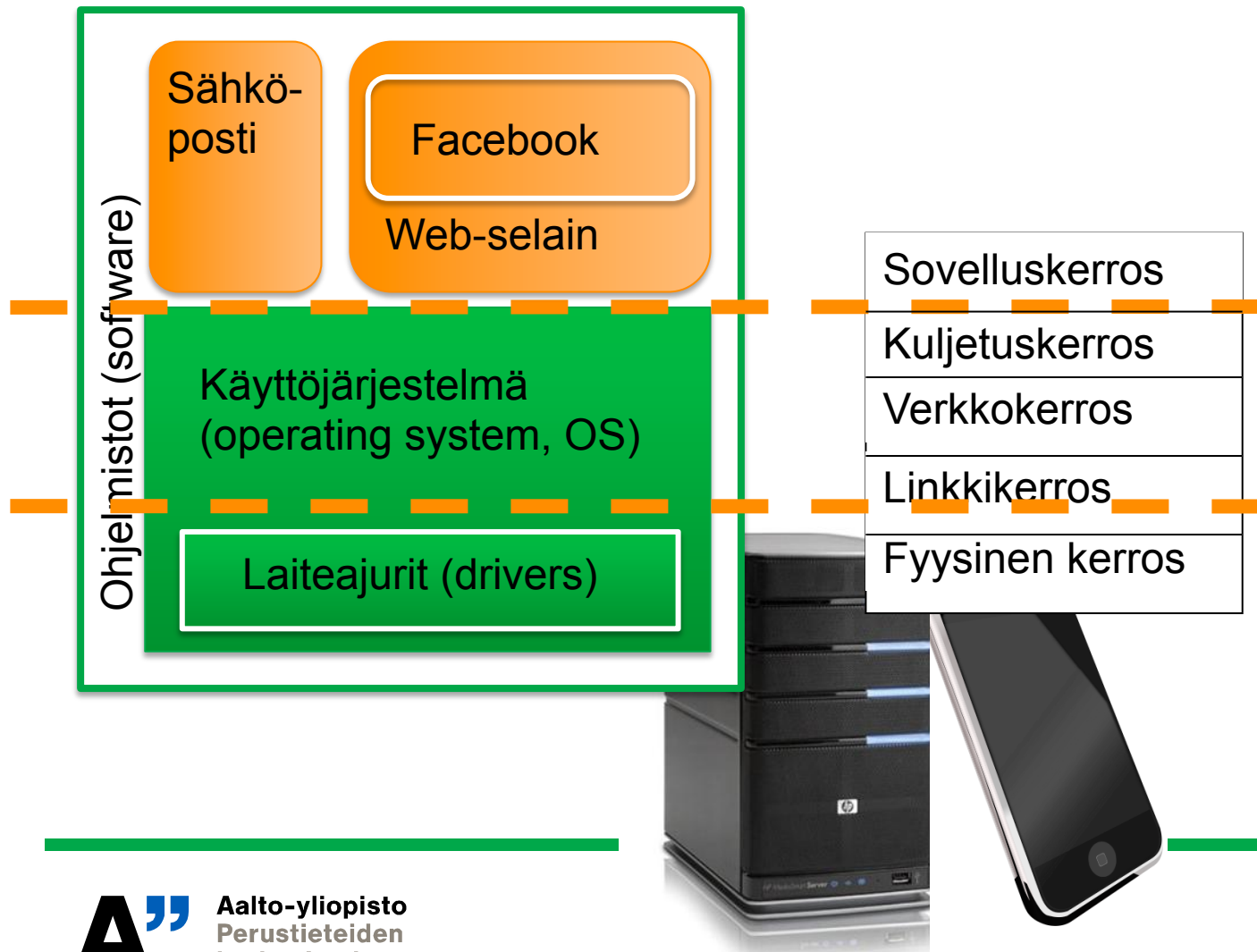
01.04.2014

Matti Siekkinen

Tällä luennolla

- • Lyhyet kertaukset aiemmista luennoista
- Kokonaiskuva
 - Miten kaikki palat toimivat yhteen?

Internet-protokollapino



Sovellukset

- Sovelluksissa tyypillisesti asiakas ja palvelin
 - Asiakas ja palvelin lähettävät viestejä toisilleen
 - Protokolla määrittää että miten toimitaan (millä viestillä vastataan tiettyyn viestiin tietyssä tilanteessa)
- Sovellukset ovat ylin kerros Internetin protokollapinossa
 - Alemmat kerrokset hoitavat viestien välityksen verkon yli
- Sähköposti
 - Asynkroninen
 - Asiakasohjelma ja palvelin juttelevat IMAP/POP3/HTTP-protokollan avulla
 - Sähköpostipalvelimet keskenään: SMTP-protokolla
 - MIME laajennus liitteille ja muulle ei perustekstille
 - Tietoturvaprotokollat varmistavat postin eheyden, luottamuksellisuuden ja autenttisuuden
 - Symmetriset ja asymmetristen avainten kryptoprotokollat
 - Sertifikaatit

Sovellukset

- WWW
 - Selain ja WWW-palvelin käyttävät HTTP(/HTTPS)-protokollaa
 - Standardimenetelmät sivujen koodaamiseen ja esittämiseen
 - HTML, XML, HTML5, JSON ...
 - Eri selaimet osaavat tulkita dataa ja tietävät miten sivu esitetään
 - URL: kertoo dokumentin sijainnin (palvelimen nimi sekä polku ja dokumentin nimi palvelimella)
 - Alunperin vain HTML –koodatut sivut, nykyään monet eri sovellukset toimivat HTTP:n päällä (esim. videostrimaus)
- Nimipalvelu DNS
 - Internetissä koneet käyttävät numeerisia IP-osoitteita mutta ihmiset muistavat nimiä
 - DNS muuntaa nimen (URL) IP-osoitteeksi
 - Hajautettu ja hierarkinen tietokanta
 - Palvelintasot: juuri, top-level domain, authoritative ja paikallinen
 - Rekursiivinen vs. iteratiivinen kysely
 - Välimuistin käyttö nopeuttaa olennaisesti toimintaa

Sovellusarkkitehtuurit

- Sovellusarkkitehtuureita on erilaisia
 - Asiakas-palvelin
 - Vertaisverkot
 - Pilvipalvelut voitaisiin vielä erotella näistä
- Vertaisverkot
 - Self scaling ominaisuus
 - Mitä enemmän käyttäjiä, sitä enemmän kapasiteettia
 - BitTorrentissa insentiivit jakaa rakennettu protokollaan
 - Tit for tat
 - Hajautettu tiivistetaulu (DHT) on avain-arvo tietokanta
 - Jokainen noodi tallentaa osan avain-arvo pareista
 - Muodostavat kuoriverkon

Verkko-ohjelmointi

- Sokettiohjelmointi
 - Sokettirajapinta sovellusten ja kulj. kerroksen välissä
- TCP ja UDP soketit
 - TCP ja UDP soketeilla ohjelmoidaan hieman eri tavalla
 - TCP soketissa pitää luoda palvelimella “tervetulosoketti”
 - Funktio luo ja palauttaa uuden soketin jokaisen yhteyden käsittelyyn
- Advanced topics
 - Blokkaava vs. blokkaamaton soketti
 - Blokkaamattomalla voidaan hoitaa usea soketti per säie
 - select() –funktion avulla voidaan käsitellä monia soketteja
 - Blokkaava usean soketin tarkistava funktio
 - Ei tarvita säiettä per soketti
 - Vältetään pollaamisen aiheuttama overhead
 - WebSocket on korkeamman tason abstraktio
 - Toteutus käyttää TCP sokettia

Kuljetuskerros

- Kuljetuskerros mahdollistaa sovellusten välisen viestinnän
 - TCP luotettava bittisiirtopalvelu
 - UDP epäluotettava viestinvälitys
- TCP ylivoimaisesti käytetyin protokolla
 - Virheenkorjaus (ARQ) ja vuonhallinta perusmekanismeja
- ARQ virheenkorjaus
 - Kuittaukset (sekä pos ja neg mahdollisia) ja uudelleenlähetykset
 - Lähetysikkuna ja putkitus
 - Kumulatiiviset ja selektiiviset kuittaukset
 - TCP käyttää tätä
- Vuonhallinta
 - Vastaanottava sovellus ei lue riittävän nopeasti dataa soketista → soketin muisti täyttyy
 - Vastaanottava TCP voi estää lähettäjää lähettämästä enempää dataa
 - Vastaanottajan ikkunankoko

Kuljetuskerros

- Ruuhkanhallinta
 - Pyritään estämään verkon ylikuormitus
 - Reitittimillä rajallinen pakettinvälityskapasiteetti → pudottavat paketteja jos ylittyy
 - Tässä tilanteessa pakko vähentää kuormaa → vaatii että päätelaitteet vähentävät lähetysnopeutta (TCP)
 - TCP:n eka versio ei sisältänyt ruuhkanhallintaa (ruuhka tuntematon käsite)
 - Monta TCP versiota → eri ruuhkanhallintamekanismit
 - Tasapuolisuus
 - TCP:n ruuhkanhallinta johtaa tietyissä tapauksissa tasapuolisuuteen muttei aina
 - Eri kiertoviiveet tai monta yhteyttä → sovellukset saavat eri määrän kaistaa
 - SSL/TLS
 - Kattava tietoturvaprotokolla sovelluksen ja TCP välissä
-

Verkkokerros

IP-protokolla

- Verkkokerroksen IP-protokolla toteuttaa epäluotettavan ja tilattoman pakettienvälityspalvelun Internetissä
 - Luotettavuus toteutetaan kuljetuskerroksella (ylempänä eli TCP)
 - Paketit välitetään perille niissä olevan osoitteen perusteella kukin itsenäisesti
 - Jokainen reititin “puhuu IP protokollaa”
 - Vrt. vain päätelaitteet ajavat kuljetuskerroksen protokollaa (TCP)!
- IPv4 osoitteet ovat lopussa, mutta protokolla yhä käytössä
 - DHCP ja NAT auttavat
 - IPv6 käytössä jo

Verkkokerros

Reititys ja pakettien välitys

- Edelleenlähetys/välitys (forwarding): Etsitään reititystaulusta pisin vastaanottajan IP-osoitetta vastaava verkko-osoite (prefix)
 - Pitää tehdä joka paketille → erittäin aikakriittistä!
- Reititys (routing): sopivimpien reittien etsiminen IP-osoitteiden välille
 - Muodostetaan reititystaulut, ei niin aikakriittistä
- Internet on jaettu autonomisiin järjestelmiin
 - AS:n välillä reititys on sopimuspohjaista
 - Reitityksen skaalautumiseksi tarvitaan hierarkiatasoja
 - Tehokkuus tärkeintä yhden toimijan verkossa
- AS:n välillä käytetään BGP4:ää
- AS:n sisällä voi käyttää mitä haluaa
 - Staattinen, linkkilareititys (esim. OSPF), etäisyysvektoreititys (esim. RIP)
- Monilähetyksessä verkko monistaa viestin kaikille tiettyyn ryhmään kuuluville tai siihen liittyneille mahdollisimman lähellä vastaanottajaa
 - Tehostaa verkon käyttöä

Linkkikerros

Perusteet

- Periaatteet datalinkkikerroksen palveluissa:
 - virheiden havainnointi, korjaus
 - usein epäluotettava palvelu (Ethernet), joskus luotettava (Wi-Fi)
- Monipääsyprotokollia tarvitaan jaetuissa kanavissa
 - ”Kuka saa milloinkin puhua”
 - Kanavan osittaminen (matkapuh.verkot), satunnaispääsy (Ethernet ja Wi-Fi), ”vuorottelu”
- MAC osoitteet
 - Uniikkeja, ei vaihdu kuten IP osoitteet
 - ARP: IP → MAC osoitteenhaku
- Ethernet
 - Yleisin langallinen lähiverkkoteknologia
 - IEEE 802.3 standardi, erilaisia versioita (siirtonopeus, fyysinen siirtotie)

Linkkikerros

Layer-2 verkot

- Bridging on linkkikerroksen lähiverkkosegmenttien verkottamista
 - Pääasiassa käytetään kytkimiä ja keskittimiä
 - Kytkin on itseoppiva (ei tarvita reititysprotokollaa)
- Kytketyissä verkoissa voi esiintyä välityssilmukoita
 - Kehykset kiertävät loputtomasti verkossa
 - Tarvitaan Spanning Tree Protocol (STP) tms.
 - Muodostaa silmukattoman topologian
 - Uudempia korvaavia protokollia jo olemassa
- Virtual LAN (VLAN)
 - Verkon osittaminen yhdellä laitteella
 - Helpottaa verkonhallintaa ja tehostaa laitteiden käyttöä
- MPLS mahdollistaa joustavan pakettien reitityksen
 - Liikenteenohjaus (traffic engineering) ja varareitit

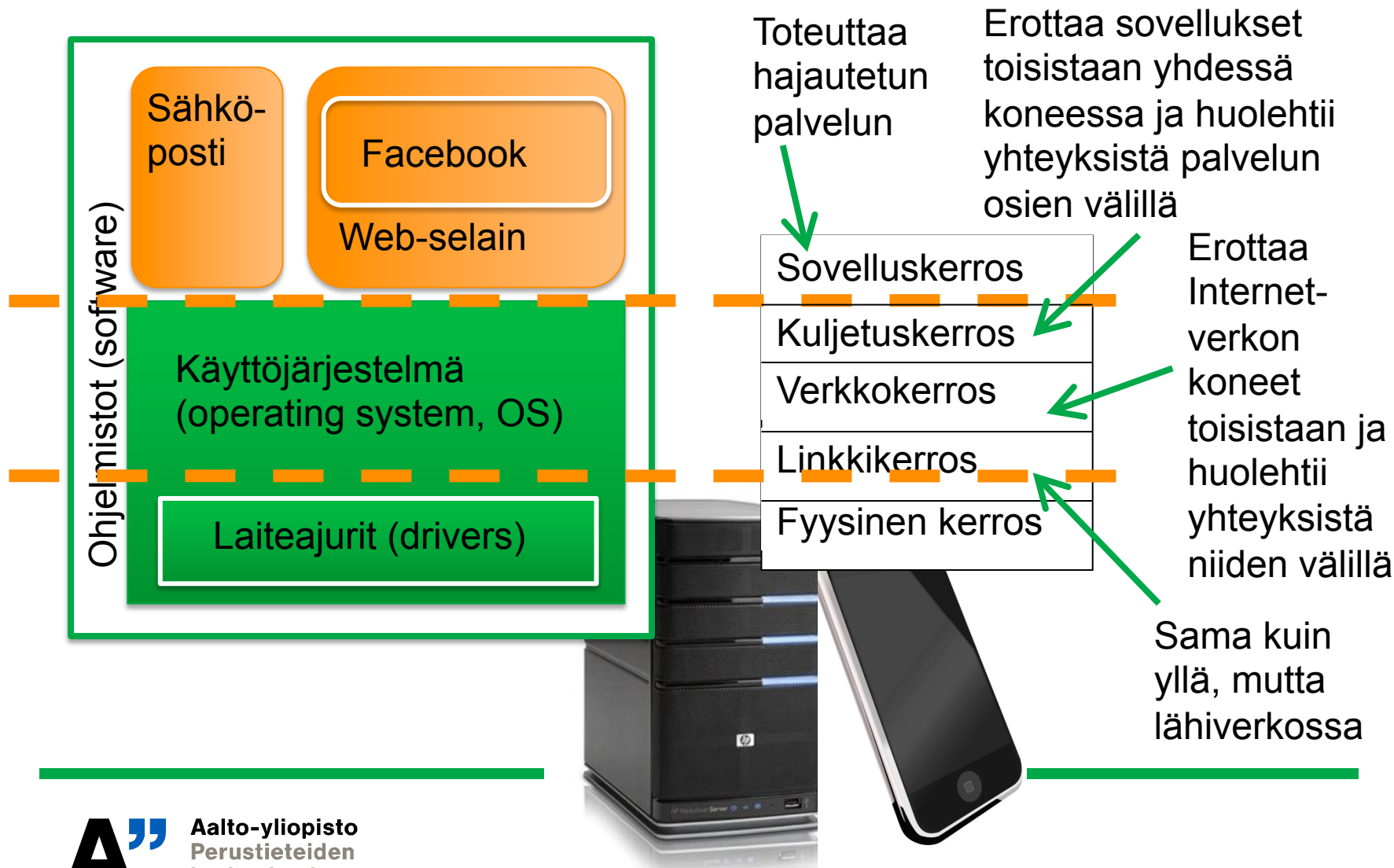
Langattomat verkot

- Langattomat verkot
 - Infrastruktuuri: tukiasemia joihin langattomat laitteet yhteydessä
 - Langaton tiedonsiirto: kaistanleveys, signaalikohinasuhde, modulaatio
- IEEE 802.11 standardi eli Wi-Fi
 - Langaton lähiverkkoteknologia (PHY/MAC kerrokset)
 - Monta versiota: suurimmat erot fyysisen kerroksen tekniikoissa (lähetysnopeus)
 - Kaksi eri taajuusaluetta, jaettu kanaviin
 - Satunnaispääsy kanavalle käyttäen CSMA/CA ja joskus RTS/CTS
- Soluverkot
 - Koostuu soluista joissa tukiasema
 - GSM (2G) → UMTS (3G) → LTE (4G)
 - 3G:ssä käytetään FDMA:n ja TDMA:n lisäksi CDMA monipääsyä
- Liikkuvuudenhallinta verkoissa
 - Osoite muuttuu verkosta toiseen liikuttaessa
 - Agentit pitävät kirjaa olinpaikasta
 - Epäsuora vs. suora reititys

Verkon suunnittelu ja hallinta

- SNMP-arkkitehtuuri
- Management Information Base
- SNMP-protokolla
- Verkonhallinta käytännössä

Internet-protokollapino



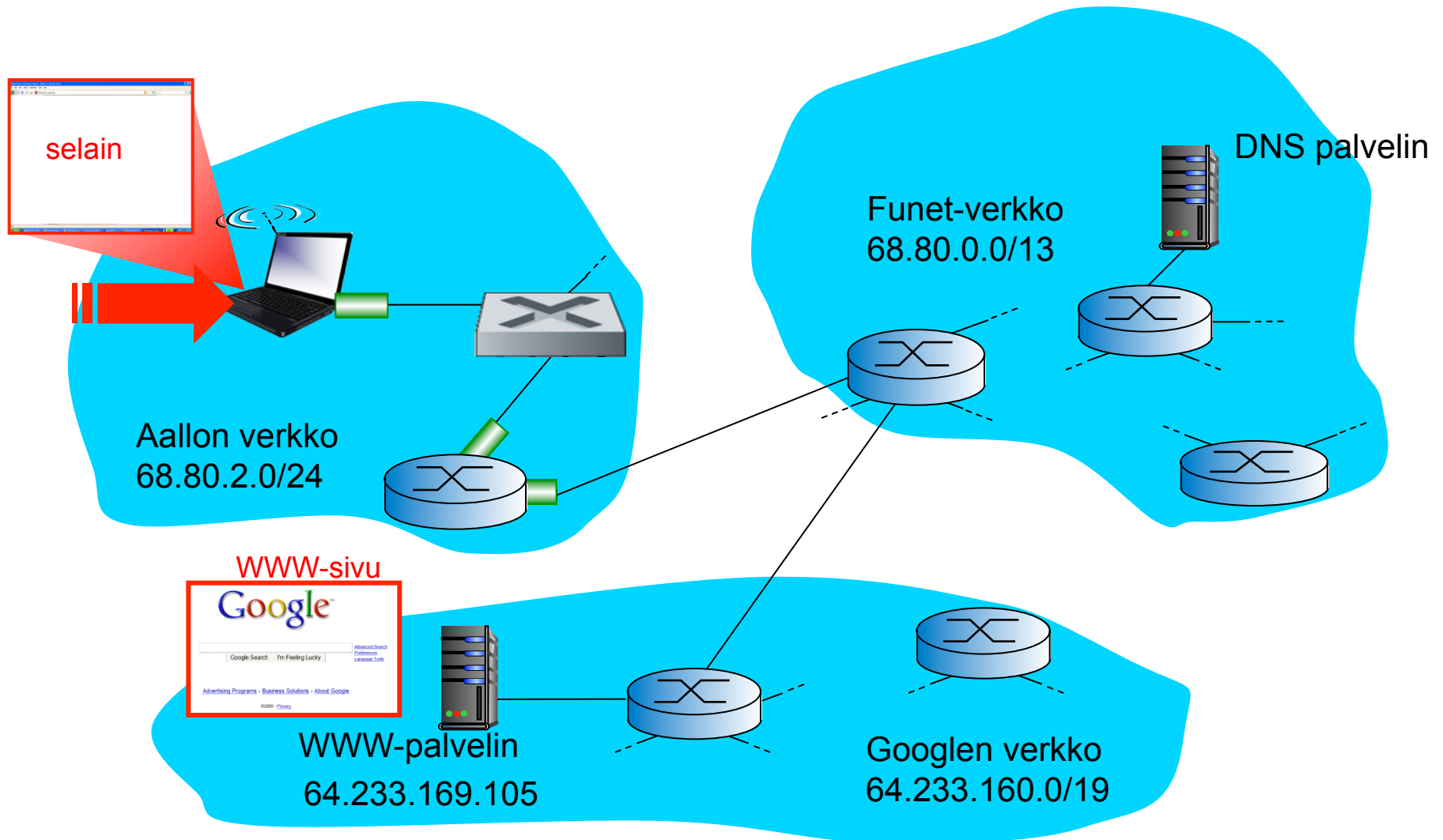
Tällä luennolla

- Lyhyet kertaukset aiemmista luennoista
- • Kokonaiskuva
 - Miten kaikki palat toimivat yhteen?

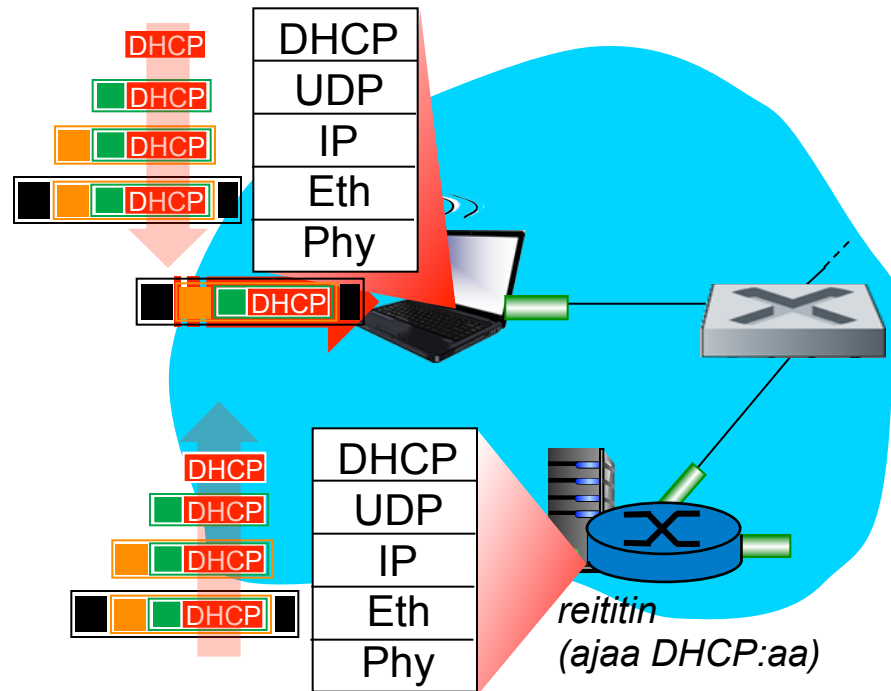
Synteesi: “A day in the life of a web request”

- Koko protokollapino käsitelty
 - application, transport, network, link
- Miten kaikki toimii yhteen?
 - **Tavoite**: ymmärrys siitä mitkä ja miten eri kerrosten protokollat osallistuvat yksinkertaisen näköiseen skenaarioon: webbisivun lataaminen
 - **Skenaario**: opiskelija yhdistää läppärin kampusen verkkoon ja avaa sivun www.google.com

Skenaario

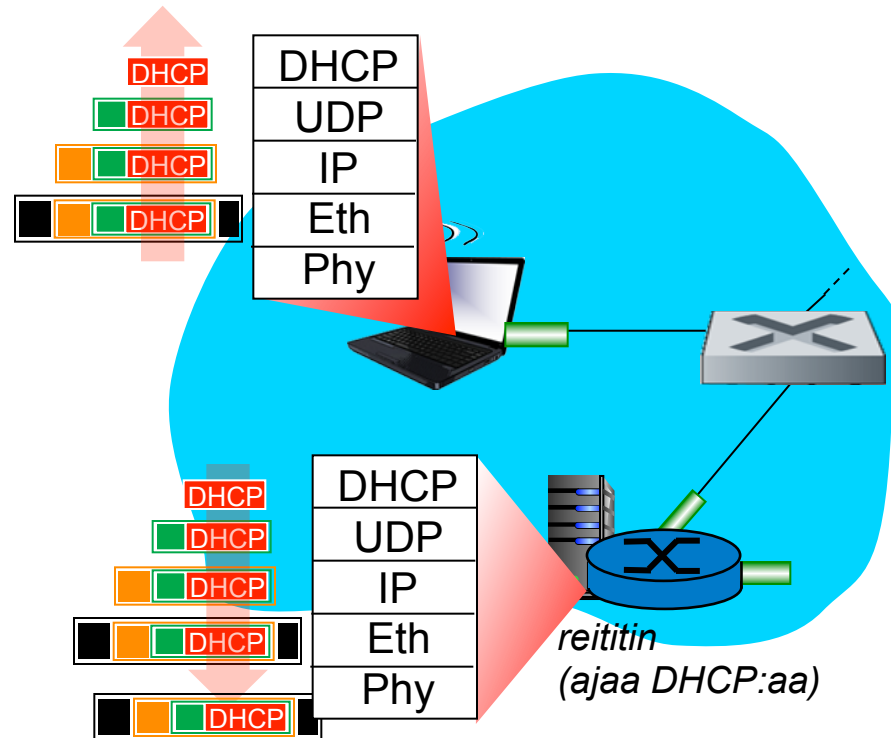


Yhdistäminen Internetiin



- läppäri tarvitsee oman IP-osoitteen, ekan reitittimen osoitteen ja DNS serverin osoitteen: käytä **DHCP**
- ❖ DHCP pyyntö kapseloitu **UDP-pakettiin**, joka kapseloitu **IP-pakettiin**, joka kapseloitu **802.3 Ethernet-kehukseen**
- ❖ Ethernet-kehys **yleislähetetään** (osoite: FFFFFFFFFFFFFFFF), reititin joka ajaa **DHCP** serveriä vastaanottaa
- ❖ Ethernet **dekapseloidaan** → IP dekaps. → UDP → DHCP

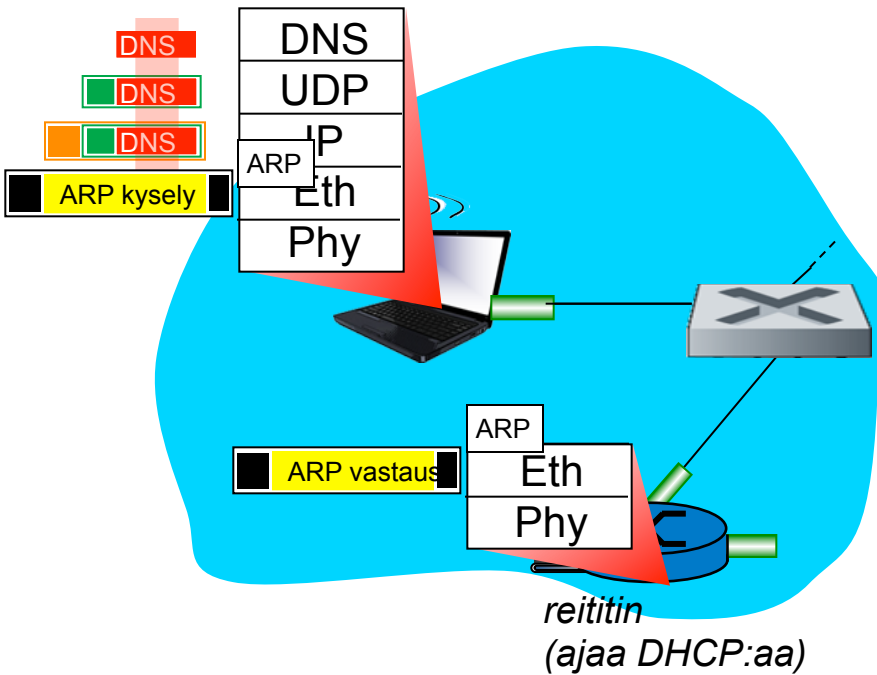
Yhdistäminen Internetiin



- DHCP-palvelin luo **DHCP ACK**-viestin joka sisältää
 - asiakkaan IP-osoitteen
 - ekan reitittimen IP-osoitteen
 - DNS-palvelimen nimen ja IP-osoitteen
- ❖ palvelin kapseloi viestin ja lähettää Eth-kehyksessä (**kytkin itseoppii MAC osoitteet**) lähiverkon läpi, asiakas dekapseloi
- ❖ DHCP-asiakas vastaanottaa DHCP ACK -viestin

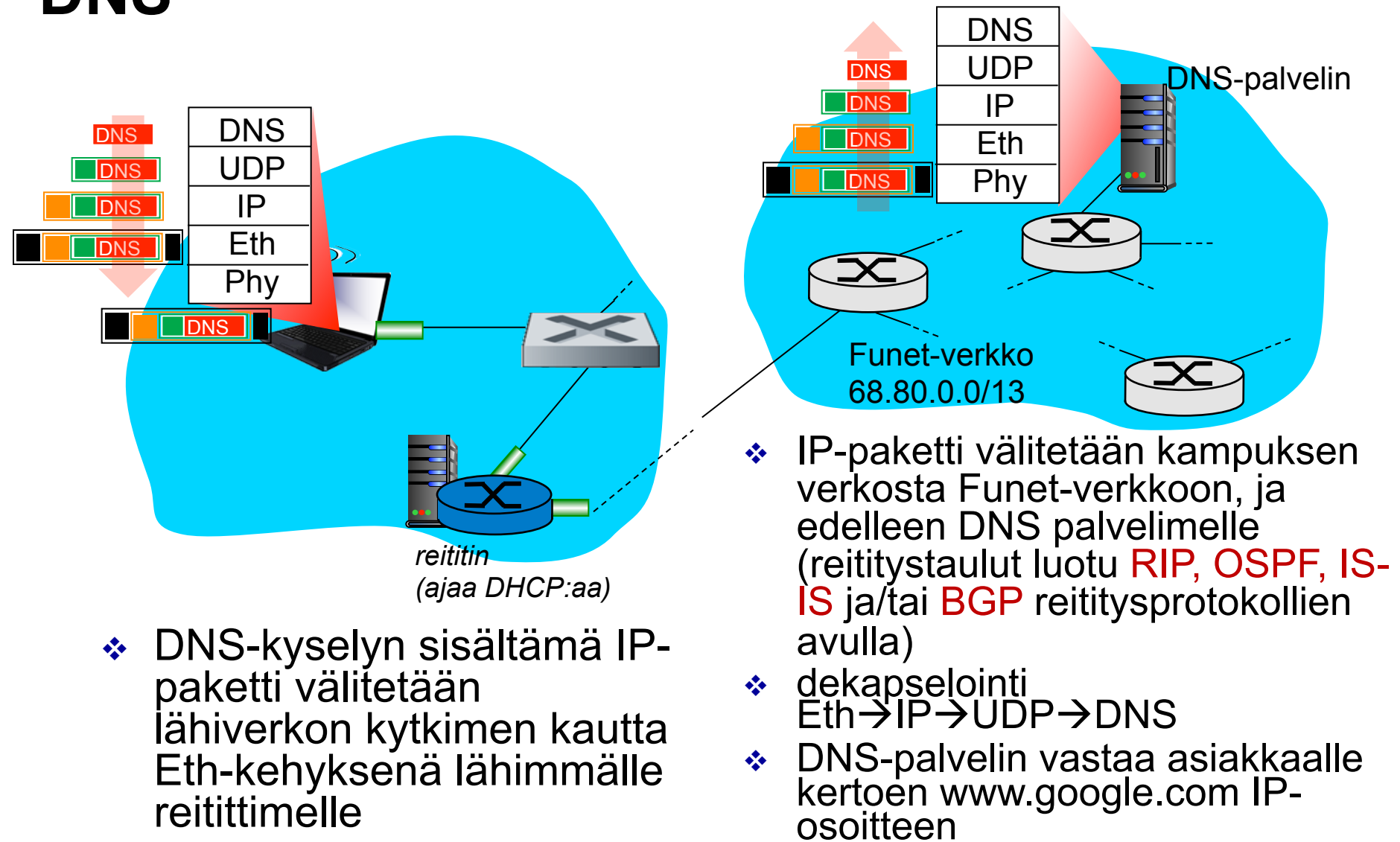
Asiakkaalla on nyt IP-osoite, se tietää DNS-palvelimen nimen ja osoitteen, sekä lähimmän reitittimen IP-osoitteen

ARP (ennen DNS ja HTTP)

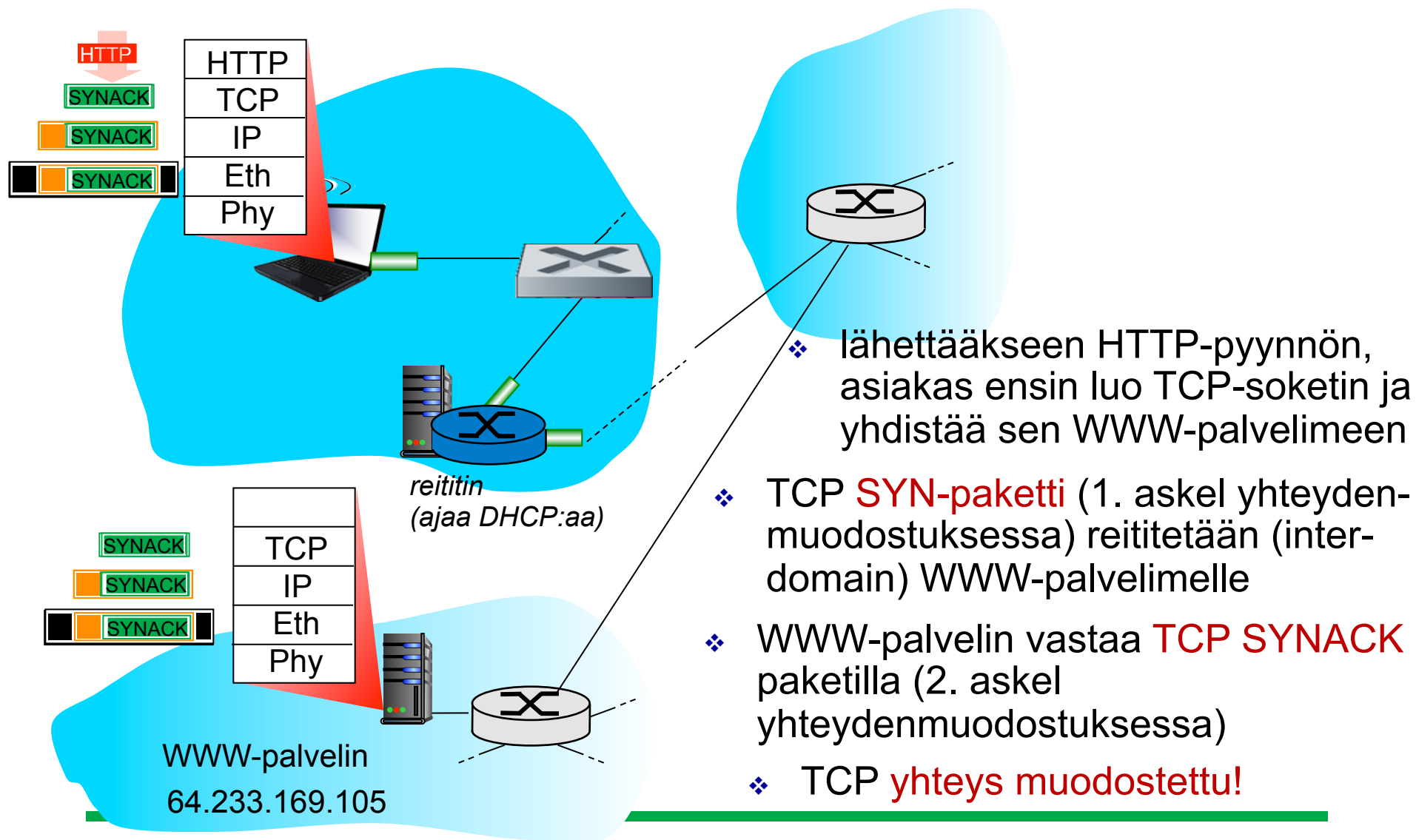


- ennen **HTTP**-kyselyn lähettämistä tarvitaan www.google.com nimeä vastaava IP osoite: **DNS**
- ❖ luodaan DNS-kysely, kapseloidaan UDP datagrammiin, kapseloidaan IP-pakettiin, kapseloidaan Eth.-kehukseen
- ❖ tarvitaan lähimmän reitittimen MAC-osoite jotta voidaan lähettää Eth-kehys: **ARP**
- ❖ **ARP-kysely** yleislähetetään, reititin vastaanottaa ja vastaa **ARP reply** -viestillä jossa kyseisen verkkokortin MAC-osoite
- ❖ asiakas tietää lähimmän reitittimen MAC-osoitteen joten voi lähettää DNS-kyselyn sisältävän Eth-kehuksen

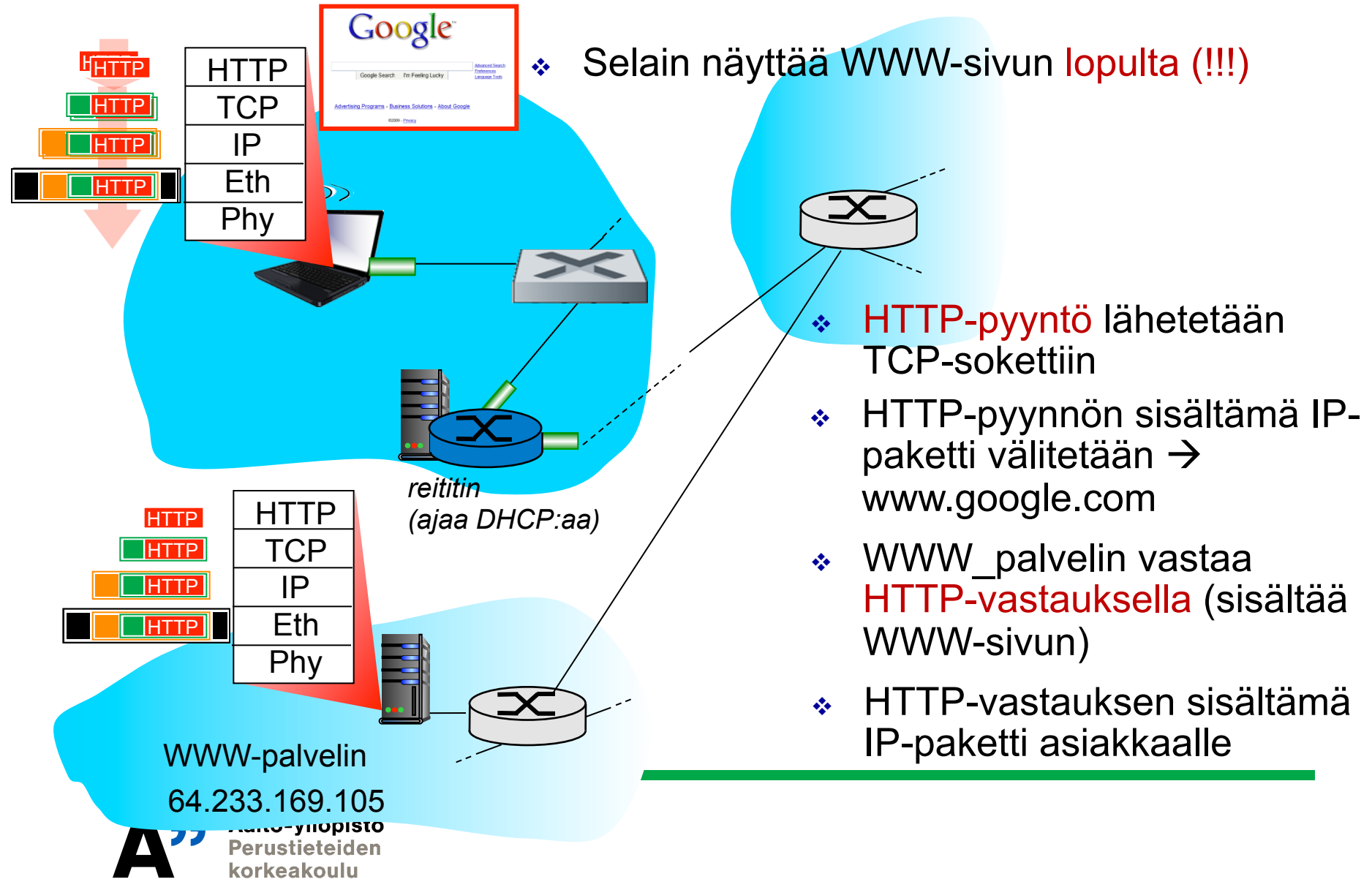
DNS



TCP-yhteys



HTTP kysely/vastaus



Kysymyksiä?